# 1 Robustness and Regularization

## *1.1.2*

### Question:

Note that we are taking the gradient of $\mathcal{L}(f(\mathbf{x}; \mathbf{w}), y)$ with respect to the input $\mathbf{x}$ instead of the weights $\mathbf{w}$, and that we are adding this gradient rather than subtracting it since the goal here is to increase the loss on $\mathbf{x}'$.

For the rest of the question, we assume we are dealing with a binary linear classifier that outputs a scalar logit as follows:

$$f(\mathbf{x}; \mathbf{w}) = \mathbf{w}^\top \mathbf{x} \tag{1}$$

where $\mathbf{w} \in \mathbb{R}^d$ where $d$ is dimension of the input $\mathbf{x}$, so $f : \mathbb{R}^d \to \mathbb{R}$. For the remainder of the question, we ignore the loss function, and simply try to reduce the output predicted by the classifier $f$.

If we remove the $\operatorname{sgn}()$ function from the FGSM, we are left with just the FGM

$$\mathbf{x}' \leftarrow \mathbf{x} - \epsilon \nabla_{\mathbf{x}} f(\mathbf{x}; \mathbf{w}) \tag{2}$$

Let us construct $\mathbf{x}'$ using the FGM. Write down the model output under the adversarial attack $f(\mathbf{x}'; \mathbf{w})$ as a function of $\epsilon, \mathbf{x}, \mathbf{w}$ in a closed form.

### Solution:

Using the FGM, we can construct the adversarial example $\mathbf{x}'$ as follows:

$$\mathbf{x}' \leftarrow \mathbf{x} - \epsilon \nabla_{\mathbf{x}} f(\mathbf{x}; \mathbf{w}) \tag{3}$$

Substituting the expression for the linear classifier $f(\mathbf{x}; \mathbf{w}) = \mathbf{w}^\top \mathbf{x}$, we get:

$$\begin{aligned} f(\mathbf{x}'; \mathbf{w}) &= \mathbf{w}^\top \mathbf{x}' \\ &= \mathbf{w}^\top (\mathbf{x} - \epsilon \nabla_{\mathbf{x}} f(\mathbf{x}; \mathbf{w})) \\ &= \mathbf{w}^\top \mathbf{x} - \epsilon \mathbf{w}^\top \nabla_{\mathbf{x}} f(\mathbf{x}; \mathbf{w}) \\ &= f(\mathbf{x}; \mathbf{w}) - \epsilon \nabla_{\mathbf{x}} f(\mathbf{x}; \mathbf{w})^\top \mathbf{w} \end{aligned} \tag{4}$$

Therefore, the model output under the adversarial attack $f(\mathbf{x}'; \mathbf{w})$ is given by:

$$f(\mathbf{x}'; \mathbf{w}) = f(\mathbf{x}; \mathbf{w}) - \epsilon \nabla_{\mathbf{x}} f(\mathbf{x}; \mathbf{w})^\top \mathbf{w} \tag{5}$$

Since $f(\mathbf{x}; \mathbf{w}) = \mathbf{w}^\top \mathbf{x}$, we can use the chain rule to simplify the gradient as follows:

$$\nabla_{\mathbf{x}} (\mathbf{w}^\top \mathbf{x}) = \mathbf{w} \tag{6}$$

Expanding the dot product and simplifying, we obtain:

$$f(\mathbf{x}'; \mathbf{w}) = \mathbf{w}^\top \mathbf{x} - \epsilon \|\mathbf{w}\|^2 \tag{7}$$

DO Note that we are taking the gradient of $\mathcal{L}(f(\mathbf{x} ; \mathbf{w}), y)$ with respect to the input $\mathbf{x}$ instead of the weights $\mathbf{w}$, and that

we are adding this gradient rather than subtracting it since the goal here is to increase the loss on $\mathbf{x}^{\prime}$.

For the rest of the question, we assume we are dealing with a binary linear classifier that outputs a scalar logit as follows:
$$
f(\mathbf{x} ; \mathbf{w})=\mathbf{w}^{\top} \mathbf{x}
$$
where $\mathbf{w} \in \mathbb{R}^d$ where $d$ is dimension of the input $\mathbf{x}$, so $f: \mathbb{R}^d \rightarrow \mathbb{R}$. For the remainder of the question, we ignore the loss function, and simply try to reduce the output predicted by the classifier $f$.

If we remove the $\operatorname{sgn}()$ function from the FGSM, we are left with just the FGM
$$
\mathbf{x}^{\prime} \leftarrow \mathbf{x}-\epsilon \nabla_{\mathbf{x}} f(\mathbf{x} ; \mathbf{w})
$$
Let us construct $\mathbf{x}^{\prime}$ using the FGM. Write down the model output under the adversarial attack $f\left(\mathbf{x}^{\prime} ; \mathbf{w}\right)$ as a function of $\epsilon, \mathbf{x}, \mathbf{w}$ in a closed form.

Solution:

Starting from the definition of FGM, we have:

$$\mathbf{x}' \leftarrow \mathbf{x} - \epsilon \nabla_{\mathbf{x}} f(\mathbf{x}; \mathbf{w})$$

Substituting the expression for $f(\mathbf{x} ; \mathbf{w})$, we get:

$$\mathbf{x}' \leftarrow \mathbf{x} - \epsilon \nabla_{\mathbf{x}} (\mathbf{w}^\top \mathbf{x})$$

Since $\mathbf{w}^{\top} \mathbf{x}$ is a scalar, we can use the chain rule to simplify the gradient as follows:

$$\nabla_{\mathbf{x}} (\mathbf{w}^\top \mathbf{x}) = \mathbf{w}$$

Substituting this back into the FGM equation, we obtain:

$$\mathbf{x}' \leftarrow \mathbf{x} - \epsilon \mathbf{w}$$

Finally, we can compute the output of the model under the adversarial attack $f\left(\mathbf{x}^{\prime} ; \mathbf{w}\right)$ as follows:

$$f(\mathbf{x}'; \mathbf{w}) = \mathbf{w}^\top \mathbf{x}' = \mathbf{w}^\top (\mathbf{x} - \epsilon \mathbf{w})$$

Expanding the dot product and simplifying, we obtain:

$$f(\mathbf{x}'; \mathbf{w}) = \mathbf{w}^\top \mathbf{x} - \epsilon \|\mathbf{w}\|^2$$

Therefore, we have derived a closed form expression for the output of the linear classifier under the FGM attack.

## *1.2.2*

### Question

The most trivial though impractical way of making a classifier robust to adversarial examples is to set $\mathbf{w} = \mathbf{0}$ such that $f(\mathbf{x}; \mathbf{w}) = 0$ for any $\mathbf{x}$. However, this just computes a constant function, and is not useful. Intuitively, it looks like the smaller the norm of $\mathbf{w}$, then less the output will change when changing $\mathbf{x}$. We explore if this is always the case.

Suppose we have a design matrix $X \in \mathbb{R}^{n \times d}$ where $n$ is the number of samples and $d$ is the dimensionality, and a target vector $\mathbf{t} \in \mathbb{R}^n$. We can define the objective of linear regression as

$$\min_{\mathbf{w}} \frac{1}{2n} \|X\mathbf{w} - \mathbf{t}\|_2^2 \tag{8}$$

If we penalize the squared $\ell^2$ norm of the weights, we end up with ridge regression:

$$\mathbf{w}_{\text{ridge}}^* = \arg\min_{\mathbf{w}} \frac{1}{2n} \|X\mathbf{w} - \mathbf{t}\|_2^2 + \lambda \|\mathbf{w}\|_2^2 \tag{9}$$

where $\lambda$ is the weight decay coefficient, $\lambda > 0$.

Recall the solution to plain regression is $\mathbf{w}^* = \left(X^\top X\right)^{-1} X^\top \mathbf{t}$. Write down the closed-form solution to ridge regression in matrix form, $\mathbf{w}_{\text{ridge}}^*$ . Show your work.

### Solution

To find the closed-form solution to ridge regression in matrix form, we need to solve the following optimization problem:

$$\mathbf{w}_{\text{ridge}}^* = \arg\min_{\mathbf{w}} \frac{1}{2n} \|X\mathbf{w} - \mathbf{t}\|_2^2 + \lambda \|\mathbf{w}\|_2^2 \tag{10}$$

Expanding the squared Euclidean norm term, we have

$$\begin{aligned} \frac{1}{2n} |X\mathbf{w} - \mathbf{t}|_2^2 + \lambda |\mathbf{w}|_2^2 &= \frac{1}{2n} \left(\mathbf{w}^\top X^\top X\mathbf{w} - 2\mathbf{w}^\top X^\top \mathbf{t} + \mathbf{t}^\top \mathbf{t}\right) + \lambda |\mathbf{w}|_2^2 \\ &= \frac{1}{2n} \mathbf{w}^\top X^\top X\mathbf{w} - \frac{1}{n} \mathbf{w}^\top X^\top \mathbf{t} + \frac{1}{2n} \mathbf{t}^\top \mathbf{t} + \lambda |\mathbf{w}|_2^2 \end{aligned} \tag{11}$$

To find the minimum of this function with respect to $\mathbf{w}$, we can take its derivative and set it to zero:

$$\frac{\partial}{\partial \mathbf{w}} \left[ \frac{1}{2n} \mathbf{w}^\top X^\top X\mathbf{w} - \frac{1}{n} \mathbf{w}^\top X^\top \mathbf{t} + \frac{1}{2n} \mathbf{t}^\top \mathbf{t} + \lambda \|\mathbf{w}\|_2^2 \right] = \frac{1}{n} X^\top (X\mathbf{w} - \mathbf{t}) + 2\lambda \mathbf{w} = 0 \tag{12}$$

Multiplying both sides by $n$ and rearranging, we get

$$X^\top X\mathbf{w} + \lambda n\mathbf{w} = X^\top \mathbf{t} \tag{13}$$

Substituting $\mathbf{A} = X^\top X + \lambda n\mathbf{I}$ and $\mathbf{b} = X^\top \mathbf{t}$, where $\mathbf{I}$ is the identity matrix, we get

$$\mathbf{A}\mathbf{w} = \mathbf{b} \tag{14}$$

and the closed-form solution is

$$\mathbf{w}^*_{\text{ridge}} = \mathbf{A}^{-1}\mathbf{b} = \left(X^\top X + \lambda n\mathbf{I}\right)^{-1} X^\top \mathbf{t} \tag{15}$$

DO

The most trivial though impractical way of making a classifier robust to adversarial examples is to set $\mathbf{w}=\mathbf{0}$ such that $f(\mathbf{x} ; \mathbf{w})=0$ for any $\mathbf{x}$. However, this just computes a constant function, and is not useful. Intuitively, it looks like the smaller the norm of $\mathbf{w}$, then less the output will change when changing $\mathbf{x}$. We explore if this is always the case.

Suppose we have a design matrix $X \in \mathbb{R}^{n \times d}$ where $n$ is the number of samples and $d$ is the dimensionality, and a target vector $\mathbf{t} \in \mathbb{R}^n$. We can define the objective of linear regression as

$$
\min _{\mathbf{w}} \frac{1}{2 n}\|X \mathbf{w}-\mathbf{t}\|_2^2
$$

If we penalize the squared $\ell^2$ norm of the weights, we end up with ridge regression:

$$
\mathbf{w}_{\text {ridge }}^*=\arg \min _{\mathbf{w}} \frac{1}{2 n}\|X \mathbf{w}-\mathbf{t}\|_2^2+\lambda\|\mathbf{w}\|_2^2
$$

where $\lambda$ is the weight decay coefficient, $\lambda>0$.

Recall the solution to plain regression is $\mathbf{w}^*=\left(X^{\top} X\right)^{-1} X^{\top} \mathbf{t}$. Write down the closed-form solution to ridge regression in matrix form, $\mathbf{w}_{\text {ridge }}^*$. Show your work.

To find the closed-form solution to ridge regression in matrix form, we need to solve the following optimization problem:

$$\mathbf{w}^*_{\text{ridge}} = \arg\min_{\mathbf{w}} \frac{1}{2n}\|X\mathbf{w} - \mathbf{t}\|_2^2 + \lambda\|\mathbf{w}\|_2^2$$

Expanding the squared Euclidean norm term, we have

$$
\begin{aligned}
\frac{1}{2 n}|X \mathbf{w}-\mathbf{t}|_2^2+\lambda|\mathbf{w}|_2^2 &=\frac{1}{2 n}\left(\mathbf{w}^{\top} X^{\top} X \mathbf{w}-2 \mathbf{w}^{\top} X^{\top} \mathbf{t}+\mathbf{t}^{\top} \mathbf{t}\right)+\lambda|\mathbf{w}|_2^2 \\
&=\frac{1}{2 n} \mathbf{w}^{\top} X^{\top} X \mathbf{w}-\frac{1}{n} \mathbf{w}^{\top} X^{\top} \mathbf{t}+\frac{1}{2 n} \mathbf{t}^{\top} \mathbf{t}+\lambda|\mathbf{w}|_2^2 \\
\end{aligned}
$$

To find the minimum of this function with respect to $\mathbf{w}$, we can take its derivative and set it to zero:

$$\frac{\partial}{\partial \mathbf{w}}\left[\frac{1}{2n}\mathbf{w}^\top X^\top X\mathbf{w} - \frac{1}{n}\mathbf{w}^\top X^\top \mathbf{t} + \frac{1}{2n}\mathbf{t}^\top \mathbf{t} + \lambda\|\mathbf{w}\|_2^2\right] = \frac{1}{n}X^\top(X\mathbf{w}-\mathbf{t}) + 2\lambda\mathbf{w} = 0$$

Multiplying both sides by $n$ and rearranging, we get

$$X^\top X\mathbf{w} + \lambda n\mathbf{w} = X^\top \mathbf{t}$$

Substituting $\mathbf{A}=X^{\top} X+\lambda n \mathbf{I}$ and $\mathbf{b}=X^{\top} \mathbf{t}$, where $\mathbf{I}$ is the identity matrix, we get

$$\mathbf{A}\mathbf{w} = \mathbf{b}$$

and the closed-form solution is

$$\mathbf{w}^*_{\text{ridge}} = \mathbf{A}^{-1}\mathbf{b} = \left(X^\top X + \lambda n\mathbf{I}\right)^{-1} X^\top \mathbf{t}$$

# 2 Trading off Resources in Neural Net Training

## *2.1.1*

### Question

Batch size affects the stochasticity in optimization, and therefore affects the choice of learning rate. We demonstrate this via a simple model called the noisy quadratic model (NQM). Despite the simplicity, the NQM captures many essential features in realistic neural network training Zhang et al. [2019].

For simplicity, we only consider the scalar version of the NQM. We have the quadratic loss $\mathcal{L}(w) = \frac{1}{2}aw^2$, where $a > 0$ and $w \in \mathbb{R}$ is the weight that we would like to optimize. Assume that we only have access to a noisy version of the gradient - each time when we make a query for the gradient, we obtain $g(w)$, which is the true gradient $\nabla\mathcal{L}(w)$ with additive Guassian noise:

$$g(w) = \nabla\mathcal{L}(w) + \epsilon, \quad \epsilon \sim \mathcal{N}\left(0, \sigma^2\right). \tag{16}$$

One way to reduce noise in the gradient is to use minibatch training. Let $B$ be the batch size, and denote the minibatch gradient as $g_B(w)$ :

$$g_B(w) = \frac{1}{B}\sum_{i=1}^{B} g_i(w), \quad \text{where } g_i(w) = \nabla\mathcal{L}(w) + \epsilon_i, \quad \epsilon_i \overset{i.i.d.}{\sim} \mathcal{N}\left(0, \sigma^2\right). \tag{17}$$

As batch size increases, how do you expect the optimal learning rate to change? Briefly explain in 2-3 sentences. (Hint: Think about how the minibatch gradient noise change with $B$.)

## Solution

As the batch size $B$ increases, the noise in the minibatch gradient $g_B(w)$ decreases, due to the averaging of the individual gradients with independent noise terms. Consequently, the optimal learning rate can be increased, because the reduced noise in the gradient estimates allows for larger steps to be taken in the optimization process without causing instability.

DO

Batch size affects the stochasticity in optimization, and therefore affects the choice of learning rate. We demonstrate this via a simple model called the noisy quadratic model (NQM). Despite the simplicity, the NQM captures many essential features in realistic neural network training Zhang et al. [2019].

For simplicity, we only consider the scalar version of the NQM. We have the quadratic loss $\mathcal{L}(w)=\frac{1}{2} a w^2$, where $a>0$ and $w \in \mathbb{R}$ is the weight that we would like to optimize. Assume that we only have access to a noisy version of the gradient - each time when we make a query for the gradient, we obtain $g(w)$, which is the true gradient $\nabla \mathcal{L}(w)$ with additive Guassian noise:

$$
g(w)=\nabla \mathcal{L}(w)+\epsilon, \quad \epsilon \sim \mathcal{N}\left(0, \sigma^2 \right) .
$$

One way to reduce noise in the gradient is to use minibatch training. Let $B$ be the batch size, and denote the minibatch gradient as $g_B(w)$ :

$$
g_B(w)=\frac{1}{B} \sum_{i=1}^B g_i(w), \quad \text { where } g_i(w)=\nabla \mathcal{L}(w)+\epsilon_i, \quad \epsilon_i \stackrel{i . i . d .}{\sim} \mathcal{N}\left(0, \sigma^2\right) .
$$

As batch size increases, how do you expect the optimal learning rate to change? Briefly explain in 2-3 sentences.
(Hint: Think about how the minibatch gradient noise change with $B$.)

As the batch size $B$ increases, the noise in the minibatch gradient $g_B(w)$ decreases, due to the averaging of the individual gradients with independent noise terms. Consequently, the optimal learning rate can be increased, because the reduced noise in the gradient estimates allows for larger steps to be taken in the optimization process without causing instability.

# *2.1.2*

## Question (a)

For the three points (A, B, C) on Figure 2, which one has the most efficient batch size (in terms of best resource and training time trade-off)? Assume that you have access to scalable (but not free) compute such that minibatches are parallelized efficiently. Briefly explain in 1-2 sentences.

## Solution

**Point C** has the most efficient batch size in terms of resource and training time trade-off, as it achieves a desired validation loss with fewer training steps and a moderate batch size. Larger batch sizes may not lead to a significant decrease in training steps, while smaller batch sizes may considerably increase the number of training steps required.

## Question (b)

> **In order to prompt ChatGPT, I included my observation from the figure as hint.**

In the curvature dominated regime, the bottleneck of optimization is the ill-conditioned loss landscape there are often two regimes in neural network training: the noise dominated regime and the curvature dominated regime. In the noise dominated regime, the bottleneck for optimization is that there exists a large amount of gradient noise. In the curvature dominated regime, the bottleneck of optimization is the ill-conditioned loss landscape. For points A and B on Figure 2, which regimes do they belong to, and what would you do to accelerate training? Fill each of the blanks with one best suited option.

## Solution

**Point A:** Regime: noise dominated. Potential way to accelerate training: seek parallel compute.

**Point B:** Regime: curvature dominated. Potential way to accelerate training: use higher order optimizers.

DO

In the curvature dominated regime, the bottleneck of optimization is the ill-conditioned loss
landscape there are often two regimes in neural network training: the noise dominated regime and the curvature dominated regime. In the noise dominated regime, the bottleneck for optimization is that there exists a large amount of gradient noise. In the curvature dominated regime, the bottleneck of optimization is the ill-conditioned loss landscape. For points A and B on Figure 2, which regimes do they belong to, and what would you do to accelerate training? Fill each of the blanks with one best suited option.
Point A: Regime: . Potential way to accelerate training: .
Point B: Regime: . Potential way to accelerate training: .

- Regimes: noise dominated / curvature dominated.
- Potential ways to accelerate training: use higher order optimizers / seek parallel compute

HInt: Point A correspond to large training steps with small batch size, and Point B correspond to large batch size and smaller training steps

Point A: Regime: noise dominated. Potential way to accelerate training: seek parallel compute.
Point B: Regime: curvature dominated. Potential way to accelerate training: use higher order optimizers.

## 2.2

### Question

> **In order to prompt ChatGPT, I included the explaination of the figures from Kaplan et al. and combined them in my prompt.**

We have seen in the previous section that batch size is an important hyperparameter during training. Besides efficiently minimizing the training loss, we are also interested in the test loss. Recently, researchers have observed an intriguing relationship between the test loss and hyperparameters such as the model size, dataset size and the amount of compute used. We explore this relationship for neural language models in this section. The figures in this question are from Kaplan et al.

Test loss of language models of different sizes, plotted against the dataset size (tokens processed) and the amount of compute (in petaflop/s-days) shows that It is easy to see from figure 2 that larger models train faster, and are therefore more sample efficient.

Test loss for different sized models after the initial transient period, plotted against the number of training steps (Smin) when using the critical batch sizes (the batch sizes that separate the two regimes in Question 2.1.2) shows that when training a given model for a finite number of parameter update steps $S$ in the infinite data limit, after an initial transient period, the learning curves can be accurately fit by

$$L(N, S) = \left(\frac{N_c}{N}\right)^{\alpha_N} + \left(\frac{S_c}{S_{\min}(S)}\right)^{\alpha_S} \tag{18}$$

where $S_c \approx 2.1 \times 10^3$ and $\alpha_S \approx 0.76$, and $S_{\min}(S)$ is the minimum possible number of optimization steps (parameter updates) estimated using Equation (5.4).

When training within a fixed compute budget $C$, but with no other constraints, Equation (1.6) leads to the prediction that the optimal model size $N$, optimal batch size $B$, optimal number of steps $S$, and dataset size $D$ should grow as

$$N \propto C^{\alpha_C^{\min}/\alpha_N}, \quad B \propto C_C^{\alpha_C^{\min}/\alpha_B}, \quad S \propto C^{\alpha_C^{\min}/\alpha_S}, \quad D = B \cdot S \tag{19}$$

with

$$\alpha_C^{\min} = 1/\left(1/\alpha_S + 1/\alpha_B + 1/\alpha_N\right) \tag{20}$$

which closely matches the empirically optimal results $N \propto C_{\min}^{0.73}$, $B \propto C_{\min}^{0.24}$, and $S \propto C_{\min}^{0.03}$. As the computational budget $C$ increases, it should be spent primarily on larger models, without dramatic increases in training time or dataset size. This also implies that as models grow larger, they become increasingly sample efficient. In practice, researchers typically train smaller models for longer than would be maximally compute-efficient because of hardware constraints. Optimal performance depends on total compute as a power law.

Previously, you have trained a neural language model and obtained somewhat adequate performance. You have now secured more compute resources (in PF-days), and want to improve the model test performance (assume you will train from scratch). Which of the following is the best option? Give a brief explanation (2-3 sentences).

A. Train the same model with the same batch size for more steps.

B. Train the same model with a larger batch size (after tuning learning rate), for the same number of steps.

C. Increase the model size

## Solution

**C. Increase the model size is the best option.**

According to the equations provided and empirical results, as the computational budget $C$ increases, it should be primarily spent on larger models. Larger models are more sample efficient and tend to achieve better test performance. Increasing the model size will make better use of the additional compute resources and lead to improved test performance compared to simply increasing the batch size or training for more steps.

We have seen in the previous section that batch size is an important hyperparameter during training. Besides efficiently minimizing the training loss, we are also interested in the test loss. Recently, researchers have observed an intriguing relationship between the test loss and hyperparameters such as the model size, dataset size and the amount of compute used. We explore this relationship for neural language models in this section. The figures in this question are from Kaplan et al.

Test loss of language models of different sizes, plotted against the dataset size (tokens processed) and the amount of compute (in petaflop/s-days) shows that it is easy to see from figure 2 that larger models train faster, and are therefore more sample efficient.

Test loss for different sized models after the initial transient period, plotted against the number of training steps (Smin) when using the critical batch sizes (the batch sizes that separate the two regimes in Question 2.1.2) shows that when training a given model for a finite number of parameter update steps $S$ in the infinite data limit, after an initial transient period, the learning curves can be accurately fit by

$$
L(N, S)=\left(\frac{N_c}{N}\right)^{\alpha_N}+\left(\frac{S_c}{S_{\min }(S)}\right)^{\alpha_S}
$$

where $S_c \approx 2.1 \times 10^3$ and $\alpha_S \approx 0.76$, and $S_{\min }(S)$ is the minimum possible number of optimization steps (parameter updates) estimated using Equation (5.4).

When training within a fixed compute budget $C$, but with no other constraints, Equation (1.6) leads to the prediction that the optimal model size $N$, optimal batch size $B$, optimal number of steps $S$, and dataset size $D$ should grow as

$$
N \propto C^{\alpha_C^{\min } / \alpha_N}, \quad B \propto C_C^{\alpha_C^{\min } / \alpha_B}, \quad S \propto C^{\alpha_C^{\min } / \alpha_S}, \quad D=B \cdot S
$$

with

$$
\alpha_C^{\min }=1 /\left(1 / \alpha_S+1 / \alpha_B+1 / \alpha_N\right)
$$

which closely matches the empirically optimal results $N \propto C_{\min }^{0.73}, B \propto C_{\min }^{0.24}$, and $S \propto C_{\min }^{0.03}$. As the computational budget $C$ increases, it should be spent primarily on larger models, without dramatic increases in training time or dataset size (see Figure 3). This also implies that as models grow larger, they become increasingly sample efficient. In practice, researchers typically train smaller models for longer than would be maximally compute-efficient because of hardware constraints. Optimal performance depends on total compute as a power law.

Previously, you have trained a neural language model and obtained somewhat adequate performance. You have now secured more compute resources (in PF-days), and want to improve the model test performance (assume you will train from scratch). Which of the following is the best option? Give a brief explanation (2-3 sentences).

A. Train the same model with the same batch size for more steps.

B. Train the same model with a larger batch size (after tuning learning rate), for the same number of steps.

C. Increase the model size

C. Increase the model size is the best option. According to the equations provided and

empirical results, as the computational budget $C$ increases, it should be primarily spent on larger models. Larger models are more sample efficient and tend to achieve better test performance. Increasing the model size will make better use of the additional compute resources and lead to improved test performance compared to simply increasing the batch size or training for more steps.

## 3.1 Scaled Dot Product Attention

### 3.1.1 Implement the scaled dot-product attention mechanism.

```python
def forward(self, queries, keys, values):
    """The forward pass of the scaled dot attention mechanism.

    Arguments:
        queries: The current decoder hidden state, 2D or 3D tensor. (batch_size x (k) x hidden_size)
        keys: The encoder hidden states for each step of the input sequence. (batch_size x seq_len x hidden_size)
        values: The encoder hidden states for each step of the input sequence. (batch_size x seq_len x hidden_size)

    Returns:
        context: weighted average of the values (batch_size x k x hidden_size)
        attention_weights: Normalized attention weights for each encoder hidden state. (batch_size x seq_len x k)

        The output must be a softmax weighting over the seq_len annotations.
    """

    batch_size = queries.size(0)

    # Apply linear transformations to the queries, keys, and values
    q = self.Q(queries)
    k = self.K(keys)
    v = self.V(values)

    # Compute the unnormalized attention weights
    unnormalized_attention = torch.bmm(q, k.transpose(1, 2))

    # Apply the scaling factor
    unnormalized_attention = unnormalized_attention * self.scaling_factor

    # Normalize the attention weights using the softmax function
    attention_weights = self.softmax(unnormalized_attention)

    # Compute the context vector by taking the weighted sum of the value vectors
    context = torch.bmm(attention_weights, v)

    return context, attention_weights
```

we learnt about Scaled Dot-product Attention used in the transformer models. The function $f$ is a dot product between the linearly transformed query and keys using weight matrices $W_q$ and $W_k$ :

$$
\begin{aligned}
& \tilde{\alpha}_i^{(t)}=f\left(Q_t, K_i\right)=\frac{\left(W_q Q_t\right)^T\left(W_k K_i\right)}{\sqrt{d}}, \\

& \alpha_i^{(t)}=\operatorname{softmax}\left(\tilde{\alpha}^{(t)}\right)_i, \\
& c_t=\sum_{i=1}^T \alpha_i^{(t)} W_v V_i,
\end{aligned}
$$
where, $d$ is the dimension of the query and the $W_v$ denotes weight matrix project the value to produce the final context vectors.

```
class ScaledDotAttention(nn.Module):
  def __init__(self, hidden_size):
    super(ScaledDotAttention, self).__init__()

    self.hidden_size = hidden_size

    self.Q = nn.Linear(hidden_size, hidden_size)
    self.K = nn.Linear(hidden_size, hidden_size)
    self.V = nn.Linear(hidden_size, hidden_size)
    self.softmax = nn.Softmax(dim=2)
    self.scaling_factor = torch.rsqrt(
      torch.tensor(self.hidden_size, dtype=torch.float)
    )

  def forward(self, queries, keys, values):
    """The forward pass of the scaled dot attention mechanism.

    Arguments:
      queries: The current decoder hidden state, 2D or 3D tensor. (batch_size x (k) x
hidden_size)
      keys: The encoder hidden states for each step of the input sequence.
(batch_size x seq_len x hidden_size)
      values: The encoder hidden states for each step of the input sequence.
(batch_size x seq_len x hidden_size)

    Returns:
      context: weighted average of the values (batch_size x k x hidden_size)
      attention_weights: Normalized attention weights for each encoder hidden
state. (batch_size x seq_len x k)

      The output must be a softmax weighting over the seq_len annotations.
    """

    # ------------
    # FILL THIS IN
    # ------------
    # batch_size = ...
    # q = ...
    # k = ...
    # v = ...
    # unnormalized_attention = ...
    # attention_weights = ...
```

```
# context = ...
# return context, attention_weights
return context, attention_weights
```

To complete the forward function of the `ScaledDotAttention` class, you'll want to perform the following steps:

1. Obtain the batch size from the input tensor 'queries'.
2. Apply the linear transformations for the query, key, and value tensors.
3. Compute the unnormalized attention weights using the dot product between the transformed query and key tensors.
4. Apply the scaling factor to the unnormalized attention weights.
5. Normalize the attention weights using the softmax function.
6. Compute the context vector by taking the weighted sum of the value vectors.
7. Return the context vector and the attention weights.

Here's the completed forward function:

python Copy code

```python
def forward(self, queries, keys, values):
    # Get the batch size from the queries tensor
    batch_size = queries.size(0)

    # Apply linear transformations to the queries, keys, and values
    q = self.Q(queries)
    k = self.K(keys)
    v = self.V(values)

    # Compute the unnormalized attention weights
    unnormalized_attention = torch.bmm(q, k.transpose(1, 2))

    # Apply the scaling factor
    unnormalized_attention = unnormalized_attention * self.scaling_factor

    # Normalize the attention weights using the softmax function
    attention_weights = self.softmax(unnormalized_attention)

    # Compute the context vector by taking the weighted sum of the value vec
    context = torch.bmm(attention_weights, v)

    return context, attention_weights
```

### 3.1.2 Implement the causal scaled dot-product attention mechanism.

```python
def forward(self, queries, keys, values):
    """The forward pass of the scaled dot attention mechanism.

    Arguments:
        queries: The current decoder hidden state, 2D or 3D tensor. (batch_size x (k) x hidden_size)
        keys: The encoder hidden states for each step of the input sequence. (batch_size x seq_len x hidden_size)
        values: The encoder hidden states for each step of the input sequence. (batch_size x seq_len x hidden_size)

    Returns:
        context: weighted average of the values (batch_size x k x hidden_size)
        attention_weights: Normalized attention weights for each encoder hidden state. (batch_size x seq_len x k)

        The output must be a softmax weighting over the seq_len annotations.
    """

    # Get the batch size from the queries tensor
    batch_size = queries.size(0)

    # Apply linear transformations to the queries, keys, and values
    q = self.Q(queries)
    k = self.K(keys)
    v = self.V(values)

    # Compute the unnormalized attention weights
    unnormalized_attention = torch.bmm(q, k.transpose(1, 2))

    # Apply the scaling factor
    unnormalized_attention = unnormalized_attention * self.scaling_factor

    # Create the mask for causal attention (lower triangular matrix with ones)
    mask = torch.tril(torch.ones_like(unnormalized_attention))

    # Apply the mask to the unnormalized_attention tensor, adding negative infinity to masked elements
    unnormalized_attention = unnormalized_attention + (self.neg_inf * (1 - mask))

    # Normalize the attention weights using the softmax function
    attention_weights = self.softmax(unnormalized_attention)

    # Compute the context vector by taking the weighted sum of the value vectors
    context = torch.bmm(attention_weights, v)

    return context, attention_weights
```

DO

2. [0.5pt] Implement the causal scaled dot-product attention mechanism. Fill in the forward method in the CausalScaledDotAttention class. It will be mostly the same as the ScaledDotAttention class. The additional computation is to mask out the attention to the future time steps. You will need to add self.neg_inf to some of the entries in the unnormalized attention weights. You may find torch.tril or torch.triu handy for this part.

```python
class CausalScaledDotAttention(nn.Module):
    def __init__(self, hidden_size):
        super(CausalScaledDotAttention, self).__init__()

        self.hidden_size = hidden_size
        self.neg_inf = torch.tensor(-1e7)

        self.Q = nn.Linear(hidden_size, hidden_size)
        self.K = nn.Linear(hidden_size, hidden_size)
        self.V = nn.Linear(hidden_size, hidden_size)
```

```
        self.softmax = nn.Softmax(dim=2)
        self.scaling_factor = torch.rsqrt(
            torch.tensor(self.hidden_size, dtype=torch.float)
        )

    def forward(self, queries, keys, values):
        """The forward pass of the scaled dot attention mechanism.

        Arguments:
            queries: The current decoder hidden state, 2D or 3D tensor. (batch_size x (k) x
hidden_size)
            keys: The encoder hidden states for each step of the input sequence.
(batch_size x seq_len x hidden_size)
            values: The encoder hidden states for each step of the input sequence.
(batch_size x seq_len x hidden_size)

        Returns:
            context: weighted average of the values (batch_size x k x hidden_size)
            attention_weights: Normalized attention weights for each encoder hidden
state. (batch_size x seq_len x k)

            The output must be a softmax weighting over the seq_len annotations.
        """

        # ------------
        # FILL THIS IN
        # ------------
        # batch_size =
        # q =
        # k =
        # v =
        # unnormalized_attention =
        # mask =
        # attention_weights =
        # context =
```

To implement the causal scaled dot-product attention mechanism, you will need to add a mask to the unnormalized_attention tensor to prevent the model from attending to future time steps. You can use `torch.tril` to create a mask with lower triangular elements set to 1 and the rest set to 0. Then, add the neg_inf tensor (multiplied by the inverted mask) to the unnormalized_attention tensor to apply the mask. Here's the completed forward function:

python Copy code

```python
def forward(self, queries, keys, values):
    # Get the batch size from the queries tensor
```

```
batch_size = queries.size(0)

# Apply linear transformations to the queries, keys, and values
q = self.Q(queries)
k = self.K(keys)
v = self.V(values)

# Compute the unnormalized attention weights
unnormalized_attention = torch.bmm(q, k.transpose(1, 2))

# Apply the scaling factor
unnormalized_attention = unnormalized_attention * self.scaling_factor

# Create the mask for causal attention (lower triangular matrix with one
mask = torch.tril(torch.ones_like(unnormalized_attention))

# Apply the mask to the unnormalized_attention tensor, adding negative i
unnormalized_attention = unnormalized_attention + (self.neg_inf * (1 - m

# Normalize the attention weights using the softmax function
attention_weights = self.softmax(unnormalized_attention)

# Compute the context vector by taking the weighted sum of the value vec
context = torch.bmm(attention_weights, v)

return context, attention_weights
```

### 3.1.3

**Q1: Describe why we need to represent the position of each word through this positional encoding in one or two sentences.**

**A:** Positional encoding is necessary because the self-attention mechanism in the Transformer model is permutation invariant, meaning it does not inherently take word order into account. By adding positional encodings to the input embeddings, the model can learn and utilize the relative positions of words in the sequence for better understanding and generating appropriate output.

**Q2: Describe the advantages of using this positional encoding method, as opposed to other positional encoding methods such as a one hot encoding in one or two sentences.**

**A:** Sinusoidal encodings are continuous and dense, allowing the model to generalize to longer sequences by interpolating positional information, whereas one-hot encoding would require a fixed and potentially large vocabulary size for different positions. Additionally, sinusoidal encodings can capture relative positional relationships between words through simple arithmetic operations, which helps the model learn more effectively.

DO

The encoder consists of three components:
- Positional encoding: To encode the position of each word, we add to its embedding a constant vector that depends on its position:
pth word embedding $=$ input embedding + positional $\operatorname{encoding}(p)$
We follow the same positional encoding methodology described in Vaswani et al. [2017]. That is we use sine and cosine functions:
$$
\begin{aligned}
\mathrm{PE}(\operatorname{pos}, 2 i) & =\sin \frac{\text { pos }}{10000^{2 i / d_{\text {model }}}} \\
\mathrm{PE}(\operatorname{pos}, 2 i+1) & =\cos \frac{\mathrm{pos}}{10000^{2 i / d_{\text {model }}}}
\end{aligned}
$$
Since we always use the same positional encodings throughout the training, we pregenerate all those we'll need while constructing this class (before training) and keep reusing them throughout the training.
- A ScaledDotAttention operation.
- A following MLP.
For this question, describe:
(1) why we need to represent the position of each word through this positional encoding in one or two sentences.
(2) Additionally, describe the advantages of using this positional encoding method, as opposed to other positional encoding methods such as a one hot encoding in one or two sentences.

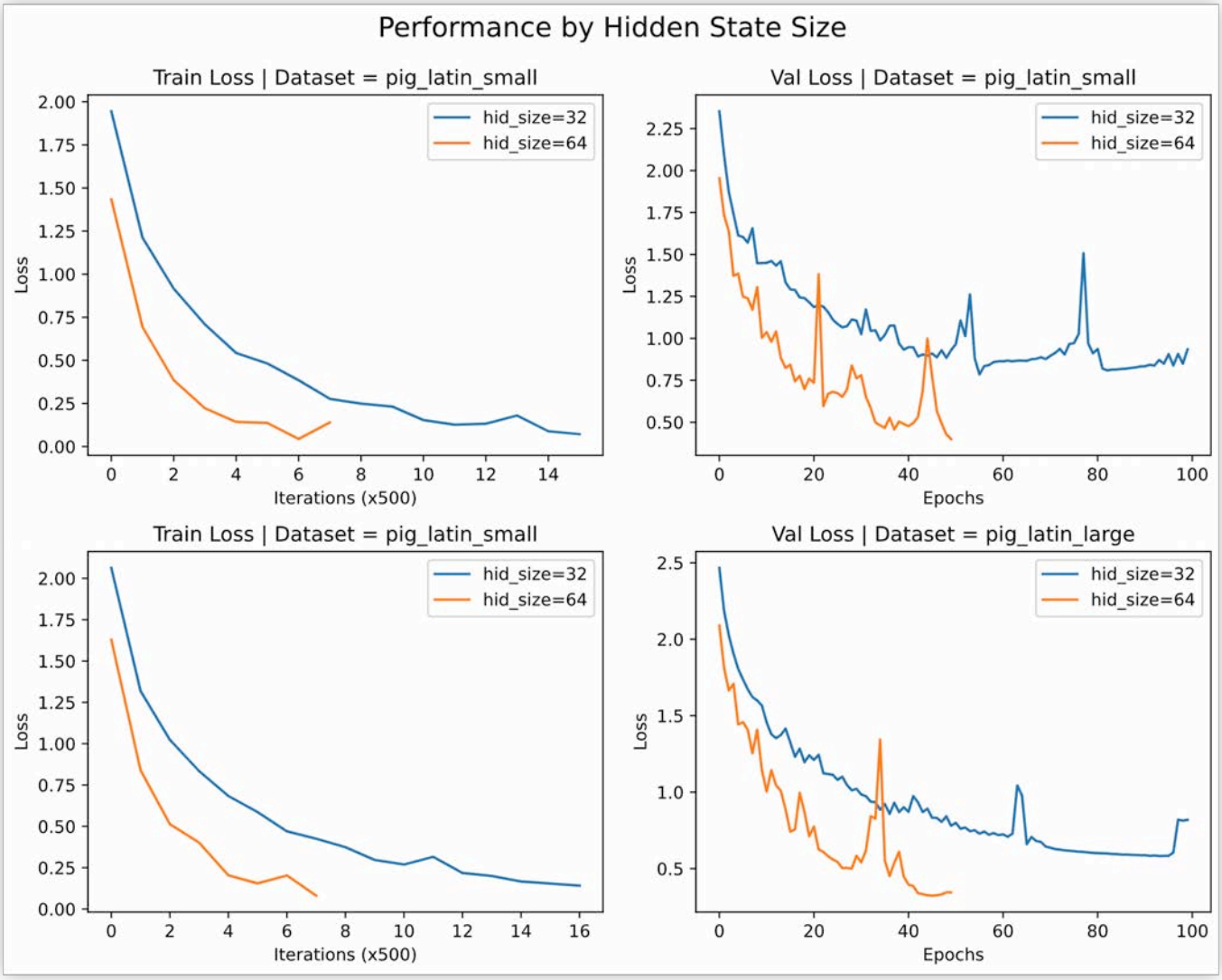
(1) Positional encoding is necessary because the self-attention mechanism in the Transformer model is permutation invariant, meaning it does not inherently take word order into account. By adding positional encodings to the input embeddings, the model can learn and utilize the relative positions of words in the sequence for better understanding and generating appropriate output.

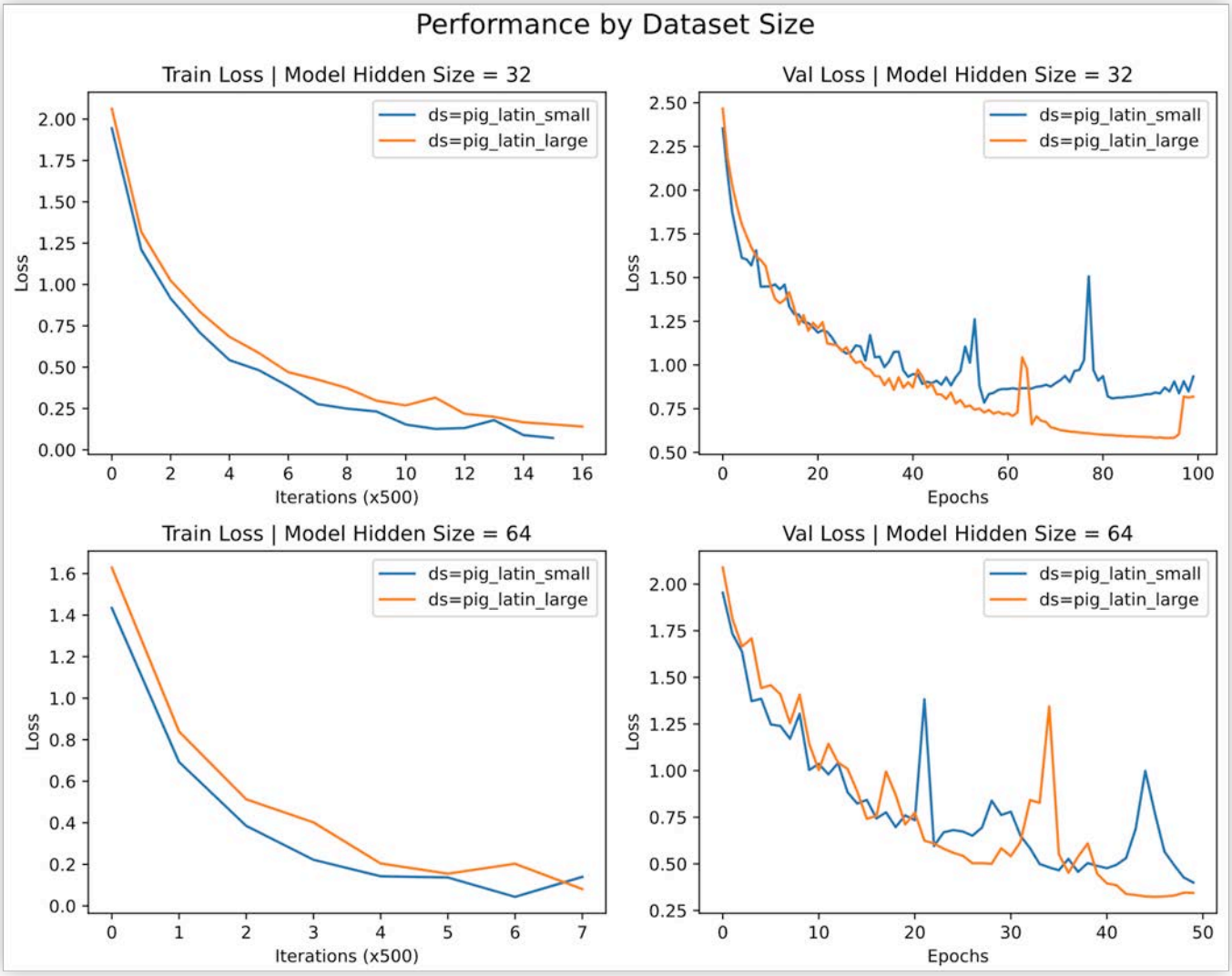
(2) The sinusoidal positional encoding method has several advantages over other methods, such as one-hot encoding. Sinusoidal encodings are continuous and dense, allowing the model to generalize to longer sequences by interpolating positional information, whereas one-hot encoding would require a fixed and potentially large vocabulary size for different positions. Additionally, sinusoidal encodings can capture relative positional relationships between words through simple arithmetic operations, which helps the model learn more effectively.

### 3.1.4

**Q: The two loss curves plots:**

**A:**






**Q: The lowest validation loss for each run:**

**A:**

| Dataset | Hidden Size | Lowest Validation Loss |
|---|---|---|
| pig_latin_small | 32 | 0.785 |
| pig_latin_large | 32 | 0.582 |
| pig_latin_small | 64 | 0.400 |
| pig_latin_large | 64 | 0.323 |

**Q: Report the effects of increasing model capacity via the hidden size, and the effects of increasing dataset size. In particular, report your observations on how loss as a function of gradient descent iterations is affected, and how changing model/dataset size affects the generalization of the model. Are these results what you would expect?**

**A:**

- Increasing the hidden size tends to decrease the lowest validation loss, indicating that increasing the model capacity can lead to better performance.
- Increasing the dataset size also tends to decrease the lowest validation loss, indicating that having more data can improve the generalization of the model.
- Looking at the loss as a function of gradient descent iterations, we can see that increasing the hidden size leads to faster convergence, with the loss reaching a lower value in fewer iterations. However, increasing the dataset size does not seem to have a significant effect on the speed of convergence.

Overall, these results are consistent with our expectations based on prior knowledge that increasing model capacity and dataset size can both lead to better performance, and larger batch size will lead to faster convergence.

# 3.2 Decoder Only NMT

## 3.2.1 Construct the input tensors and the target tensors for training a decoder

```python
def generate_tensors_for_training_decoder_nmt(src_EOP, tgt_EOS, start_token, cuda):

    # Step1: concatenate input_EOP, and target_EOS vectors to form a target tensor.
    src_EOP_tgt_EOS = torch.cat((src_EOP, tgt_EOS), dim=1)

    # Step2: make a sos vector
    sos_vector = torch.full((src_EOP.size(0), 1), start_token, dtype=torch.long)

    # Check if GPU is available and move the tensor to GPU if specified
    sos_vector = to_var(sos_vector, cuda)

    # Step3: make a concatenated input tensor to the decoder-only NMT (format: Start-of-token source end-of-prompt target)
    SOS_src_EOP_tgt = torch.cat((sos_vector, src_EOP, tgt_EOS), dim=1)

    return SOS_src_EOP_tgt, src_EOP_tgt_EOS
```

In this subsection, we will train a decoder-only NMT model using the CausalAttention mechanism. The key difference between this approach and the previous encoder-decoder approach is that we do not encode a hidden state of the input sequence first using an encoder. Instead, we feed both the input sequence and the target sequence to a decoder simultaneously,

Construct the input tensors and the target tensors for training a decoder. For this question, we ask you to implement the function generate_tensors_for_training_decoder_nmt that takes in an input sequence plus an end-of-prompt token and an output sequence plus an end-of-sentence token and returns two concatenated sequences. One has the form
〈SOS〉 input sequence 〈EOP〉 output sequence
, as in the input to the decoder shown in Figure 6, and the other has the form
input sequence <EOP> output sequence <EOS>

def generate_tensors_for_training_decoder_nmt(src_EOP, tgt_EOS, start_token,

cuda):

```
    # ------------
    # FILL THIS IN
    # ------------
    # Step1: concatenate input_EOP, and target_EOS vectors to form a target tensor.
    # src_EOP_tgt_EOS =
    # Step2: make a sos vector
    # sos_vector =
    # sos_vector = to_var(sos_vector, cuda)
    # Step3: make a concatenated input tensor to the decoder-only NMT (format: Start-of-token source end-of-prompt target)
    # SOS_src_EOP_tgt =
    return SOS_src_EOP_tgt, src_EOP_tgt_EOS
```

please finish the function above

Here's the completed function for generating tensors for training a decoder-only NMT model using the CausalAttention mechanism:

python Copy code

```python
import torch

def generate_tensors_for_training_decoder_nmt(src_EOP, tgt_EOS, start_token,
    # Step1: concatenate input_EOP and target_EOS vectors to form a target t
    src_EOP_tgt_EOS = torch.cat((src_EOP, tgt_EOS), dim=1)

    # Step2: make a start-of-sequence (SOS) vector
    sos_vector = torch.full((src_EOP.size(0), 1), start_token, dtype=torch.l
    if cuda:
        sos_vector = sos_vector.cuda()

    # Step3: make a concatenated input tensor to the decoder-only NMT (forma
    SOS_src_EOP_tgt = torch.cat((sos_vector, src_EOP, tgt_EOS), dim=1)

    return SOS_src_EOP_tgt, src_EOP_tgt_EOS
```

### 3.2.2 Implement the forward function in `DecoderOnlyTransformer`

```python
def forward(self, inputs):
    """Forward pass of the attention-based decoder RNN.

    Arguments:
        inputs: Input token indexes across a batch for all the time step. (batch_size x decoder_seq_len)
    Returns:
        output: Un-normalized scores for each token in the vocabulary, across a batch for all the decoding time steps. (batch_size x decoder_seq_len x vocab_size)
        attentions: The stacked attention weights applied to the encoder annotations (batch_size x encoder_seq_len x decoder_seq_len)
    """
    # Embed input tokens
    embedded_inputs = self.embedding(inputs)  # (batch_size x decoder_seq_len x hidden_size)

    # Add positional encodings
    embedded_inputs += self.positional_encodings[: embedded_inputs.size(1), :].unsqueeze(0)

    x = embedded_inputs

    self_attention_weights_list = []

    for i in range(self.num_layers):
        # Apply self-attention
        x, self_attention_weights = self.self_attentions[i](x, x, x)
        self_attention_weights_list.append(self_attention_weights)

        # Apply MLP
        x = x + self.attention_mlps[i](x)

    # Stack attention weights across all layers
    self_attention_weights = torch.stack(self_attention_weights_list, dim=1)  # (batch_size x num_layers x decoder_seq_len x decoder_seq_len)

    # Output layer
    output = self.out(x)  # (batch_size x decoder_seq_len x vocab_size)

    return output, self_attention_weights
```

```python
class TransformerDecoder(nn.Module):
    def __init__(self, vocab_size, hidden_size, num_layers, loss_seq_span):
        super(TransformerDecoder, self).__init__()
        self.vocab_size = vocab_size
        self.hidden_size = hidden_size

        self.embedding = nn.Embedding(vocab_size, hidden_size)
        self.num_layers = num_layers

        self.self_attentions = nn.ModuleList(
            [
                CausalScaledDotAttention(
                    hidden_size=hidden_size,
                )
                for i in range(self.num_layers)
            ]
        )

        self.attention_mlps = nn.ModuleList(
            [
                nn.Sequential(
                    nn.Linear(hidden_size, hidden_size),
                    nn.ReLU(),
                )
                for i in range(self.num_layers)
            ]
```

```
        )
        self.out = nn.Linear(hidden_size, vocab_size)

        self.positional_encodings = self.create_positional_encodings()
        self.loss_seq_span = loss_seq_span

    def forward(self, inputs):
        """Forward pass of the attention-based decoder RNN.

        Arguments:
            inputs: Input token indexes across a batch for all the time step. (batch_size x decoder_seq_len)
        Returns:
            output: Un-normalized scores for each token in the vocabulary, across a batch for all the decoding time steps. (batch_size x decoder_seq_len x vocab_size)
            attentions: The stacked attention weights applied to the encoder annotations (batch_size x encoder_seq_len x decoder_seq_len)
        """
        # ------------
        # FILL THIS IN
        # ------------
        return output, self_attention_weights
```

please Implement the forward function in DecoderOnlyTransformer.

Here's the forward function implemented in the DecoderOnlyTransformer class:

python Copy code

```python
class TransformerDecoder(nn.Module):
    def __init__(self, vocab_size, hidden_size, num_layers, loss_seq_span):
        super(TransformerDecoder, self).__init__()
        self.vocab_size = vocab_size
        self.hidden_size = hidden_size

        self.embedding = nn.Embedding(vocab_size, hidden_size)
        self.num_layers = num_layers

        self.self_attentions = nn.ModuleList(
            [
                CausalScaledDotAttention(
                    hidden_size=hidden_size,
                )
                for i in range(self.num_layers)
            ]
        )
```

```
        self.attention_mlps = nn.ModuleList(
            [
                nn.Sequential(
                    nn.Linear(hidden_size, hidden_size),
                    nn.ReLU(),
                )
                for i in range(self.num_layers)
            ]
        )
        self.out = nn.Linear(hidden_size, vocab_size)

        self.positional_encodings = self.create_positional_encodings()
        self.loss_seq_span = loss_seq_span

    def forward(self, inputs):
        """Forward pass of the attention-based decoder RNN.

        Arguments:
            inputs: Input token indexes across a batch for all the time step
        Returns:
            output: Un-normalized scores for each token in the vocabulary, a
            attentions: The stacked attention weights applied to the encoder
        """
        # Embed input tokens
        embedded_inputs = self.embedding(inputs)  # (batch_size x decoder_se

        # Add positional
```

DO

continue

encodings

embedded_inputs += self.positional_encodings[: embedded_inputs.size(1), :].unsqueeze(0)

python Copy code

```
    x = embedded_inputs
    self_attention_weights_list = []

    for i in range(self.num_layers):
        # Apply self-attention
```

```
        x, self_attention_weights = self.self_attentions[i](x, x, x)
        self_attention_weights_list.append(self_attention_weights)

        # Apply MLP
        x = x + self.attention_mlps[i](x)

    # Stack attention weights across all layers
    self_attention_weights = torch.stack(self_attention_weights_list, dim=1)

    # Output layer
    output = self.out(x)  # (batch_size x decoder_seq_len x vocab_size)

    return output, self_attention_weights
```

sql Copy code

```
In the forward function, we first embed the input tokens and add the positic
```

### 3.2.3 Train the model

**Q: Comment on the pros and cons of the decoder only approach.**

**A:**

Pros:

1. Simplicity: The decoder-only approach has a simpler architecture compared to the encoder-decoder approach, as it only consists of a single decoder model. This can make the model easier to understand, implement, and maintain.
2. Lower memory requirements: The absence of an encoder reduces the memory requirements of the model, which can be advantageous when training on limited hardware resources.
3. Faster inference: With only one network (decoder) to process the input, the decoder-only approach can potentially provide faster inference compared to the encoder-decoder approach.

Cons:

1. Limited context: The decoder-only approach does not explicitly model the source and target sequences separately. This can limit the model's ability to capture the context of the input sequence, potentially leading to suboptimal translations.
2. Less expressive: Since the decoder-only model relies solely on the self-attention mechanism, it might struggle to capture long-range dependencies and complex relationships between source and target sequences compared to the encoder-decoder approach, which has a dedicated encoder to process the input sequence.

3. May require larger models: To compensate for the lack of an encoder, decoder-only models might need to be larger to achieve comparable performance to encoder-decoder models, which can increase training time and computational

DO

Comment on the pros and cons of the decoder only approach compared to encoder-decoder approach

The decoder-only approach and the encoder-decoder approach are two different methods for training neural machine translation (NMT) models. Each has its own advantages and disadvantages, which are outlined below:

Decoder-only Approach:
Pros:

1. Simplicity: The decoder-only approach has a simpler architecture compared to the encoder-decoder approach, as it only consists of a single decoder model. This can make the model easier to understand, implement, and maintain.
2. Lower memory requirements: The absence of an encoder reduces the memory requirements of the model, which can be advantageous when training on limited hardware resources.
3. Faster inference: With only one network (decoder) to process the input, the decoder-only approach can potentially provide faster inference compared to the encoder-decoder approach.

Cons:

1. Limited context: The decoder-only approach does not explicitly model the source and target sequences separately. This can limit the model's ability to capture the context of the input sequence, potentially leading to suboptimal translations.
2. Less expressive: Since the decoder-only model relies solely on the self-attention mechanism, it might struggle to capture long-range dependencies and complex relationships between source and target sequences compared to the encoder-decoder approach, which has a dedicated encoder to process the input sequence.
3. May require larger models: To compensate for the lack of an encoder, decoder-only models might need to be larger to achieve comparable performance to encoder-decoder models, which can increase training time and computational

**Q: How is the quality of your generated results compared to the ones using the encoder-decoder model?**

**A:** Although the train and val loss converges, the generated results (as is shown below) are poor compared to the ones using encoder-decoder model, this is possibly because of the model isn't large enought and number of training epochs needs to be increased.

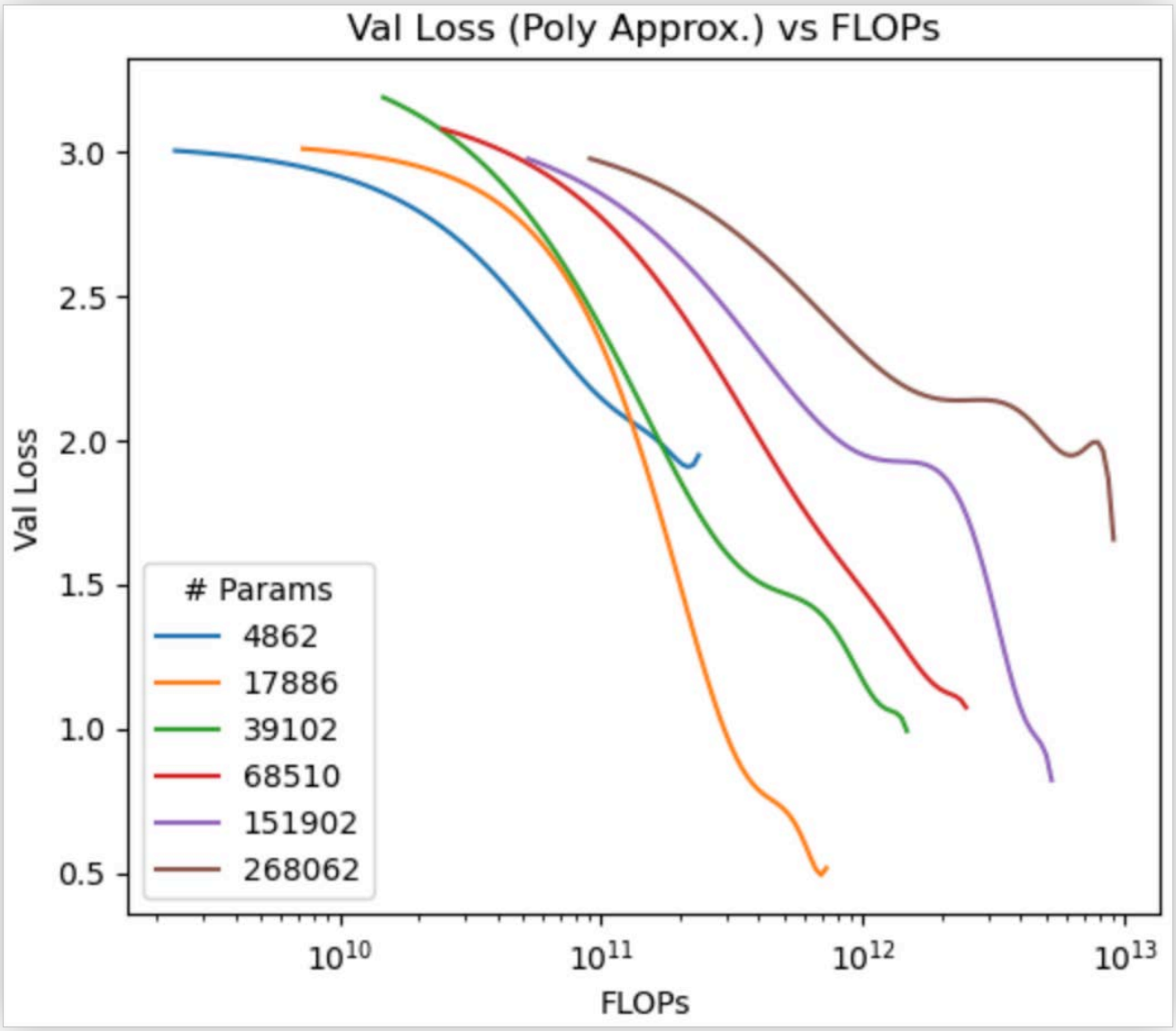
```
================================================================================
                                    Opts
--------------------------------------------------------------------------------
                        data_file_name: pig_latin_small
                                  cuda: 1
                               nepochs: 100
                        checkpoint_dir: checkpoints
                         learning_rate: 0.0005
                 early_stopping_patience: 100
                                lr_decay: 0.99
                             batch_size: 64
                            hidden_size: 128
                   num_transformer_layers: 4
                            loss_seq_span: target_only
                               num_layers: 4
                               vocab_size: 30
================================================================================
================================================================================
                                  Data Stats
--------------------------------------------------------------------------------
('small', 'allsmay')
('irreconcilable', 'irreconcilableway')
('dismounted', 'ismountedday')
('announce', 'announceway')
('merriment', 'errimentmay')
...
Epoch:  98 | Flops (T): 8.976621393432, Tokens (M) 5.186808, Train Loss 1.114, Val Loss 1.202 | Gen: hhhhhhhhhhhahhhhheee aaaaaaaaaaaaaaaaaaa oooooooooooooooooooo aaaaaaaaaaaaaaaaaaa aaaaaaa
Epoch:  99 | Flops (T): 9.0672943368, Tokens (M) 5.2392, Train Loss 1.123, Val Loss 1.154 | Gen: hhhhhhhhhheeeeeeeeee aaaaaaaaaaaaaaaaaa nnnnnnnnnnnnnnnnnnn ssssssssssssssssss aaaaaaaaaaa
Obtained lowest validation loss of: 1.145613039794721
Avg tokens per example (Seq Len): 20.481626270523847, Avg flops per example: 35446811.32447224
```

# 3.3 Scaling Law and IsoFLOP Profiles

## 3.3.1

**Q: Train six decoder-only translation models using the code provided and plot the validation loss as the function of FLOPs. Comment on any interesting thing you observe. Does larger model always have a smaller validation loss?**

**A:** We can observe that the the second smallest model achieved the lowest val loss while the smallest model got the highest val loss. Larger model don''t always provide better validation loss and the largest model's performance is even terrible although costing the highest FLOPs. So the optimal model size should be moderate to maximize performance for a given computational budget.

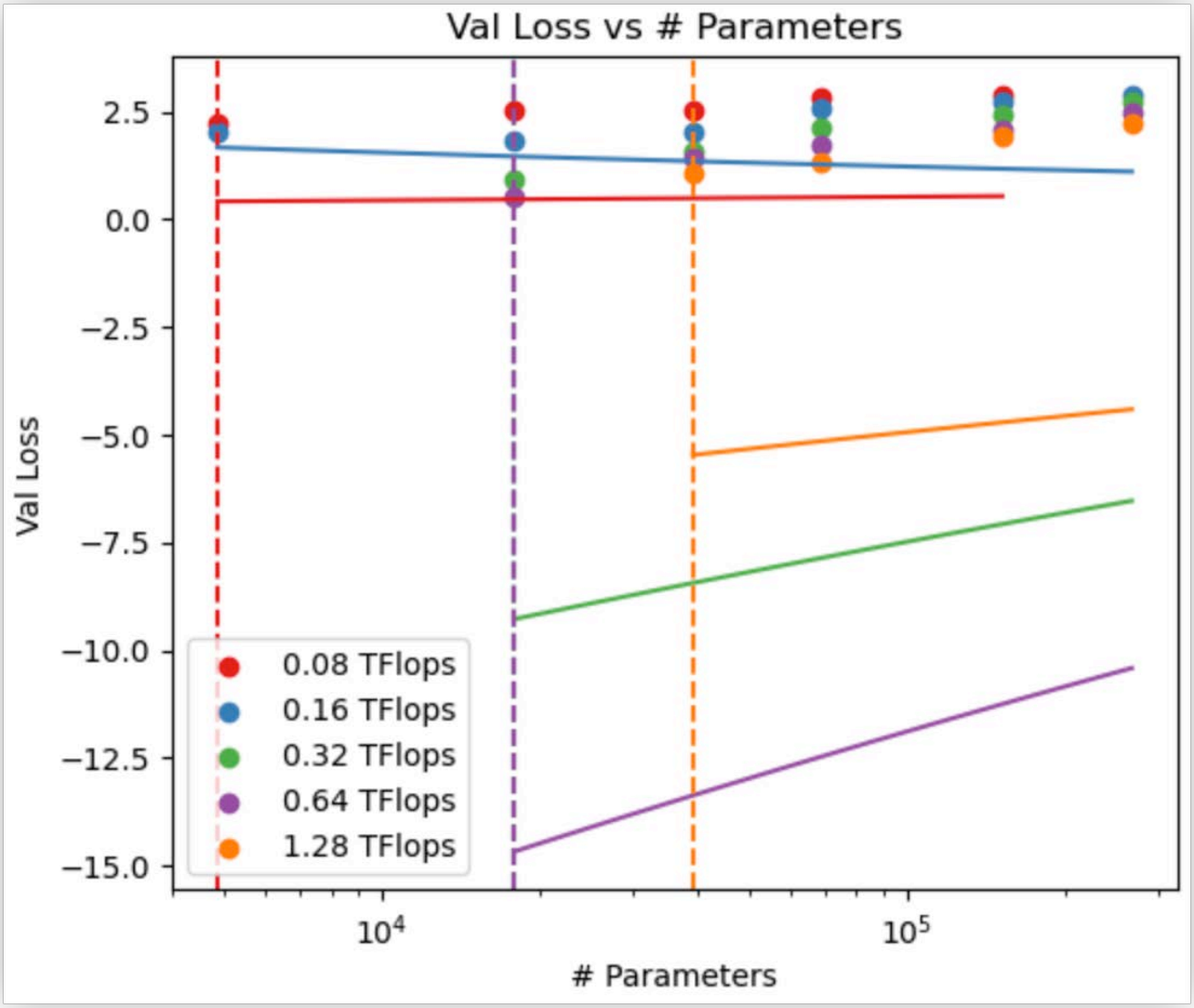



### 3.3.2

**Q: Find the optimal number of parameters using the quadratic function.**

**A:**

```
def find_optimal_params(x, y):
    # Fit a quadratic function to the validation loss and number of parameters in the log space
    p = np.polyfit(np.log(x), np.log(y), 2)

    # Find the optimal number of parameters using the quadratic function
    optimal_params_index = np.argmin(y)
    optimal_params = x[optimal_params_index]

    return p, optimal_params
```




This section will give you hands-on experience charting scaling law curves to forecast neural network performance. Scaling law is a fundamental concept that describes how the performance of a neural network changes with its size. Specifically, it relates the number of parameters or computations required by a neural network to achieve a certain level of performance, such as accuracy or loss. The scaling law provides a useful tool for predicting the performance of neural networks as they are scaled up or down. IsoFLOP is a method proposed in the"Training Compute-Optimal Large Language Models"paper[Hoffmann et al., 2022]to study the scaling law of large language models. The authors of the paper used IsoFLOP to study the effect of model size on the performance of large language models and to determine the optimal model size that maximizes performance for a given computational budget. The motivation for using IsoFLOP to forecast neural network performance is twofold. Firstly, it provides a more accurate and efficient way to explore the scaling law of large language models than traditional methods, which involve training multiple models at different sizes. Secondly, IsoFLOP allows for a better understanding of the trade-off between model size and training cost, which is crucial for designing large-scale neural network architectures that are both efficient and effective. By17CSC413/2516Winter2023with Professor Jimmy Ba&Professor Bo Wang Assignment1leveraging IsoFLOP, researchers

can gain insights into the scaling properties of neural networks, such as their accuracy and computational efficiency, and optimize their performance for specific applications and computational resources. In this question, we will plot the scaling law curve for the decoder-only translation models from the previous section. The notebook provided trains six translation models with different model sizes and varies the FLOP counts by training for different numbers of epochs. You are asked to complete the functions to make the final IsoFLOP curve consisting of models ranging from0.08TFLOPs to1.28TFLOPs.

For a given FLOPs, fit a quadratic function to the validation loss and number of parameters in the log space. Find the optimal number of parameters using the quadratic function. Specifically, you need to fill the "find optimal params" function.

```
def get_scatter_data(data_list, num_params, f_list, tflops):
  x, y = [], []
  for i, f in enumerate(f_list):
    if tflops <= max(data_list[i][3]) and tflops >= min(data_list[i][3]):
      x.append(num_params[i])
      y.append(np.polyval(f, tflops))
  return x, y


def find_optimal_params(x, y):
  # ------------
  # FILL THIS IN
  # ------------
  # p = np.polyfit( , , 2)
  # optimal_params =
  return p, optimal_params
```

Here's the completed find_optimal_params function:

python Copy code

```python
import numpy as np

def get_scatter_data(data_list, num_params, f_list, tflops):
    x, y = [], []
    for i, f in enumerate(f_list):
        if tflops <= max(data_list[i][3]) and tflops >= min(data_list[i][3])
            x.append(num_params[i])
            y.append(np.polyval(f, tflops))
    return x, y

def find_optimal_params(x, y):
    # Fit a quadratic function to the validation loss and number of paramete
    p = np.polyfit(np.log(x), np.log(y), 2)
```

```
    # Find the optimal number of parameters using the quadratic function
    optimal_params_index = np.argmin(y)
    optimal_params = x[optimal_params_index]

    return p, optimal_params
```

In the find_optimal_params function, we first fit a quadratic function to the validation loss and number of parameters in the log space using np.polyfit. Then, we find the optimal number of parameters using the quadratic function by finding the index with the minimum loss, and then return the corresponding number of parameters.

### 3.3.3

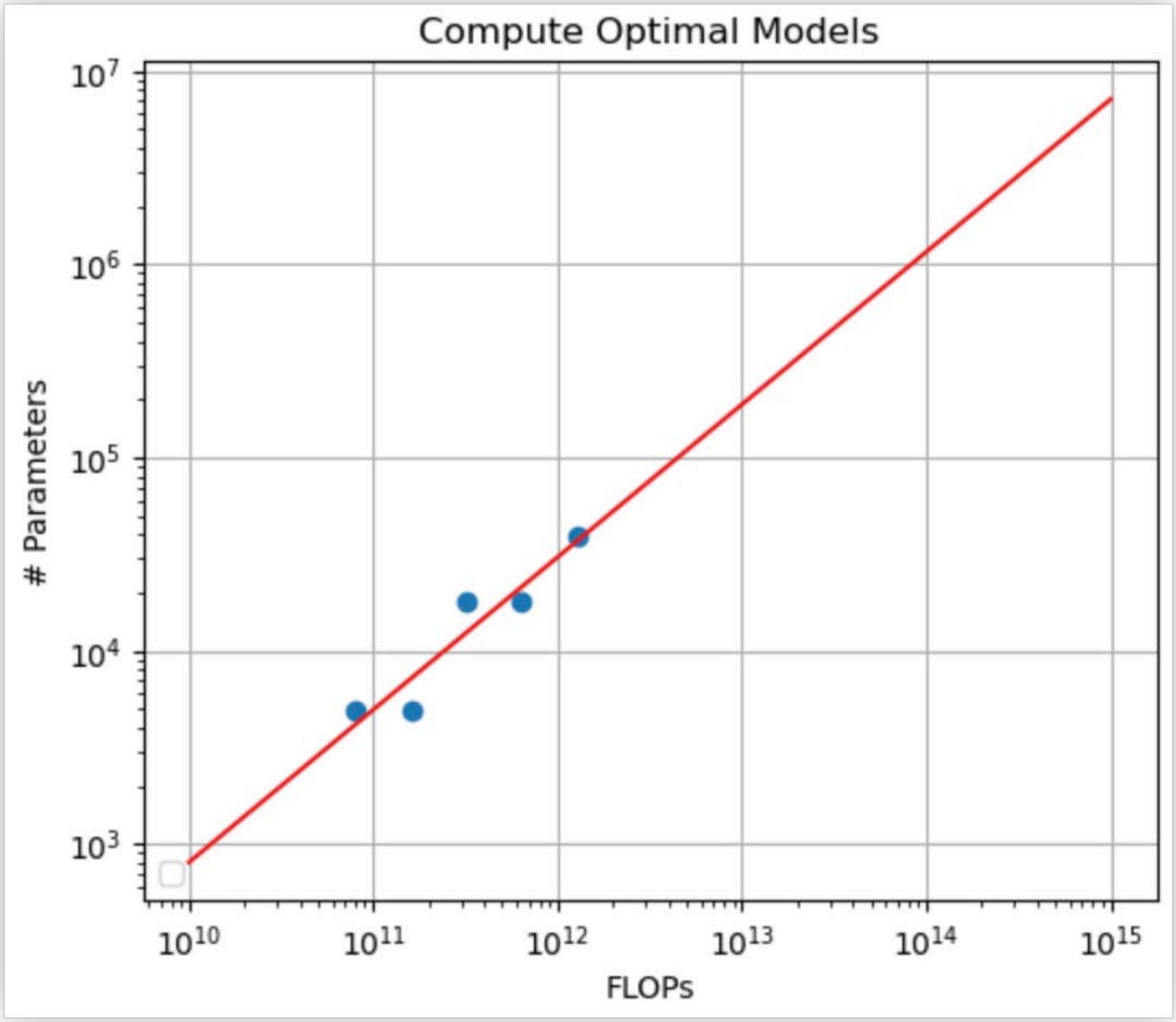
**Q: Compute Optimal Model plot by fitting a linear line to the target FLOPs and the optimal model parameters. Based on the plot, estimate the optimal number of parameters when we have a compute budget of 1e15.**

**A:** The fitted function is $y = 0.7894450187301238x - 4.988512273482354$. So when we have a compute budget of $1e15$

$$y = 10^{(0.7894450187301238 * log_{10}x - 4.988512273482354)} = 7131206.415822954. \tag{21}$$

The optimal number parameters will be around 7.13M.

```
def fit_linear_log(x, y):
    # Fit a linear line to the target FLOPs and the optimal model parameters in the log space
    m, c = np.polyfit(np.log10(x), np.log10(y), 1)
    return m, c
```


Compute Optimal Models
# Parameters
FLOPs
$10^7$
$10^6$
$10^5$
$10^4$
$10^3$
$10^{10}$
$10^{11}$
$10^{12}$
$10^{13}$
$10^{14}$
$10^{15}$

DO

Complete the Compute Optimal Model plot by fitting a linear line to the target FLOPs and the optimal model parameters. Based on the plot, estimate the optimal number of parameters when we have a compute budget of 1e15.

```
def fit_linear_log(x, y):
    # ------------
    # FILL THIS IN
    # ------------
    # m, c = np.polyfit(, , 1)
    return m, c
```

Here's the completed fit_linear_log function:

python Copy code

```python
def fit_linear_log(x, y):
    # Fit a linear line to the target FLOPs and the optimal model parameters
    m, c = np.polyfit(np.log(x), np.log(y), 1)
    return m, c
```

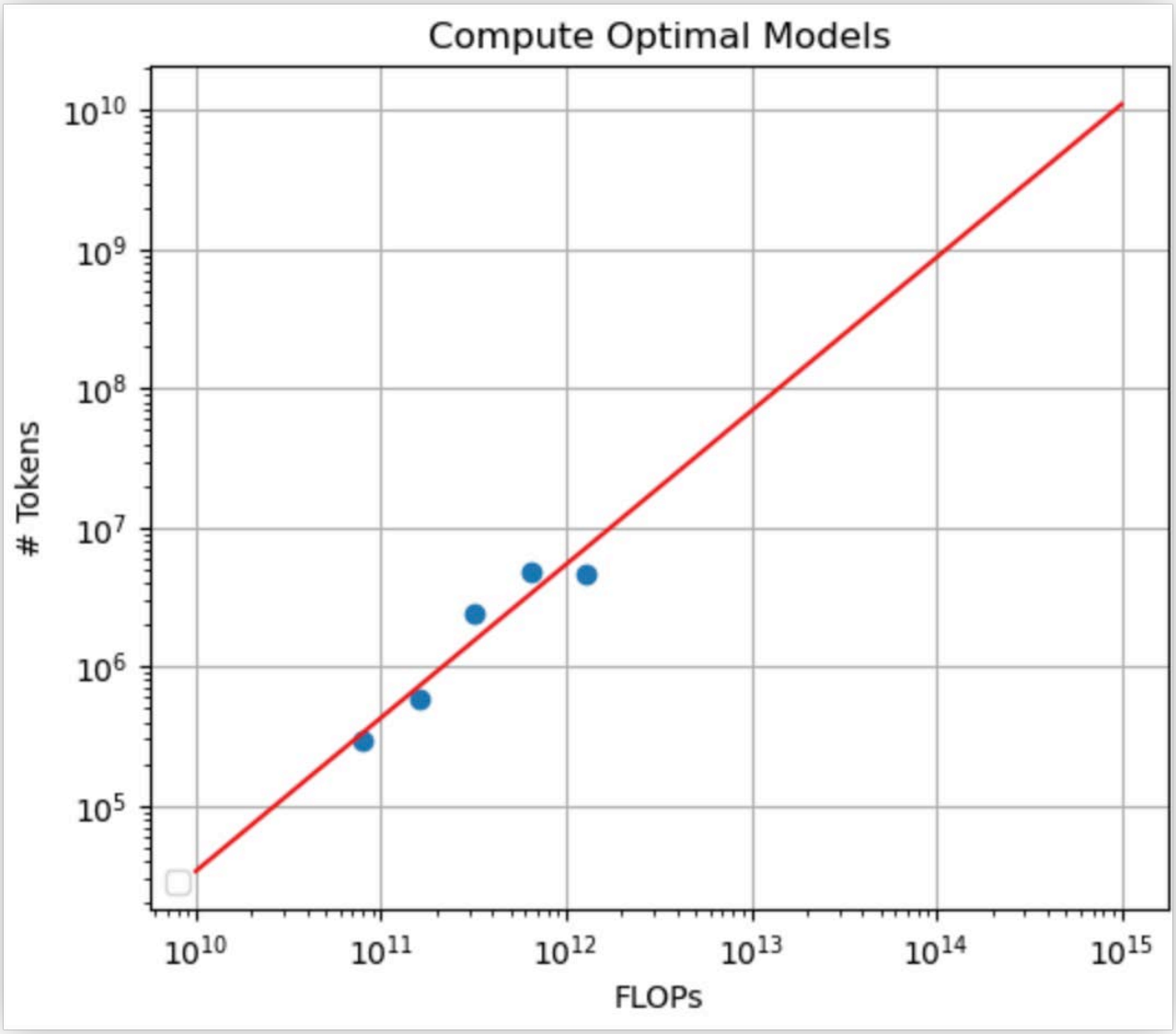
> I changed the `np.log` from ChatGPT to `np.log10` based on the codes for creating the figure.

### 3.3.4

**Q: Plot Compute Optimal Token using the code provided. Now, given the Compute Optimal Model plot and Compute Optimal Token plot, is the training setup in Section 3.2.3 compute optimal? If not, how should we change it?**

**A:** The training setup in Section 3.2.3 is not compute optimal, the number of parameters is about 268K, the FLOPs is around 9.1T and the number of tokens is about 5.2M.

- Based the Compute Optimal Model plot, the optimal model parameters should be around 173K, so increasing the FLOPs (train for more epochs) or reducing the hidden size will make it compute optimal.
- Based on the Compute Optimal Token plot, the optimal #Tokens is about 8.8M, so increasing the #Tokens (using larger dataset of train for more epochs) or reducing the FLOPs (reducing the hidden size) will make it compute optimal.




## 4.1 Add a classifier to BERT

**The `__init__` function:**

```python
class BertForSentenceClassification(BertModel):
    def __init__(self, config):
        super().__init__(config)

        ##### START YOUR CODE HERE #####
        # Add a linear classifier that map BERTs [CLS] token representation to the unnormalized
        # output probabilities for each class (logits).
        # Notes:
        #  * See the documentation for torch.nn.Linear
        #  * You do not need to add a softmax, as this is included in the loss function
        #  * The size of BERTs token representation can be accessed at config.hidden_size
        #  * The number of output classes can be accessed at config.num_labels
        self.classifier = nn.Linear(config.hidden_size, config.num_labels)
        ##### END YOUR CODE HERE #####
        self.loss = torch.nn.CrossEntropyLoss()
```

**The `forward` function:**

```python
    def forward(self, labels=None, **kwargs):
        outputs = super().forward(**kwargs)
        ##### START YOUR CODE HERE #####
        # Pass BERTs [CLS] token representation to this new classifier to produce the logits.
        # Notes:
        #  * The [CLS] token representation can be accessed at outputs.pooler_output
        cls_token_repr = outputs.pooler_output
        logits = self.classifier(cls_token_repr)
        ##### END YOUR CODE HERE #####
        if labels is not None:
            outputs = (logits, self.loss(logits, labels))
        else:
            outputs = (logits,)
        return outputs
```

### Question 1: Add a classifier to BERT [1pts]

Here, we will add a simple classifier to the [BertModel](https://huggingface.co/docs/transformers/v4.16.2/en/model_doc/bert#transformers.BertModel) provided by the Transformers library.

Your tasks are:

1. In `__init__`, add a linear classifier that will map BERTs `[CLS]` token representation to the unnormalized output probabilities for each class (`logits`).
2. In `forward`, pass BERTs `[CLS]` token representation to this new classifier to produce the `logits`.

In total, you won't have to write more than three new lines of code. See the comments in the code for help!

from transformers import BertModel

```
import torch.nn as nn

class BertForSentenceClassification(BertModel):
  def __init__(self, config):
    super().__init__(config)

    ##### START YOUR CODE HERE #####
    # Add a linear classifier that map BERTs [CLS] token representation to the
unnormalized
    # output probabilities for each class (logits).
    # Notes:
    #  * See the documentation for torch.nn.Linear
    #  * You do not need to add a softmax, as this is included in the loss function
    #  * The size of BERTs token representation can be accessed at
config.hidden_size
    #  * The number of output classes can be accessed at config.num_labels
    self.classifier = ...
    ##### END YOUR CODE HERE #####
    self.loss = torch.nn.CrossEntropyLoss()

  def forward(self, labels=None, **kwargs):
    outputs = super().forward(**kwargs)
    ##### START YOUR CODE HERE #####
    # Pass BERTs [CLS] token representation to this new classifier to produce the
logits.
    # Notes:
    #  * The [CLS] token representation can be accessed at outputs.pooler_output
    cls_token_repr = ...
    logits = ...
    ##### END YOUR CODE HERE #####
    if labels is not None:
      outputs = (logits, self.loss(logits, labels))
    else:
      outputs = (logits,)
    return outputs
```

from transformers import BertModel
import torch.nn as nn

class BertForSentenceClassification(BertModel):
def **init**(self, config):
super().**init**(config)

python Copy code

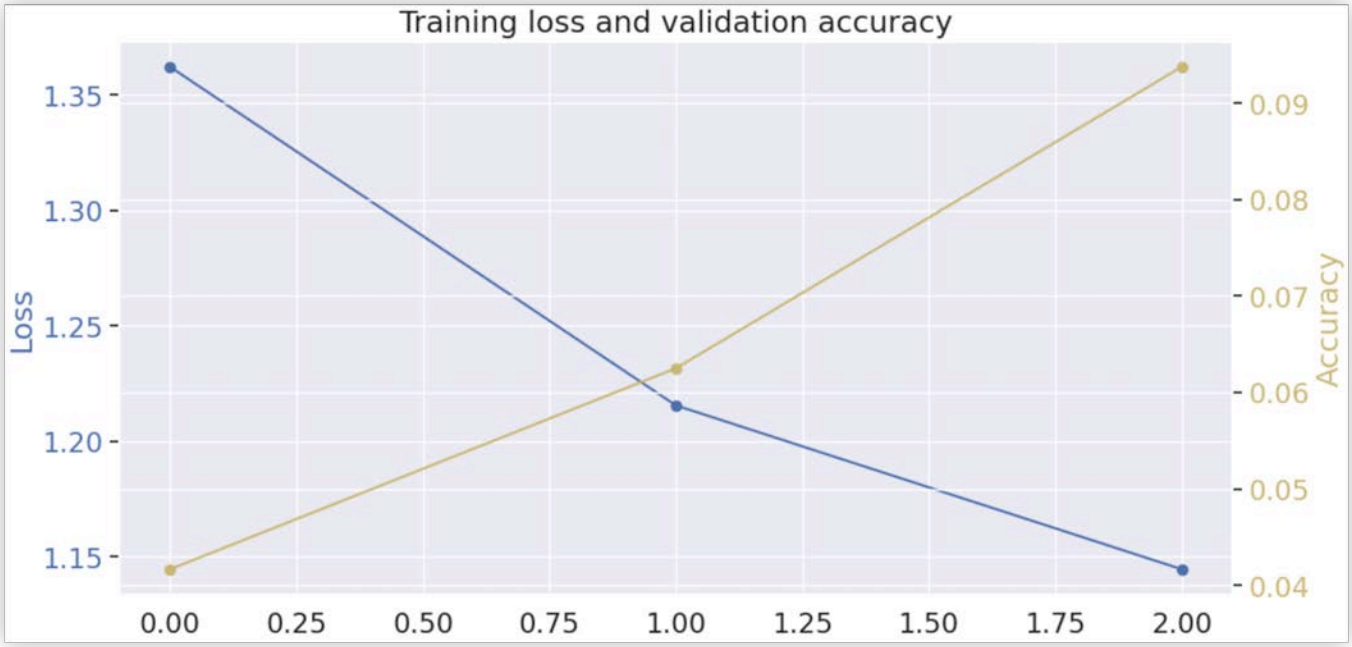
```
    ##### START YOUR CODE HERE #####
    # Add a linear classifier that map BERTs [CLS] token representation to t
    # output probabilities for each class (logits).
    # Notes:
```

```
    #  * See the documentation for torch.nn.Linear
    #  * You do not need to add a softmax, as this is included in the loss f
    #  * The size of BERTs token representation can be accessed at config.hi
    #  * The number of output classes can be accessed at config.num_labels
    self.classifier = nn.Linear(config.hidden_size, config.num_labels)
    ##### END YOUR CODE HERE #####
    self.loss = torch.nn.CrossEntropyLoss()

def forward(self, labels=None, **kwargs):
    outputs = super().forward(**kwargs)
    ##### START YOUR CODE HERE #####
    # Pass BERTs [CLS] token representation to this new classifier to produc
    # Notes:
    #  * The [CLS] token representation can be accessed at outputs.pooler_ou
    cls_token_repr = outputs.pooler_output
    logits = self.classifier(cls_token_repr)
    ##### END YOUR CODE HERE #####
    if labels is not None:
        outputs = (logits, self.loss(logits, labels))
    else:
        outputs =
```

## 4.3 Freezing the pretrained weights

**Fine-tuning:**

**Frozen:**




**Q: What is the effect on train time when BERTs weights are frozen? Why?**

**A:** The training time is much shorten (17 batch/s => 97 batch/s). Because when BERTs weights are frozen, we only need to compute the gradients for the final linear layer, which has much fewer parameters.

**Q: Compared to fine-tuning, what is the effect on performance (i.e. validation accuracy) when BERTs weights are frozen? Why?**

**A:** Freezing the pretrained weights can significantly reduce the performance of the model compared to fine-tuning all the weights, as the model is not able to adapt to the specific task and data. This is because the frozen weights are already optimized for a different task, and not allowing them to be updated prevents the model from learning task-specific features.

## 4.4 Effect of pretraining data

**Using BERTweet:**

**Q: Compared to fine-tuning BERT with the pretrained weights from MathBERT (see Question 2), what is the effect on performance (i.e. validation accuracy) when we fine-tune BERT with the pretrained weights from BERTweet? Why might this be the case?**

**A:** Using pretrained weights from BERTweet will result in lower performance compared to MathBERT due to the differences in domain and tokenization. BERTweet was trained on Twitter data, while MathBERT was trained on mathematical text, which can affect the quality of the learned representations. Additionally, the different vocabulary and tokenization scheme of BERTweet can impact its ability to represent the input sentences effectively.

## 5.2 Prompting CLIP

**Q: Come up with a caption that will "prompt" CLIP to select the following target image:**

**A:** The prompt I came out is **"several organge and white fish in coral"**.

**Q: Comment on the process of finding the caption: was it easy, or were there any difficulties?**

**A:** The search process is easy and it took only 2 tries for me to find the right caption (added the description of colors in my second try).

[17]

```python
caption = "several orange and white fish in coral"
```

Python

Below, we will display the image that produces the highest probability under the model given your caption. You should continue to tweak your caption until the target image (see above) is produced!

[18]

```python
original_images = []
images = []

for img_path in img_paths:

    image = Image.open(img_path).convert("RGB")
    original_images.append(image)
    images.append(preprocess(image))

image_input = torch.tensor(np.stack(images)).cuda()
with torch.no_grad():
    image_features = model.encode_image(image_input).float()
image_features /= image_features.norm(dim=-1, keepdim=True)

# Delete these to avoid consuming up too much memory
del images
del image_input
torch.cuda.empty_cache()

text_tokens = clip.tokenize(caption).cuda()

with torch.no_grad():
    text_features = model.encode_text(text_tokens).float()
    text_features /= text_features.norm(dim=-1, keepdim=True)

text_probs = (100.0 * image_features @ text_features.T).softmax(dim=0).cpu().detach().numpy()
highest_prob = np.argmax(text_probs)
plt.axis('off')
plt.imshow(original_images[highest_prob])

print(img_paths[highest_prob])
```

Python

```
./IMagenet/tiny-imagenet-200/train/n09256479/images/n09256479_452.JPEG
```

# nmt

March 27, 2023

# 1 Colab FAQ

For some basic overview and features offered in Colab notebooks, check out: Overview of Colaboratory Features

You need to use the colab GPU for this assignment by selecting:

**Runtime** → **Change runtime type** → **Hardware Accelerator: GPU**

# 2 Setup PyTorch

All files will be stored at /content/csc421/a3/ folder

# 3 Helper code

## 3.1 Utility functions

[ ]:
```
%matplotlib inline

import os
import pdb
import argparse
import pickle as pkl
from pathlib import Path

from collections import defaultdict

import numpy as np
import matplotlib as mpl
import matplotlib.pyplot as plt
import matplotlib.ticker as ticker

import torch
import torch.nn as nn
import torch.optim as optim
import torch.nn.functional as F
from torch.autograd import Variable
```

```python
from six.moves.urllib.request import urlretrieve
import tarfile
import pickle
import sys

import pandas as pd
import seaborn as sns


def get_file(
    fname, origin, untar=False, extract=False, archive_format="auto",␣
 ↪cache_dir="data"
):
    datadir = os.path.join(cache_dir)
    if not os.path.exists(datadir):
        os.makedirs(datadir)

    if untar:
        untar_fpath = os.path.join(datadir, fname)
        fpath = untar_fpath + ".tar.gz"
    else:
        fpath = os.path.join(datadir, fname)

    print(fpath)
    if not os.path.exists(fpath):
        print("Downloading data from", origin)

        error_msg = "URL fetch failure on {}: {} -- {}"
        try:
            try:
                urlretrieve(origin, fpath)
            except URLError as e:
                raise Exception(error_msg.format(origin, e.errno, e.reason))
            except HTTPError as e:
                raise Exception(error_msg.format(origin, e.code, e.msg))
        except (Exception, KeyboardInterrupt) as e:
            if os.path.exists(fpath):
                os.remove(fpath)
            raise

    if untar:
        if not os.path.exists(untar_fpath):
            print("Extracting file.")
            with tarfile.open(fpath) as archive:
                archive.extractall(datadir)
        return untar_fpath
```

```
    if extract:
        _extract_archive(fpath, datadir, archive_format)

    return fpath


class AttrDict(dict):
    def __init__(self, *args, **kwargs):
        super(AttrDict, self).__init__(*args, **kwargs)
        self.__dict__ = self


def to_var(tensor, cuda):
    """Wraps a Tensor in a Variable, optionally placing it on the GPU.

    Arguments:
        tensor: A Tensor object.
        cuda: A boolean flag indicating whether to use the GPU.

    Returns:
        A Variable object, on the GPU if cuda==True.
    """
    if cuda:
        return Variable(tensor.cuda())
    else:
        return Variable(tensor)


def create_dir_if_not_exists(directory):
    """Creates a directory if it doesn't already exist."""
    if not os.path.exists(directory):
        os.makedirs(directory)


def save_loss_plot(train_losses, val_losses, opts):
    """Saves a plot of the training and validation loss curves."""
    plt.figure()
    plt.plot(range(len(train_losses)), train_losses)
    plt.plot(range(len(val_losses)), val_losses)
    plt.title("BS={}, nhid={}".format(opts.batch_size, opts.hidden_size),␣
 ↪fontsize=20)
    plt.xlabel("Epochs", fontsize=16)
    plt.ylabel("Loss", fontsize=16)
    plt.xticks(fontsize=14)
    plt.yticks(fontsize=14)
    plt.tight_layout()
    plt.savefig(os.path.join(opts.checkpoint_path, "loss_plot.pdf"))
```

```
    plt.close()


def save_loss_comparison_gru(l1, l2, o1, o2, fn, s=500):
    """Plot comparison of training and val loss curves from GRU runs.

    Arguments:
        l1: Tuple of lists containing training / val losses for model 1.
        l2: Tuple of lists containing training / val losses for model 2.
        o1: Options for model 1.
        o2: Options for model 2.
        fn: Output file name.
        s: Number of training iterations to average over.
    """
    mean_l1 = [np.mean(l1[0][i * s : (i + 1) * s]) for i in range(len(l1[0]) //␣
↪s)]
    mean_l2 = [np.mean(l2[0][i * s : (i + 1) * s]) for i in range(len(l2[0]) //␣
↪s)]

    plt.figure()

    fig, ax = plt.subplots(1, 2, figsize=(10, 4))

    ax[0].plot(range(len(mean_l1)), mean_l1, label="ds=" + o1.data_file_name)
    ax[0].plot(range(len(mean_l2)), mean_l2, label="ds=" + o2.data_file_name)
    ax[0].title.set_text("Train Loss | GRU Hidden Size = {}".format(o2.
↪hidden_size))

    # Validation losses are assumed to be by epoch
    ax[1].plot(range(len(l1[1])), l1[1], label="ds=" + o1.data_file_name)
    ax[1].plot(range(len(l2[1])), l2[1], label="ds=" + o2.data_file_name)
    ax[1].title.set_text("Val Loss | GRU Hidden Size = {}".format(o2.
↪hidden_size))

    ax[0].set_xlabel("Iterations (x{})".format(s), fontsize=10)
    ax[0].set_ylabel("Loss", fontsize=10)
    ax[1].set_xlabel("Epochs", fontsize=10)
    ax[1].set_ylabel("Loss", fontsize=10)
    ax[0].legend(loc="upper right")
    ax[1].legend(loc="upper right")

    fig.suptitle("GRU Performance by Dataset", fontsize=14)
    plt.tight_layout()
    fig.subplots_adjust(top=0.85)
    plt.legend()

    plt_path = "./loss_plot_{}.pdf".format(fn)
```

```
    plt.savefig(plt_path)
    print(f"Plot saved to: {Path(plt_path).resolve()}")


def save_loss_comparison_by_dataset(l1, l2, l3, l4, o1, o2, o3, o4, fn, s=500):
    """Plot comparison of training and validation loss curves from all four
    runs in Part 3, comparing by dataset while holding hidden size constant.

    Models within each pair (l1, l2) and (l3, l4) have the same hidden sizes.

    Arguments:
        l1: Tuple of lists containing training / val losses for model 1.
        l2: Tuple of lists containing training / val losses for model 2.
        l3: Tuple of lists containing training / val losses for model 3.
        l4: Tuple of lists containing training / val losses for model 4.
        o1: Options for model 1.
        o2: Options for model 2.
        o3: Options for model 3.
        o4: Options for model 4.
        fn: Output file name.
        s: Number of training iterations to average over.
    """
    mean_l1 = [np.mean(l1[0][i * s : (i + 1) * s]) for i in range(len(l1[0]) //␣
↪s)]
    mean_l2 = [np.mean(l2[0][i * s : (i + 1) * s]) for i in range(len(l2[0]) //␣
↪s)]
    mean_l3 = [np.mean(l3[0][i * s : (i + 1) * s]) for i in range(len(l3[0]) //␣
↪s)]
    mean_l4 = [np.mean(l4[0][i * s : (i + 1) * s]) for i in range(len(l4[0]) //␣
↪s)]

    plt.figure()
    fig, ax = plt.subplots(2, 2, figsize=(10, 8))

    ax[0][0].plot(range(len(mean_l1)), mean_l1, label="ds=" + o1.data_file_name)
    ax[0][0].plot(range(len(mean_l2)), mean_l2, label="ds=" + o2.data_file_name)
    ax[0][0].title.set_text(
        "Train Loss | Model Hidden Size = {}".format(o1.hidden_size)
    )

    # Validation losses are assumed to be by epoch
    ax[0][1].plot(range(len(l1[1])), l1[1], label="ds=" + o1.data_file_name)
    ax[0][1].plot(range(len(l2[1])), l2[1], label="ds=" + o2.data_file_name)
    ax[0][1].title.set_text("Val Loss | Model Hidden Size = {}".format(o1.
↪hidden_size))

    ax[1][0].plot(range(len(mean_l3)), mean_l3, label="ds=" + o3.data_file_name)
```

```
    ax[1][0].plot(range(len(mean_l4)), mean_l4, label="ds=" + o4.data_file_name)
    ax[1][0].title.set_text(
        "Train Loss | Model Hidden Size = {}".format(o3.hidden_size)
    )

    ax[1][1].plot(range(len(l3[1])), l3[1], label="ds=" + o3.data_file_name)
    ax[1][1].plot(range(len(l4[1])), l4[1], label="ds=" + o4.data_file_name)
    ax[1][1].title.set_text("Val Loss | Model Hidden Size = {}".format(o4.
↪hidden_size))

    for i in range(2):
        ax[i][0].set_xlabel("Iterations (x{})".format(s), fontsize=10)
        ax[i][0].set_ylabel("Loss", fontsize=10)
        ax[i][1].set_xlabel("Epochs", fontsize=10)
        ax[i][1].set_ylabel("Loss", fontsize=10)
        ax[i][0].legend(loc="upper right")
        ax[i][1].legend(loc="upper right")

    fig.suptitle("Performance by Dataset Size", fontsize=16)
    plt.tight_layout()
    fig.subplots_adjust(top=0.9)
    plt.legend()
    plt.savefig("./loss_plot_{}.pdf".format(fn))
    # plt.close()
    plt.show()


def save_loss_comparison_by_hidden(l1, l2, l3, l4, o1, o2, o3, o4, fn, s=500):
    """Plot comparison of training and validation loss curves from all four
    runs in Part 3, comparing by hidden size while holding dataset constant.

    Models within each pair (l1, l3) and (l2, l4) have the same dataset.

    Arguments:
        l1: Tuple of lists containing training / val losses for model 1.
        l2: Tuple of lists containing training / val losses for model 2.
        l3: Tuple of lists containing training / val losses for model 3.
        l4: Tuple of lists containing training / val losses for model 4.
        o1: Options for model 1.
        o2: Options for model 2.
        o3: Options for model 3.
        o4: Options for model 4.
        fn: Output file name.
        s: Number of training iterations to average over.
    """
    mean_l1 = [np.mean(l1[0][i * s : (i + 1) * s]) for i in range(len(l1[0]) //␣
↪s)]
```

```
    mean_l2 = [np.mean(l2[0][i * s : (i + 1) * s]) for i in range(len(l2[0]) //␣
↪s)]
    mean_l3 = [np.mean(l3[0][i * s : (i + 1) * s]) for i in range(len(l3[0]) //␣
↪s)]
    mean_l4 = [np.mean(l4[0][i * s : (i + 1) * s]) for i in range(len(l4[0]) //␣
↪s)]

    plt.figure()
    fig, ax = plt.subplots(2, 2, figsize=(10, 8))

    ax[0][0].plot(range(len(mean_l1)), mean_l1, label="hid_size=" + str(o1.
↪hidden_size))
    ax[0][0].plot(range(len(mean_l3)), mean_l3, label="hid_size=" + str(o3.
↪hidden_size))
    ax[0][0].title.set_text("Train Loss | Dataset = " + o1.data_file_name)

    # Validation losses are assumed to be by epoch
    ax[0][1].plot(range(len(l1[1])), l1[1], label="hid_size=" + str(o1.
↪hidden_size))
    ax[0][1].plot(range(len(l3[1])), l3[1], label="hid_size=" + str(o3.
↪hidden_size))
    ax[0][1].title.set_text("Val Loss | Dataset = " + o1.data_file_name)

    ax[1][0].plot(range(len(mean_l2)), mean_l2, label="hid_size=" + str(o2.
↪hidden_size))
    ax[1][0].plot(range(len(mean_l4)), mean_l4, label="hid_size=" + str(o4.
↪hidden_size))
    ax[1][0].title.set_text("Train Loss | Dataset = " + o3.data_file_name)

    ax[1][1].plot(range(len(l2[1])), l2[1], label="hid_size=" + str(o2.
↪hidden_size))
    ax[1][1].plot(range(len(l4[1])), l4[1], label="hid_size=" + str(o4.
↪hidden_size))
    ax[1][1].title.set_text("Val Loss | Dataset = " + o4.data_file_name)

    for i in range(2):
        ax[i][0].set_xlabel("Iterations (x{})".format(s), fontsize=10)
        ax[i][0].set_ylabel("Loss", fontsize=10)
        ax[i][1].set_xlabel("Epochs", fontsize=10)
        ax[i][1].set_ylabel("Loss", fontsize=10)
        ax[i][0].legend(loc="upper right")
        ax[i][1].legend(loc="upper right")

    fig.suptitle("Performance by Hidden State Size", fontsize=16)
    plt.tight_layout()
    fig.subplots_adjust(top=0.9)
```

```
    plt.legend()
    plt.savefig("./loss_plot_{}.pdf".format(fn))
    # plt.close()
    plt.show()



def checkpoint(encoder, decoder, idx_dict, opts):
    """Saves the current encoder and decoder models, along with idx_dict, which
    contains the char_to_index and index_to_char mappings, and the start_token
    and end_token values.
    """
    with open(os.path.join(opts.checkpoint_path, "encoder.pt"), "wb") as f:
        torch.save(encoder, f)

    with open(os.path.join(opts.checkpoint_path, "decoder.pt"), "wb") as f:
        torch.save(decoder, f)

    with open(os.path.join(opts.checkpoint_path, "idx_dict.pkl"), "wb") as f:
        pkl.dump(idx_dict, f)
```

```
/home/yuling/miniconda3/envs/cuda11.6/lib/python3.9/site-
packages/tqdm/auto.py:22: TqdmWarning: IProgress not found. Please update
jupyter and ipywidgets. See
https://ipywidgets.readthedocs.io/en/stable/user_install.html
  from .autonotebook import tqdm as notebook_tqdm
```

### 3.2 Data loader

[ ]:
```
def read_lines(filename):
    """Read a file and split it into lines."""
    lines = open(filename).read().strip().lower().split("\n")
    return lines


def read_pairs(filename):
    """Reads lines that consist of two words, separated by a space.

    Returns:
        source_words: A list of the first word in each line of the file.
        target_words: A list of the second word in each line of the file.
    """
    lines = read_lines(filename)
    source_words, target_words = [], []
    for line in lines:
        line = line.strip()
        if line:
```

```
            source, target = line.split()
            source_words.append(source)
            target_words.append(target)
    return source_words, target_words


def all_alpha_or_dash(s):
    """Helper function to check whether a string is alphabetic, allowing dashes␣
↪'-'."""
    return all(c.isalpha() or c == "-" for c in s)


def filter_lines(lines):
    """Filters lines to consist of only alphabetic characters or dashes "-"."""
    return [line for line in lines if all_alpha_or_dash(line)]


def load_data(file_name):
    """Loads (English, Pig-Latin) word pairs, and creates mappings from␣
↪characters to indexes."""
    """Loads (English, Pig-Latin) word pairs, and creates mappings from␣
↪characters to indexes."""
    path = "./data/{}.txt".format(file_name)
    source_lines, target_lines = read_pairs(path)

    # Filter lines
    source_lines = filter_lines(source_lines)
    target_lines = filter_lines(target_lines)

    all_characters = set("".join(source_lines)) | set("".join(target_lines))
    # Create a dictionary mapping each character to a unique index
    char_to_index = {
        char: index for (index, char) in enumerate(sorted(list(all_characters)))
    }

    # Add start, end, and  tokens to the dictionary
    start_token = len(char_to_index)
    end_token = len(char_to_index) + 1
    end_prompt_token = len(char_to_index) + 2
    char_to_index["SOS"] = start_token
    char_to_index["EOS"] = end_token
    char_to_index["EOP"] = end_prompt_token

    # Create the inverse mapping, from indexes to characters (used to decode␣
↪the model's predictions)
    index_to_char = {index: char for (char, index) in char_to_index.items()}
```

```
    # Store the final size of the vocabulary
    vocab_size = len(char_to_index)

    line_pairs = list(set(zip(source_lines, target_lines)))  # Python 3
    # import pdb; pdb.set_trace()
    idx_dict = {
        "char_to_index": char_to_index,
        "index_to_char": index_to_char,
        "start_token": start_token,
        "end_token": end_token,
        "end_prompt_token": end_prompt_token
    }

    return line_pairs, vocab_size, idx_dict


def create_dict(pairs):
    """Creates a mapping { (source_length, target_length): [list of (source,␣
 ↪target) pairs]
    This is used to make batches: each batch consists of two parallel tensors,␣
 ↪one containing
    all source indexes and the other containing all corresponding target␣
 ↪indexes.
    Within a batch, all the source words are the same length, and all the␣
 ↪target words are
    the same length.
    """
    unique_pairs = list(set(pairs))  # Find all unique (source, target) pairs

    d = defaultdict(list)
    for (s, t) in unique_pairs:
        d[(len(s), len(t))].append((s, t))

    return d
```

### 3.3 Download dataset

[ ]:
```
#######################################################################
# Download Translation datasets
#######################################################################
data_fpath = get_file(
    fname="pig_latin_small.txt",
    origin="http://www.cs.toronto.edu/~jba/pig_latin_small.txt",
    untar=False,
)
```

```
data_fpath = get_file(
    fname="pig_latin_large.txt",
    origin="http://www.cs.toronto.edu/~jba/pig_latin_large.txt",
    untar=False,
)
```

```
data/pig_latin_small.txt
data/pig_latin_large.txt
```

# 4 Part 1: Scaled Dot Production Attention

## 4.1 Training and evaluation code

[ ]:
```
def string_to_index_list(s, char_to_index, end_token):
    """Converts a sentence into a list of indexes (for each character)."""
    return [char_to_index[char] for char in s] + [
        end_token
    ]  # Adds the end token to each index list


def translate_sentence(sentence, encoder, decoder, idx_dict, opts):
    """Translates a sentence from English to Pig-Latin, by splitting the␣
↪sentence into
    words (whitespace-separated), running the encoder-decoder model to␣
↪translate each
    word independently, and then stitching the words back together with spaces␣
↪between them.
    """
    if idx_dict is None:
        line_pairs, vocab_size, idx_dict = load_data(opts["data_file_name"])
    return " ".join(
        [translate(word, encoder, decoder, idx_dict, opts) for word in sentence.
↪split()]
    )


def translate(input_string, encoder, decoder, idx_dict, opts):
    """Translates a given string from English to Pig-Latin."""

    char_to_index = idx_dict["char_to_index"]
    index_to_char = idx_dict["index_to_char"]
    start_token = idx_dict["start_token"]
    end_token = idx_dict["end_token"]

    max_generated_chars = 20
    gen_string = ""
```

```
    indexes = string_to_index_list(input_string, char_to_index, end_token)
    indexes = to_var(
        torch.LongTensor(indexes).unsqueeze(0), opts.cuda
    )  # Unsqueeze to make it like BS = 1

    encoder_annotations, encoder_last_hidden = encoder(indexes)

    decoder_hidden = encoder_last_hidden
    decoder_input = to_var(torch.LongTensor([[start_token]]), opts.cuda)  # For␣
↪BS = 1
    decoder_inputs = decoder_input

    for i in range(max_generated_chars):
        ## slow decoding, recompute everything at each time
        decoder_outputs, attention_weights = decoder(
            decoder_inputs, encoder_annotations, decoder_hidden
        )

        generated_words = F.softmax(decoder_outputs, dim=2).max(2)[1]
        ni = generated_words.cpu().numpy().reshape(-1)  # LongTensor of size 1
        ni = ni[-1]  # latest output token

        decoder_inputs = torch.cat([decoder_input, generated_words], dim=1)

        if ni == end_token:
            break
        else:
            gen_string = "".join(
                [
                    index_to_char[int(item)]
                    for item in generated_words.cpu().numpy().reshape(-1)
                ]
            )

    return gen_string


def visualize_attention(input_string, encoder, decoder, idx_dict, opts):
    """Generates a heatmap to show where attention is focused in each decoder␣
↪step."""
    if idx_dict is None:
        line_pairs, vocab_size, idx_dict = load_data(opts["data_file_name"])
    char_to_index = idx_dict["char_to_index"]
    index_to_char = idx_dict["index_to_char"]
    start_token = idx_dict["start_token"]
    end_token = idx_dict["end_token"]
```

```
    max_generated_chars = 20
    gen_string = ""

    indexes = string_to_index_list(input_string, char_to_index, end_token)
    indexes = to_var(
        torch.LongTensor(indexes).unsqueeze(0), opts.cuda
    )  # Unsqueeze to make it like BS = 1

    encoder_annotations, encoder_hidden = encoder(indexes)

    decoder_hidden = encoder_hidden
    decoder_input = to_var(torch.LongTensor([[start_token]]), opts.cuda)  # For␣
↪BS = 1
    decoder_inputs = decoder_input

    produced_end_token = False

    for i in range(max_generated_chars):
        ## slow decoding, recompute everything at each time
        decoder_outputs, attention_weights = decoder(
            decoder_inputs, encoder_annotations, decoder_hidden
        )
        generated_words = F.softmax(decoder_outputs, dim=2).max(2)[1]
        ni = generated_words.cpu().numpy().reshape(-1)  # LongTensor of size 1
        ni = ni[-1]  # latest output token

        decoder_inputs = torch.cat([decoder_input, generated_words], dim=1)

        if ni == end_token:
            break
        else:
            gen_string = "".join(
                [
                    index_to_char[int(item)]
                    for item in generated_words.cpu().numpy().reshape(-1)
                ]
            )

    if isinstance(attention_weights, tuple):
        ## transformer's attention mweights
        attention_weights, self_attention_weights = attention_weights

    all_attention_weights = attention_weights.data.cpu().numpy()

    for i in range(len(all_attention_weights)):
        attention_weights_matrix = all_attention_weights[i].squeeze()
        fig = plt.figure()
```

```
        ax = fig.add_subplot(111)
        cax = ax.matshow(attention_weights_matrix, cmap="bone")
        fig.colorbar(cax)

        # Set up axes
        ax.set_yticklabels([""] + list(input_string) + ["EOS"], rotation=90)
        ax.set_xticklabels(
            [""] + list(gen_string) + (["EOS"] if produced_end_token else [])
        )

        # Show label at every tick
        ax.xaxis.set_major_locator(ticker.MultipleLocator(1))
        ax.yaxis.set_major_locator(ticker.MultipleLocator(1))
        # Add title
        plt.xlabel("Attention weights to the source sentence in layer {}".
 ↪format(i + 1))
        plt.tight_layout()
        plt.grid("off")
        plt.show()

    return gen_string


def compute_loss(data_dict, encoder, decoder, idx_dict, criterion, optimizer,␣
 ↪opts):
    """Train/Evaluate the model on a dataset.

    Arguments:
        data_dict: The validation/test word pairs, organized by source and␣
 ↪target lengths.
        encoder: An encoder model to produce annotations for each step of the␣
 ↪input sequence.
        decoder: A decoder model (with or without attention) to generate output␣
 ↪tokens.
        idx_dict: Contains char-to-index and index-to-char mappings, and start␣
 ↪& end token indexes.
        criterion: Used to compute the CrossEntropyLoss for each decoder output.
        optimizer: Train the weights if an optimizer is given. None if only␣
 ↪evaluate the model.
        opts: The command-line arguments.

    Returns:
        mean_loss: The average loss over all batches from data_dict.
    """
    start_token = idx_dict["start_token"]
    end_token = idx_dict["end_token"]
```

```
    char_to_index = idx_dict["char_to_index"]

    losses = []
    for key in data_dict:
        input_strings, target_strings = zip(*data_dict[key])
        input_tensors = [
            torch.LongTensor(string_to_index_list(s, char_to_index, end_token))
            for s in input_strings
        ]
        target_tensors = [
            torch.LongTensor(string_to_index_list(s, char_to_index, end_token))
            for s in target_strings
        ]

        num_tensors = len(input_tensors)
        num_batches = int(np.ceil(num_tensors / float(opts.batch_size)))

        for i in range(num_batches):

            start = i * opts.batch_size
            end = start + opts.batch_size

            inputs = to_var(torch.stack(input_tensors[start:end]), opts.cuda)
            targets = to_var(torch.stack(target_tensors[start:end]), opts.cuda)

            # The batch size may be different in each epoch
            BS = inputs.size(0)

            encoder_annotations, encoder_hidden = encoder(inputs)

            # The last hidden state of the encoder becomes the first hidden␣
↪state of the decoder
            decoder_hidden = encoder_hidden

            start_vector = (
                torch.ones(BS).long().unsqueeze(1) * start_token
            )  # BS x 1 --> 16x1  CHECKED
            decoder_input = to_var(start_vector, opts.cuda)  # BS x 1 --> 16x1 ␣
↪CHECKED

            loss = 0.0

            seq_len = targets.size(1)  # Gets seq_len from BS x seq_len

            decoder_inputs = torch.cat(
                [decoder_input, targets[:, 0:-1]], dim=1
            )  # Gets decoder inputs by shifting the targets to the right
```

```
            decoder_outputs, attention_weights = decoder(
                decoder_inputs, encoder_annotations, decoder_hidden
            )
            decoder_outputs_flatten = decoder_outputs.view(-1, decoder_outputs.
↪size(2))
            targets_flatten = targets.view(-1)

            loss = criterion(decoder_outputs_flatten, targets_flatten)

            losses.append(loss.item())

            ## training if an optimizer is provided
            if optimizer:
                # Zero gradients
                optimizer.zero_grad()
                # Compute gradients
                loss.backward()
                # Update the parameters of the encoder and decoder
                optimizer.step()

    return losses


def training_loop(
    train_dict, val_dict, idx_dict, encoder, decoder, criterion, optimizer, opts
):
    """Runs the main training loop; evaluates the model on the val set every␣
↪epoch.
        * Prints training and val loss each epoch.
        * Prints qualitative translation results each epoch using TEST_SENTENCE
        * Saves an attention map for TEST_WORD_ATTN each epoch
        * Returns loss curves for comparison

    Arguments:
        train_dict: The training word pairs, organized by source and target␣
↪lengths.
        val_dict: The validation word pairs, organized by source and target␣
↪lengths.
        idx_dict: Contains char-to-index and index-to-char mappings, and start␣
↪& end token indexes.
        encoder: An encoder model to produce annotations for each step of the␣
↪input sequence.
        decoder: A decoder model (with or without attention) to generate output␣
↪tokens.
        criterion: Used to compute the CrossEntropyLoss for each decoder output.
```

```
        optimizer: Implements a step rule to update the parameters of the␣
↪encoder and decoder.
        opts: The command-line arguments.

    Returns:
        losses: Lists containing training and validation loss curves.
    """

    start_token = idx_dict["start_token"]
    end_token = idx_dict["end_token"]
    char_to_index = idx_dict["char_to_index"]

    loss_log = open(os.path.join(opts.checkpoint_path, "loss_log.txt"), "w")

    best_val_loss = 1e6
    train_losses = []
    val_losses = []

    mean_train_losses = []
    mean_val_losses = []

    early_stopping_counter = 0

    for epoch in range(opts.nepochs):

        optimizer.param_groups[0]["lr"] *= opts.lr_decay

        train_loss = compute_loss(
            train_dict, encoder, decoder, idx_dict, criterion, optimizer, opts
        )
        val_loss = compute_loss(
            val_dict, encoder, decoder, idx_dict, criterion, None, opts
        )

        mean_train_loss = np.mean(train_loss)
        mean_val_loss = np.mean(val_loss)

        if mean_val_loss < best_val_loss:
            checkpoint(encoder, decoder, idx_dict, opts)
            best_val_loss = mean_val_loss
            early_stopping_counter = 0
        else:
            early_stopping_counter += 1

        if early_stopping_counter > opts.early_stopping_patience:
            print(
```

```
                "Validation loss has not improved in {} epochs, stopping early".
↪format(
                    opts.early_stopping_patience
                )
            )
            print("Obtained lowest validation loss of: {}".
↪format(best_val_loss))
            return (train_losses, mean_val_losses)

        gen_string = translate_sentence(TEST_SENTENCE, encoder, decoder,␣
↪idx_dict, opts)
        print(
            "Epoch: {:3d} | Train loss: {:.3f} | Val loss: {:.3f} | Gen: {:
↪20s}".format(
                epoch, mean_train_loss, mean_val_loss, gen_string
            )
        )

        loss_log.write("{} {} {}\n".format(epoch, train_loss, val_loss))
        loss_log.flush()

        train_losses += train_loss
        val_losses += val_loss

        mean_train_losses.append(mean_train_loss)
        mean_val_losses.append(mean_val_loss)

        save_loss_plot(mean_train_losses, mean_val_losses, opts)

    print("Obtained lowest validation loss of: {}".format(best_val_loss))
    return (train_losses, mean_val_losses)


def print_data_stats(line_pairs, vocab_size, idx_dict):
    """Prints example word pairs, the number of data points, and the vocabulary.
↪"""
    print("=" * 80)
    print("Data Stats".center(80))
    print("-" * 80)
    for pair in line_pairs[:5]:
        print(pair)
    print("Num unique word pairs: {}".format(len(line_pairs)))
    print("Vocabulary: {}".format(idx_dict["char_to_index"].keys()))
    print("Vocab size: {}".format(vocab_size))
    print("=" * 80)
```

```python
def train(opts):
    line_pairs, vocab_size, idx_dict = load_data(opts["data_file_name"])
    print_data_stats(line_pairs, vocab_size, idx_dict)

    # Split the line pairs into an 80% train and 20% val split
    num_lines = len(line_pairs)
    num_train = int(0.8 * num_lines)
    train_pairs, val_pairs = line_pairs[:num_train], line_pairs[num_train:]

    # Group the data by the lengths of the source and target words, to form␣
↪batches
    train_dict = create_dict(train_pairs)
    val_dict = create_dict(val_pairs)

    ##########################################################################
    ### Setup: Create Encoder, Decoder, Learning Criterion, and Optimizers ###
    ##########################################################################
    encoder = TransformerEncoder(
        vocab_size=vocab_size,
        hidden_size=opts.hidden_size,
        num_layers=opts.num_transformer_layers,
        opts=opts,
    )

    decoder = TransformerDecoder(
        vocab_size=vocab_size,
        hidden_size=opts.hidden_size,
        num_layers=opts.num_transformer_layers,
    )

    #### setup checkpoint path
    model_name = "encoder-decoder-h{}-bs{}-{}".format(
        opts.hidden_size, opts.batch_size, opts.data_file_name
    )
    opts.checkpoint_path = model_name
    create_dir_if_not_exists(opts.checkpoint_path)
    ####

    if opts.cuda:
        encoder.cuda()
        decoder.cuda()
        print("Moved models to GPU!")

    criterion = nn.CrossEntropyLoss()
    optimizer = optim.Adam(
        list(encoder.parameters()) + list(decoder.parameters()), lr=opts.
↪learning_rate
```

```
    )

    try:
        losses = training_loop(
            train_dict, val_dict, idx_dict, encoder, decoder, criterion,␣
↪optimizer, opts
        )
    except KeyboardInterrupt:
        print("Exiting early from training.")
        return encoder, decoder, losses

    return encoder, decoder, losses


def print_opts(opts):
    """Prints the values of all command-line arguments."""
    print("=" * 80)
    print("Opts".center(80))
    print("-" * 80)
    for key in opts.__dict__:
        print("{:>30}: {:<30}".format(key, opts.__dict__[key]).center(80))
    print("=" * 80)
```

### 4.2 Step 1: Implement scaled dot-product attention

In the next cell, you will implement the scaled dot-product attention mechanism. See the assignment handouts for details.

[ ]:
```
class ScaledDotAttention(nn.Module):
    def __init__(self, hidden_size):
        super(ScaledDotAttention, self).__init__()

        self.hidden_size = hidden_size

        self.Q = nn.Linear(hidden_size, hidden_size)
        self.K = nn.Linear(hidden_size, hidden_size)
        self.V = nn.Linear(hidden_size, hidden_size)
        self.softmax = nn.Softmax(dim=2)
        self.scaling_factor = torch.rsqrt(
            torch.tensor(self.hidden_size, dtype=torch.float)
        )

    def forward(self, queries, keys, values):
        """The forward pass of the scaled dot attention mechanism.

        Arguments:
```

```
            queries: The current decoder hidden state, 2D or 3D tensor.␣
↪(batch_size x (k) x hidden_size)
            keys: The encoder hidden states for each step of the input sequence.
↪ (batch_size x seq_len x hidden_size)
            values: The encoder hidden states for each step of the input␣
↪sequence. (batch_size x seq_len x hidden_size)

        Returns:
            context: weighted average of the values (batch_size x k x␣
↪hidden_size)
            attention_weights: Normalized attention weights for each encoder␣
↪hidden state. (batch_size x seq_len x k)

            The output must be a softmax weighting over the seq_len annotations.
        """

        batch_size = queries.size(0)

        # Apply linear transformations to the queries, keys, and values
        q = self.Q(queries)
        k = self.K(keys)
        v = self.V(values)

        # Compute the unnormalized attention weights
        unnormalized_attention = torch.bmm(q, k.transpose(1, 2))

        # Apply the scaling factor
        unnormalized_attention = unnormalized_attention * self.scaling_factor

        # Normalize the attention weights using the softmax function
        attention_weights = self.softmax(unnormalized_attention)

        # Compute the context vector by taking the weighted sum of the value␣
↪vectors
        context = torch.bmm(attention_weights, v)

        return context, attention_weights
```

### 4.3 Step 2: Implement causal dot-product Attention

Now, implement the casual scaled dot-product attention mechanism. It will be very similar to your implementation for `ScaledDotAttention`. The additional step is to mask out the attention to future timesteps so this attention mechanism can be used in a decoder. See the assignment handouts for details.

[ ]:
```
class CausalScaledDotAttention(nn.Module):
    def __init__(self, hidden_size):
```

```python
        super(CausalScaledDotAttention, self).__init__()

        self.hidden_size = hidden_size
        self.neg_inf = torch.tensor(-1e7)

        self.Q = nn.Linear(hidden_size, hidden_size)
        self.K = nn.Linear(hidden_size, hidden_size)
        self.V = nn.Linear(hidden_size, hidden_size)
        self.softmax = nn.Softmax(dim=2)
        self.scaling_factor = torch.rsqrt(
            torch.tensor(self.hidden_size, dtype=torch.float)
        )

    def forward(self, queries, keys, values):
        """The forward pass of the scaled dot attention mechanism.

        Arguments:
            queries: The current decoder hidden state, 2D or 3D tensor.␣
↪(batch_size x (k) x hidden_size)
            keys: The encoder hidden states for each step of the input sequence.
↪ (batch_size x seq_len x hidden_size)
            values: The encoder hidden states for each step of the input␣
↪sequence. (batch_size x seq_len x hidden_size)

        Returns:
            context: weighted average of the values (batch_size x k x␣
↪hidden_size)
            attention_weights: Normalized attention weights for each encoder␣
↪hidden state. (batch_size x seq_len x k)

            The output must be a softmax weighting over the seq_len annotations.
        """

        # Get the batch size from the queries tensor
        batch_size = queries.size(0)

        # Apply linear transformations to the queries, keys, and values
        q = self.Q(queries)
        k = self.K(keys)
        v = self.V(values)

        # Compute the unnormalized attention weights
        unnormalized_attention = torch.bmm(q, k.transpose(1, 2))

        # Apply the scaling factor
        unnormalized_attention = unnormalized_attention * self.scaling_factor
```

```
        # Create the mask for causal attention (lower triangular matrix with␣
↪ones)
        mask = torch.tril(torch.ones_like(unnormalized_attention))

        # Apply the mask to the unnormalized_attention tensor, adding negative␣
↪infinity to masked elements
        unnormalized_attention = unnormalized_attention + (self.neg_inf * (1 -␣
↪mask))

        # Normalize the attention weights using the softmax function
        attention_weights = self.softmax(unnormalized_attention)

        # Compute the context vector by taking the weighted sum of the value␣
↪vectors
        context = torch.bmm(attention_weights, v)

        return context, attention_weights
```

### 4.4 Step 3: Transformer encoder and decoder

The following cells provide an implementation of the transformer encoder and decoder that use your `ScaledDotAttention` and `CausalScaledDotAttention`. Please read through them to understand what they are doing.

[ ]:
```
class TransformerEncoder(nn.Module):
    def __init__(self, vocab_size, hidden_size, num_layers, opts):
        super(TransformerEncoder, self).__init__()

        self.vocab_size = vocab_size
        self.hidden_size = hidden_size
        self.num_layers = num_layers
        self.opts = opts

        self.embedding = nn.Embedding(vocab_size, hidden_size)

        self.self_attentions = nn.ModuleList(
            [
                ScaledDotAttention(
                    hidden_size=hidden_size,
                )
                for i in range(self.num_layers)
            ]
        )
        self.attention_mlps = nn.ModuleList(
            [
                nn.Sequential(
                    nn.Linear(hidden_size, hidden_size),
```

```
                    nn.ReLU(),
                )
                for i in range(self.num_layers)
            ]
        )

        self.positional_encodings = self.create_positional_encodings()

    def forward(self, inputs):
        """Forward pass of the encoder RNN.

        Arguments:
            inputs: Input token indexes across a batch for all time steps in␣
↪the sequence. (batch_size x seq_len)

        Returns:
            annotations: The hidden states computed at each step of the input␣
↪sequence. (batch_size x seq_len x hidden_size)
            None: Used to conform to standard encoder return signature.
            None: Used to conform to standard encoder return signature.
        """
        batch_size, seq_len = inputs.size()

        encoded = self.embedding(inputs)  # batch_size x seq_len x hidden_size

        # Add positinal embeddings from self.create_positional_encodings. (a'la␣
↪https://arxiv.org/pdf/1706.03762.pdf, section 3.5)
        encoded = encoded + self.positional_encodings[:seq_len]

        annotations = encoded
        for i in range(self.num_layers):
            new_annotations, self_attention_weights = self.self_attentions[i](
                annotations, annotations, annotations
            )  # batch_size x seq_len x hidden_size
            residual_annotations = annotations + new_annotations
            new_annotations = self.attention_mlps[i](residual_annotations)
            annotations = residual_annotations + new_annotations

        # Transformer encoder does not have a last hidden or cell layer.
        return annotations, None
        # return annotations, None, None
    def create_positional_encodings(self, max_seq_len=1000):
        """Creates positional encodings for the inputs.

        Arguments:
            max_seq_len: a number larger than the maximum string length we␣
↪expect to encounter during training
```

```
        Returns:
            pos_encodings: (max_seq_len, hidden_dim) Positional encodings for a␣
↪sequence with length max_seq_len.
        """
        pos_indices = torch.arange(max_seq_len)[..., None]
        dim_indices = torch.arange(self.hidden_size // 2)[None, ...]
        exponents = (2 * dim_indices).float() / (self.hidden_size)
        trig_args = pos_indices / (10000**exponents)
        sin_terms = torch.sin(trig_args)
        cos_terms = torch.cos(trig_args)

        pos_encodings = torch.zeros((max_seq_len, self.hidden_size))
        pos_encodings[:, 0::2] = sin_terms
        pos_encodings[:, 1::2] = cos_terms

        if self.opts.cuda:
            pos_encodings = pos_encodings.cuda()

        return pos_encodings
```

[ ]:
```
class TransformerDecoder(nn.Module):
    def __init__(self, vocab_size, hidden_size, num_layers):
        super(TransformerDecoder, self).__init__()
        self.vocab_size = vocab_size
        self.hidden_size = hidden_size

        self.embedding = nn.Embedding(vocab_size, hidden_size)
        self.num_layers = num_layers

        self.self_attentions = nn.ModuleList(
            [
                CausalScaledDotAttention(
                    hidden_size=hidden_size,
                )
                for i in range(self.num_layers)
            ]
        )
        self.encoder_attentions = nn.ModuleList(
            [
                ScaledDotAttention(
                    hidden_size=hidden_size,
                )
                for i in range(self.num_layers)
            ]
        )
        self.attention_mlps = nn.ModuleList(
```

```
            [
                nn.Sequential(
                    nn.Linear(hidden_size, hidden_size),
                    nn.ReLU(),
                )
                for i in range(self.num_layers)
            ]
        )
        self.out = nn.Linear(hidden_size, vocab_size)

        self.positional_encodings = self.create_positional_encodings()

    def forward(self, inputs, annotations, hidden_init):
        """Forward pass of the attention-based decoder RNN.

        Arguments:
            inputs: Input token indexes across a batch for all the time step.␣
↪(batch_size x decoder_seq_len)
            annotations: The encoder hidden states for each step of the input.
                         sequence. (batch_size x seq_len x hidden_size)
            hidden_init: Not used in the transformer decoder
        Returns:
            output: Un-normalized scores for each token in the vocabulary,␣
↪across a batch for all the decoding time steps. (batch_size x␣
↪decoder_seq_len x vocab_size)
            attentions: The stacked attention weights applied to the encoder␣
↪annotations (batch_size x encoder_seq_len x decoder_seq_len)
        """

        batch_size, seq_len = inputs.size()
        embed = self.embedding(inputs)  # batch_size x seq_len x hidden_size

        embed = embed + self.positional_encodings[:seq_len]

        encoder_attention_weights_list = []
        self_attention_weights_list = []
        contexts = embed
        for i in range(self.num_layers):
            new_contexts, self_attention_weights = self.self_attentions[i](
                contexts, contexts, contexts
            )  # batch_size x seq_len x hidden_size
            residual_contexts = contexts + new_contexts
            new_contexts, encoder_attention_weights = self.
↪encoder_attentions[i](
                residual_contexts, annotations, annotations
            )  # batch_size x seq_len x hidden_size
            residual_contexts = residual_contexts + new_contexts
```

```
            new_contexts = self.attention_mlps[i](residual_contexts)
            contexts = residual_contexts + new_contexts

            encoder_attention_weights_list.append(encoder_attention_weights)
            self_attention_weights_list.append(self_attention_weights)

        output = self.out(contexts)
        encoder_attention_weights = torch.stack(encoder_attention_weights_list)
        self_attention_weights = torch.stack(self_attention_weights_list)

        return output, (encoder_attention_weights, self_attention_weights)

    def create_positional_encodings(self, max_seq_len=1000):
        """Creates positional encodings for the inputs.

        Arguments:
            max_seq_len: a number larger than the maximum string length we␣
↪expect to encounter during training

        Returns:
            pos_encodings: (max_seq_len, hidden_dim) Positional encodings for a␣
↪sequence with length max_seq_len.
        """
        pos_indices = torch.arange(max_seq_len)[..., None]
        dim_indices = torch.arange(self.hidden_size // 2)[None, ...]
        exponents = (2 * dim_indices).float() / (self.hidden_size)
        trig_args = pos_indices / (10000**exponents)
        sin_terms = torch.sin(trig_args)
        cos_terms = torch.cos(trig_args)

        pos_encodings = torch.zeros((max_seq_len, self.hidden_size))
        pos_encodings[:, 0::2] = sin_terms
        pos_encodings[:, 1::2] = cos_terms

        pos_encodings = pos_encodings.cuda()

        return pos_encodings
```

### 4.5 Step 4: Training and analysis (with scaled dot-product attention)

Now we will train a (simplified) transformer encoder-decoder model.

First, we train our smaller model on the small dataset. Use this model to answer Question 4 in the handout.

[ ]:
```
use_cuda = torch.cuda.is_available()
TEST_SENTENCE = "the air conditioning is working"
```

```python
trans32_args_s = AttrDict()
args_dict = {
    "data_file_name": "pig_latin_small",
    "cuda": use_cuda,
    "nepochs": 100,
    "checkpoint_dir": "checkpoints",
    "learning_rate": 5e-4,
    "early_stopping_patience": 100,
    "lr_decay": 0.99,
    "batch_size": 64,
    "hidden_size": 32,
    "num_transformer_layers": 4,
}


trans32_args_s.update(args_dict)
print_opts(trans32_args_s)

trans32_encoder_s, trans32_decoder_s, trans32_losses_s = train(trans32_args_s)

translated = translate_sentence(
    TEST_SENTENCE, trans32_encoder_s, trans32_decoder_s, None, trans32_args_s
)
print("source:\t\t{} \ntranslated:\t{}".format(TEST_SENTENCE, translated))
```

```
================================================================================
                                       Opts
--------------------------------------------------------------------------------
                         data_file_name: pig_latin_small
                                   cuda: 1
                               nepochs: 100
                         checkpoint_dir: checkpoints
                          learning_rate: 0.0005
                early_stopping_patience: 100
                                lr_decay: 0.99
                             batch_size: 64
                            hidden_size: 32
                 num_transformer_layers: 4
================================================================================
================================================================================
                                   Data Stats
--------------------------------------------------------------------------------
('irresistible', 'irresistibleway')
('repentance', 'epentanceray')
('pretty', 'ettypray')
('parcels', 'arcelspay')
('survey', 'urveysay')
```

```
Num unique word pairs: 3198
Vocabulary: dict_keys(['-', 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j',
'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z',
'SOS', 'EOS', 'EOP'])
Vocab size: 30
================================================================================
Moved models to GPU!
Epoch:   0 | Train loss: 3.021 | Val loss: 2.465 | Gen: ay ayay
ointintintntntntntay iiiiitay ay
Epoch:   1 | Train loss: 2.178 | Val loss: 2.098 | Gen: ay iay ontinay iiiitay
ingay
Epoch:   2 | Train loss: 1.883 | Val loss: 1.908 | Gen: etay ay onginginay isway
oray
Epoch:   3 | Train loss: 1.695 | Val loss: 1.804 | Gen: etay ilay onananay isway
ogay
Epoch:   4 | Train loss: 1.551 | Val loss: 1.709 | Gen: etay iaway incncncinay
isway oray-ay-ay-ongay
Epoch:   5 | Train loss: 1.445 | Val loss: 1.741 | Gen: etay iray oninginay
isway oringray
Epoch:   6 | Train loss: 1.367 | Val loss: 1.585 | Gen: etay ilway inctincionay
isway oway
Epoch:   7 | Train loss: 1.335 | Val loss: 1.706 | Gen: ehay irway inciondionway
isway owingray
Epoch:   8 | Train loss: 1.318 | Val loss: 1.555 | Gen: eay irway inctincay iway
oway
Epoch:   9 | Train loss: 1.197 | Val loss: 1.505 | Gen: eay irway
ontingingingiongiong iway owringway
Epoch:  10 | Train loss: 1.120 | Val loss: 1.439 | Gen: eay ariray ontinday iway
owringway
Epoch:  11 | Train loss: 1.052 | Val loss: 1.408 | Gen: eay arway
onciondiongiongway iway owringway
Epoch:  12 | Train loss: 1.004 | Val loss: 1.399 | Gen: eay irway ontinday isway
owwringway
Epoch:  13 | Train loss: 0.959 | Val loss: 1.343 | Gen: eay irway oncioncay iway
owwray
Epoch:  14 | Train loss: 0.923 | Val loss: 1.341 | Gen: eay arway ondingionday
isway owwringway
Epoch:  15 | Train loss: 0.891 | Val loss: 1.303 | Gen: ehay arway
ontionayionday isway owwray
Epoch:  16 | Train loss: 0.857 | Val loss: 1.357 | Gen: etwhay arway ontionay
isway owwray
Epoch:  17 | Train loss: 0.822 | Val loss: 1.295 | Gen: ethay arway
ontionayionday isway owwray
Epoch:  18 | Train loss: 0.795 | Val loss: 1.294 | Gen: ehay iraywayway
oncionciontionay isway owwringway
Epoch:  19 | Train loss: 0.776 | Val loss: 1.283 | Gen: ehay iray
ontiondiondationgigw isway owingway
Epoch:  20 | Train loss: 0.746 | Val loss: 1.261 | Gen: ehay ariway
```

```
oncionayioncay isway owwringway
Epoch:  21 | Train loss: 0.709 | Val loss: 1.212 | Gen: ethay arway onditionday
isway owringway
Epoch:  22 | Train loss: 0.673 | Val loss: 1.244 | Gen: ehay arway
onctionayionday isway oringway
Epoch:  23 | Train loss: 0.658 | Val loss: 1.198 | Gen: ethay arway ontionday
isway owringway
Epoch:  24 | Train loss: 0.633 | Val loss: 1.255 | Gen: ethay ariray onctionday
isway owringway
Epoch:  25 | Train loss: 0.634 | Val loss: 1.294 | Gen: ethay arway ontinday
isway orway
Epoch:  26 | Train loss: 0.619 | Val loss: 1.198 | Gen: ehay airway
ontindayiongiongnway isway oringway
Epoch:  27 | Train loss: 0.575 | Val loss: 1.347 | Gen: ethay arway ontinday
sway orway
Epoch:  28 | Train loss: 0.605 | Val loss: 1.177 | Gen: ehay irayway
ontinaionginday isway owringway
Epoch:  29 | Train loss: 0.551 | Val loss: 1.184 | Gen: ethay iray ontintionay
isway oringway
Epoch:  30 | Train loss: 0.523 | Val loss: 1.170 | Gen: ethay ariiway
ontinciondiongiongwa isway oringway
Epoch:  31 | Train loss: 0.516 | Val loss: 1.185 | Gen: ethay iray ontitionday
isway origway
Epoch:  32 | Train loss: 0.490 | Val loss: 1.170 | Gen: ethay arway
ontindiongigigway isway origway
Epoch:  33 | Train loss: 0.462 | Val loss: 1.215 | Gen: ethay airway
ontintionday isway oringway
Epoch:  34 | Train loss: 0.452 | Val loss: 1.170 | Gen: ethay airway
ontinciongiongday isway oringway
Epoch:  35 | Train loss: 0.434 | Val loss: 1.232 | Gen: ethay ariway ontitionday
isway origway
Epoch:  36 | Train loss: 0.424 | Val loss: 1.145 | Gen: ethay airway
ontinciongiongway isway owringway
Epoch:  37 | Train loss: 0.411 | Val loss: 1.246 | Gen: ethay arway
ontionditingway isway oringway
Epoch:  38 | Train loss: 0.399 | Val loss: 1.134 | Gen: ethay arway
ontionditiongday isway owringway
Epoch:  39 | Train loss: 0.390 | Val loss: 1.312 | Gen: ethay arway ontitionday
isway oringway
Epoch:  40 | Train loss: 0.431 | Val loss: 1.202 | Gen: etway arway
onctionationday isway origway
Epoch:  41 | Train loss: 0.403 | Val loss: 1.188 | Gen: ethay arway
ontidiongigday isway origway
Epoch:  42 | Train loss: 0.370 | Val loss: 1.253 | Gen: ethay ariway
ontionditiongdationa isway oringway
Epoch:  43 | Train loss: 0.406 | Val loss: 1.450 | Gen: ethay irway
oncitionciocay isway orkingway
Epoch:  44 | Train loss: 0.542 | Val loss: 1.069 | Gen: ethay ariway
```

```
ontitinaingionday isway orkingway
Epoch:  45 | Train loss: 0.439 | Val loss: 1.056 | Gen: ethay arway ontidioncay
isway orkingway
Epoch:  46 | Train loss: 0.382 | Val loss: 1.019 | Gen: ethay airway
onditiondidiongway isway orkingway
Epoch:  47 | Train loss: 0.344 | Val loss: 1.002 | Gen: ethay arway
ontitiondiday isway oringway
Epoch:  48 | Train loss: 0.315 | Val loss: 0.990 | Gen: ethay airway
ontitiondiday isway oringway
Epoch:  49 | Train loss: 0.304 | Val loss: 0.982 | Gen: ethay ariway
ontitiondiday isway oringway
Epoch:  50 | Train loss: 0.287 | Val loss: 0.989 | Gen: ethay aiway
onditiondiday isway oringway
Epoch:  51 | Train loss: 0.272 | Val loss: 0.991 | Gen: ethay ariway
onditionditiongway isway oringway
Epoch:  52 | Train loss: 0.261 | Val loss: 1.001 | Gen: ethay ariway
onditionditiongway isway oringway
Epoch:  53 | Train loss: 0.252 | Val loss: 1.000 | Gen: ethay ariway
onditiondidationgway isway oringway
Epoch:  54 | Train loss: 0.243 | Val loss: 1.005 | Gen: ethay ariway
onditionditiongway isway oringway
Epoch:  55 | Train loss: 0.235 | Val loss: 1.007 | Gen: ethay ariway
onditionditiongway isway oringway
Epoch:  56 | Train loss: 0.227 | Val loss: 1.018 | Gen: ethay ariway
onditionditiongway isway oringway
Epoch:  57 | Train loss: 0.220 | Val loss: 1.023 | Gen: ethay ariway
onditiondidationgway isway oringway
Epoch:  58 | Train loss: 0.213 | Val loss: 1.029 | Gen: ethay ariway
onditiondidationgway isway oringway
Epoch:  59 | Train loss: 0.205 | Val loss: 1.031 | Gen: ethay ariway
onditiondidationgway isway oringway
Epoch:  60 | Train loss: 0.199 | Val loss: 1.035 | Gen: ethay ariway
onditionditiongway isway oringway
Epoch:  61 | Train loss: 0.193 | Val loss: 1.051 | Gen: ethay ariway
onditionditiongway isway oringway
Epoch:  62 | Train loss: 0.187 | Val loss: 1.056 | Gen: ethay ariway
onditionditiongway isway oringway
Epoch:  63 | Train loss: 0.181 | Val loss: 1.069 | Gen: ethay ariway
onditionditiongway isway oringway
Epoch:  64 | Train loss: 0.190 | Val loss: 1.090 | Gen: ethay ariway
onditionationcay isway oringway
Epoch:  65 | Train loss: 0.185 | Val loss: 1.058 | Gen: ethay ariway
onditionditiongway isway orkingway
Epoch:  66 | Train loss: 0.209 | Val loss: 1.274 | Gen: ethay airway
onditionditway isway owringway
Epoch:  67 | Train loss: 0.327 | Val loss: 1.266 | Gen: etay aiway
onvioniongincay isway orkingway
Epoch:  68 | Train loss: 0.303 | Val loss: 1.088 | Gen: ethay airway
```

```
onditioncioncay isway oringway
Epoch:  69 | Train loss: 0.249 | Val loss: 1.023 | Gen: ethay arway onditioncay
isway orkingway
Epoch:  70 | Train loss: 0.204 | Val loss: 0.971 | Gen: ethay ariway
onditionditioncay isway owringway
Epoch:  71 | Train loss: 0.199 | Val loss: 0.983 | Gen: ethay ariway
onditiondingday isway oringway
Epoch:  72 | Train loss: 0.179 | Val loss: 0.950 | Gen: ethay ariway onditioncay
isway orkingway
Epoch:  73 | Train loss: 0.156 | Val loss: 0.954 | Gen: ethay ariway onditioncay
isway oringway
Epoch:  74 | Train loss: 0.145 | Val loss: 0.975 | Gen: ethay ariway onditioncay
isway oringway
Epoch:  75 | Train loss: 0.139 | Val loss: 0.991 | Gen: ethay ariway onditioncay
isway oringway
Epoch:  76 | Train loss: 0.134 | Val loss: 1.005 | Gen: ethay ariway onditioncay
isway oringway
Epoch:  77 | Train loss: 0.129 | Val loss: 1.013 | Gen: ethay ariway onditioncay
isway oringway
Epoch:  78 | Train loss: 0.124 | Val loss: 1.024 | Gen: ethay ariway onditioncay
isway oringway
Epoch:  79 | Train loss: 0.120 | Val loss: 1.035 | Gen: ethay ariway onditioncay
isway oringway
Epoch:  80 | Train loss: 0.116 | Val loss: 1.045 | Gen: ethay ariway onditioncay
isway orkingway
Epoch:  81 | Train loss: 0.112 | Val loss: 1.057 | Gen: ethay ariway onditioncay
isway orkingway
Epoch:  82 | Train loss: 0.108 | Val loss: 1.066 | Gen: ethay ariway onditioncay
isway orkingway
Epoch:  83 | Train loss: 0.104 | Val loss: 1.079 | Gen: ethay ariway onditioncay
isway orkingway
Epoch:  84 | Train loss: 0.101 | Val loss: 1.089 | Gen: ethay ariway onditioncay
isway orkingway
Epoch:  85 | Train loss: 0.098 | Val loss: 1.106 | Gen: ethay ariway
onditiondioncay isway orkingway
Epoch:  86 | Train loss: 0.094 | Val loss: 1.111 | Gen: ethay ariway onditioncay
isway orkingway
Epoch:  87 | Train loss: 0.091 | Val loss: 1.129 | Gen: ethay ariway
onditiondioncay isway orkingway
Epoch:  88 | Train loss: 0.088 | Val loss: 1.138 | Gen: ethay ariway onditioncay
isway orkingway
Epoch:  89 | Train loss: 0.085 | Val loss: 1.157 | Gen: ethay ariway onditioncay
isway orkingway
Epoch:  90 | Train loss: 0.082 | Val loss: 1.164 | Gen: ethay ariway onditioncay
isway orkingway
Epoch:  91 | Train loss: 0.082 | Val loss: 1.218 | Gen: ethay ariway onditioncay
isway orkingway
Epoch:  92 | Train loss: 0.177 | Val loss: 1.210 | Gen: ethay ariway
```

```
onditindoncay isway orkingway
Epoch:  93 | Train loss: 0.212 | Val loss: 1.294 | Gen: ethay ariway onditioncay
isway orkingway
Epoch:  94 | Train loss: 0.224 | Val loss: 1.305 | Gen: ethay ariway
ontitiondioncay isway oringway
Epoch:  95 | Train loss: 0.243 | Val loss: 1.106 | Gen: ethay ariway
onditioncioncay isway orkingway
Epoch:  96 | Train loss: 0.166 | Val loss: 1.079 | Gen: ethay ariway
onditiondioncay isway orkingway
Epoch:  97 | Train loss: 0.117 | Val loss: 0.956 | Gen: ethay ariway onditioncay
isway orkingway
Epoch:  98 | Train loss: 0.100 | Val loss: 0.965 | Gen: ethay ariway onditioncay
isway orkingway
Epoch:  99 | Train loss: 0.090 | Val loss: 0.971 | Gen: ethay ariway onditioncay
isway orkingway
Obtained lowest validation loss of: 0.9500901771839275
source:         the air conditioning is working
translated:     ethay ariway onditioncay isway orkingway
```

[ ]:
```
TEST_SENTENCE = "the air conditioning is working"
translated = translate_sentence(
    TEST_SENTENCE, trans32_encoder_s, trans32_decoder_s, None, trans32_args_s
)
print("source:\t\t{} \ntranslated:\t{}".format(TEST_SENTENCE, translated))
```

```
source:         the air conditioning is working
translated:     ethay ariway onditioncay isway orkingway
```

In the following cells, we investigate the effects of increasing model size and dataset size on the training / validation curves and generalization of the Transformer. We will increase hidden size to 64, and also increase dataset size. Include the best achieved validation loss in your report.

[ ]:
```
TEST_SENTENCE = "the air conditioning is working"

trans32_args_l = AttrDict()
args_dict = {
    "data_file_name": "pig_latin_large",  # Increased data set size
    "cuda": True,
    "nepochs": 100,
    "checkpoint_dir": "checkpoints",
    "learning_rate": 5e-4,
    "early_stopping_patience": 10,
    "lr_decay": 0.99,
    "batch_size": 512,
    "hidden_size": 32,
    "num_transformer_layers": 3,
}
trans32_args_l.update(args_dict)
```

```python
print_opts(trans32_args_l)

trans32_encoder_l, trans32_decoder_l, trans32_losses_l = train(trans32_args_l)

translated = translate_sentence(
    TEST_SENTENCE, trans32_encoder_l, trans32_decoder_l, None, trans32_args_l
)
print("source:\t\t{} \ntranslated:\t{}".format(TEST_SENTENCE, translated))
```

```
================================================================================
                                       Opts
--------------------------------------------------------------------------------
                          data_file_name: pig_latin_large
                                    cuda: 1
                                  nepochs: 100
                          checkpoint_dir: checkpoints
                           learning_rate: 0.0005
                  early_stopping_patience: 10
                                 lr_decay: 0.99
                              batch_size: 512
                             hidden_size: 32
                   num_transformer_layers: 3
================================================================================
================================================================================
                                     Data Stats
--------------------------------------------------------------------------------
('tribune', 'ibunetray')
('sheriff', 'eriffshay')
('sobbed', 'obbedsay')
('monte', 'ontemay')
('ere', 'ereway')
Num unique word pairs: 22402
Vocabulary: dict_keys(['-', 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j',
'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z',
'SOS', 'EOS', 'EOP'])
Vocab size: 30
================================================================================
Moved models to GPU!
Epoch:   0 | Train loss: 2.923 | Val loss: 2.348 | Gen: elay aay-ay-ay
iiiiiiinay-ay-ay-oin esssstay iiiiiiay-ay-ay-ay
Epoch:   1 | Train loss: 2.120 | Val loss: 2.082 | Gen: ed-ted-ed-ay away-ay
intay-inay-ay-onay-a esstay oway
Epoch:   2 | Train loss: 1.885 | Val loss: 1.958 | Gen: etetetetetay away-ay-ay
intintintay-intay-in estay oway
Epoch:   3 | Train loss: 1.723 | Val loss: 1.826 | Gen: etetay away
ontintintintintay istay oway
Epoch:   4 | Train loss: 1.606 | Val loss: 1.800 | Gen: etetay-ay away-ay-ay-ay-
```

```
ay ontintintintay istay ootinway
Epoch:   5 | Train loss: 1.526 | Val loss: 1.711 | Gen: eway away
ontintintintintay ssay oway
Epoch:   6 | Train loss: 1.407 | Val loss: 1.639 | Gen: eteway away-iway
ontintintintay istay otinway-inway
Epoch:   7 | Train loss: 1.342 | Val loss: 1.578 | Gen: eway array
ontindintintay isay ootingay
Epoch:   8 | Train loss: 1.276 | Val loss: 1.589 | Gen: eteyway arrray
ondindintintay istay ootingay
Epoch:   9 | Train loss: 1.260 | Val loss: 1.535 | Gen: eay array onindindinay
isay ootingay
Epoch:  10 | Train loss: 1.198 | Val loss: 1.598 | Gen: eteway arrriray
ontintintintay isay otingay
Epoch:  11 | Train loss: 1.151 | Val loss: 1.499 | Gen: eway aririray
ontintintintay isway otingay
Epoch:  12 | Train loss: 1.094 | Val loss: 1.560 | Gen: eway arirway ontintintay
isay otingayway
Epoch:  13 | Train loss: 1.059 | Val loss: 1.423 | Gen: eway aririray
ontintintintay isay ootingay
Epoch:  14 | Train loss: 1.014 | Val loss: 1.326 | Gen: ehay arrray ondintintay
isay otghay
Epoch:  15 | Train loss: 0.967 | Val loss: 1.307 | Gen: ehay arirray
ontintintinway isay otingray
Epoch:  16 | Train loss: 0.940 | Val loss: 1.371 | Gen: ethay arrway ondinday
isay otghay
Epoch:  17 | Train loss: 0.911 | Val loss: 1.197 | Gen: eway arirway
ondiotintntay isay otingway
Epoch:  18 | Train loss: 0.878 | Val loss: 1.258 | Gen: ethay arrray ondidiontay
isay otghay-ghray
Epoch:  19 | Train loss: 0.836 | Val loss: 1.148 | Gen: ethay arirway
ondiotintay isay otingray
Epoch:  20 | Train loss: 0.806 | Val loss: 1.171 | Gen: ethway arirway
ondintinay isay oringray
Epoch:  21 | Train loss: 0.783 | Val loss: 1.102 | Gen: ethway arirway
ondiotintinway isay oringray
Epoch:  22 | Train loss: 0.752 | Val loss: 1.093 | Gen: ethway arirway
ondintiontinay isway oringray
Epoch:  23 | Train loss: 0.727 | Val loss: 1.097 | Gen: ethay arirway
ondiotinantinay isway othghay
Epoch:  24 | Train loss: 0.696 | Val loss: 1.075 | Gen: ethway arirway
ondintiontinay isway oringray
Epoch:  25 | Train loss: 0.672 | Val loss: 1.014 | Gen: ethay arirway ondiotinay
isway oringay
Epoch:  26 | Train loss: 0.658 | Val loss: 1.083 | Gen: ethway arrrway
ondintiontinay isway orkingray
Epoch:  27 | Train loss: 0.638 | Val loss: 1.006 | Gen: ethay arirway
onditinantinay isway oringay
Epoch:  28 | Train loss: 0.616 | Val loss: 1.029 | Gen: ethway arirway
```

```
ondintiongnay isway orkingrway
Epoch:  29 | Train loss: 0.593 | Val loss: 0.971 | Gen: ethay arirway
onditionionay isway oringay
Epoch:  30 | Train loss: 0.576 | Val loss: 0.964 | Gen: ethay arirway
ondintioninway isway orkingway
Epoch:  31 | Train loss: 0.553 | Val loss: 0.954 | Gen: ethay ariway
onditionionay isway oringay
Epoch:  32 | Train loss: 0.539 | Val loss: 0.943 | Gen: ethay arirway
ondintioningway isway orkingway
Epoch:  33 | Train loss: 0.523 | Val loss: 1.104 | Gen: ethay ariway
onditionionionay isway oringay
Epoch:  34 | Train loss: 0.529 | Val loss: 1.017 | Gen: ethway arirway
ondintioniongway isway orkingway
Epoch:  35 | Train loss: 0.514 | Val loss: 1.118 | Gen: ethay arirway
onditionioniongway isway oringkway
Epoch:  36 | Train loss: 0.530 | Val loss: 0.970 | Gen: ethay arrrway
ondintiongnay isway orkingray
Epoch:  37 | Train loss: 0.520 | Val loss: 0.888 | Gen: ethay ariway
onditioniongway isway orkingway
Epoch:  38 | Train loss: 0.509 | Val loss: 0.968 | Gen: ethay ariway
onditioniongay isway orkingway
Epoch:  39 | Train loss: 0.511 | Val loss: 0.986 | Gen: thway airway
onditioninay isay oway-ingway
Epoch:  40 | Train loss: 0.474 | Val loss: 1.015 | Gen: thay ariway
onditioniongway isay orkingway
Epoch:  41 | Train loss: 0.460 | Val loss: 0.919 | Gen: ethay array
onditioniongnay isway orkingway
Epoch:  42 | Train loss: 0.438 | Val loss: 0.857 | Gen: ethay ariway
onditioniongway isay oringhay
Epoch:  43 | Train loss: 0.420 | Val loss: 0.818 | Gen: ethay arrway
onditioniongway isway orkingway
Epoch:  44 | Train loss: 0.401 | Val loss: 0.817 | Gen: ethay ariway
onditioniongway isway orkingway
Epoch:  45 | Train loss: 0.386 | Val loss: 0.868 | Gen: ethay ariway
onditioniongcay isay orkingway
Epoch:  46 | Train loss: 0.378 | Val loss: 0.771 | Gen: ethay ariway
onditioniongway isway orkingway
Epoch:  47 | Train loss: 0.363 | Val loss: 0.850 | Gen: ethay ariway
onditioniongcay isway orkingway
Epoch:  48 | Train loss: 0.358 | Val loss: 0.755 | Gen: ethay ariway
onditioniongcay isway orkingway
Epoch:  49 | Train loss: 0.347 | Val loss: 0.880 | Gen: ethay ariway
onditioniongcay isway orkingway
Epoch:  50 | Train loss: 0.349 | Val loss: 0.720 | Gen: ethay ariway
onditioningcay isway orkingway
Epoch:  51 | Train loss: 0.335 | Val loss: 0.906 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  52 | Train loss: 0.337 | Val loss: 0.704 | Gen: ethay ariway
```

```
onditioningcay isway orkingway
Epoch:  53 | Train loss: 0.320 | Val loss: 0.846 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  54 | Train loss: 0.318 | Val loss: 0.703 | Gen: ethay ariway
onditioniongcay isway orkingway
Epoch:  55 | Train loss: 0.305 | Val loss: 0.844 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  56 | Train loss: 0.304 | Val loss: 0.702 | Gen: ethay ariway
onditioniongcay isway orkingway
Epoch:  57 | Train loss: 0.289 | Val loss: 0.801 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  58 | Train loss: 0.285 | Val loss: 0.712 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  59 | Train loss: 0.276 | Val loss: 0.795 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  60 | Train loss: 0.274 | Val loss: 0.698 | Gen: ethay ariway
onditioniongcay isway orkingway
Epoch:  61 | Train loss: 0.268 | Val loss: 0.812 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  62 | Train loss: 0.262 | Val loss: 0.694 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  63 | Train loss: 0.253 | Val loss: 0.754 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  64 | Train loss: 0.246 | Val loss: 0.693 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  65 | Train loss: 0.240 | Val loss: 0.747 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  66 | Train loss: 0.234 | Val loss: 0.685 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  67 | Train loss: 0.232 | Val loss: 0.744 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  68 | Train loss: 0.224 | Val loss: 0.668 | Gen: ethay airway
onditioningcay isway orkingway
Epoch:  69 | Train loss: 0.221 | Val loss: 0.734 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  70 | Train loss: 0.218 | Val loss: 0.681 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  71 | Train loss: 0.215 | Val loss: 0.703 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  72 | Train loss: 0.209 | Val loss: 0.645 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  73 | Train loss: 0.206 | Val loss: 0.688 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  74 | Train loss: 0.195 | Val loss: 0.649 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  75 | Train loss: 0.192 | Val loss: 0.661 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  76 | Train loss: 0.191 | Val loss: 0.665 | Gen: ethay airway
```

```
onditioniongcay isway orkingway
Epoch:  77 | Train loss: 0.187 | Val loss: 0.645 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  78 | Train loss: 0.178 | Val loss: 0.661 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  79 | Train loss: 0.175 | Val loss: 0.653 | Gen: ethay airway
onditioningcay isway orkingway
Epoch:  80 | Train loss: 0.170 | Val loss: 0.668 | Gen: ethay airway
onditioningcay isway orkingway
Epoch:  81 | Train loss: 0.170 | Val loss: 0.657 | Gen: ethay airway
onditioniongcay isway orkingway
Epoch:  82 | Train loss: 0.165 | Val loss: 0.666 | Gen: ethay airway
onditioningcay isway orkingway
Epoch:  83 | Train loss: 0.162 | Val loss: 0.656 | Gen: ethay airway
onditioningcay isway orkingway
Epoch:  84 | Train loss: 0.156 | Val loss: 0.663 | Gen: ethay airway
onditioningcay isway orkingway
Epoch:  85 | Train loss: 0.152 | Val loss: 0.660 | Gen: ethay airway
onditioningcay isway orkingway
Epoch:  86 | Train loss: 0.148 | Val loss: 0.668 | Gen: ethay airway
onditioningcay isway orkingway
Epoch:  87 | Train loss: 0.145 | Val loss: 0.670 | Gen: ethay airway
onditioningcay isway orkingway
Validation loss has not improved in 10 epochs, stopping early
Obtained lowest validation loss of: 0.6452513326491628
source:         the air conditioning is working
translated:     ethay airway onditioningcay isway orkingway
```

[ ]:
```python
TEST_SENTENCE = "the air conditioning is working"

trans64_args_s = AttrDict()
args_dict = {
    "data_file_name": "pig_latin_small",
    "cuda": True,
    "nepochs": 50,
    "checkpoint_dir": "checkpoints",
    "learning_rate": 5e-4,
    "early_stopping_patience": 20,
    "lr_decay": 0.99,
    "batch_size": 64,
    "hidden_size": 64,  # Increased model size
    "num_transformer_layers": 3,
}
trans64_args_s.update(args_dict)
print_opts(trans64_args_s)

trans64_encoder_s, trans64_decoder_s, trans64_losses_s = train(trans64_args_s)
```

```
translated = translate_sentence(
    TEST_SENTENCE, trans64_encoder_s, trans64_decoder_s, None, trans64_args_s
)
print("source:\t\t{} \ntranslated:\t{}".format(TEST_SENTENCE, translated))
```

```
================================================================================
                                       Opts
--------------------------------------------------------------------------------
                         data_file_name: pig_latin_small
                                   cuda: 1
                                 nepochs: 50
                         checkpoint_dir: checkpoints
                          learning_rate: 0.0005
                early_stopping_patience: 20
                                lr_decay: 0.99
                             batch_size: 64
                            hidden_size: 64
                 num_transformer_layers: 3
================================================================================
================================================================================
                                     Data Stats
--------------------------------------------------------------------------------
('irresistible', 'irresistibleway')
('repentance', 'epentanceray')
('pretty', 'ettypray')
('parcels', 'arcelspay')
('survey', 'urveysay')
Num unique word pairs: 3198
Vocabulary: dict_keys(['-', 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j',
'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z',
'SOS', 'EOS', 'EOP'])
Vocab size: 30
================================================================================
Moved models to GPU!
Epoch:   0 | Train loss: 2.380 | Val loss: 1.933 | Gen: etay iway inday
isssssssssay ooooay-ay-ay-ay-ay-
Epoch:   1 | Train loss: 1.629 | Val loss: 1.647 | Gen: etay irway onday-inday
isway ongray-ay
Epoch:   2 | Train loss: 1.356 | Val loss: 1.775 | Gen: eytay away onday-ingay
isay ongray
Epoch:   3 | Train loss: 1.249 | Val loss: 1.542 | Gen: ethay iway ontinay isway
ongray
Epoch:   4 | Train loss: 1.089 | Val loss: 1.369 | Gen: etay irway ondinday
isway orongay
Epoch:   5 | Train loss: 0.942 | Val loss: 1.192 | Gen: ethay irway
oniondiongiongiongio isway oingray
```

```
Epoch:   6 | Train loss: 0.802 | Val loss: 1.088 | Gen: etay irway ondindingay
isway oringway
Epoch:   7 | Train loss: 0.716 | Val loss: 1.109 | Gen: ethay irway
onindinginginay isway oringway
Epoch:   8 | Train loss: 0.652 | Val loss: 1.180 | Gen: etay ariway
ondioningcingiongcay isway orkingsgay
Epoch:   9 | Train loss: 0.606 | Val loss: 0.924 | Gen: ehay irway
ondioncatingcay isway orkingsay
Epoch:  10 | Train loss: 0.518 | Val loss: 0.943 | Gen: ehay arway onditingcay
isway orkingsay
Epoch:  11 | Train loss: 0.486 | Val loss: 0.984 | Gen: etay arway onditingnay
isway orkingsway
Epoch:  12 | Train loss: 0.437 | Val loss: 0.838 | Gen: ethay arway onditiocay
isway orkingsway
Epoch:  13 | Train loss: 0.366 | Val loss: 0.866 | Gen: ethay arway onditingcay
isway orkingway
Epoch:  14 | Train loss: 0.308 | Val loss: 0.838 | Gen: ethay arrway onditingcay
isway okrkingsway
Epoch:  15 | Train loss: 0.267 | Val loss: 0.900 | Gen: etway arway onditingcay
isway okingray
Epoch:  16 | Train loss: 0.250 | Val loss: 0.933 | Gen: ethay arway onditingcay
isway orkingway
Epoch:  17 | Train loss: 0.244 | Val loss: 0.891 | Gen: ethay arway ondioningcay
isway orkingway
Epoch:  18 | Train loss: 0.233 | Val loss: 0.795 | Gen: ethay arrway
onditiningcay issway orkingnay
Epoch:  19 | Train loss: 0.197 | Val loss: 0.728 | Gen: ethay arway onditingcay
isway orkingnway
Epoch:  20 | Train loss: 0.174 | Val loss: 0.789 | Gen: ethay arway
onditinmingcay isway orkingway
Epoch:  21 | Train loss: 0.180 | Val loss: 0.873 | Gen: ethay arway
ondiontingcay isway okrkingwrway
Epoch:  22 | Train loss: 0.198 | Val loss: 0.750 | Gen: ethay arway onditingcay
isway orkingnway
Epoch:  23 | Train loss: 0.176 | Val loss: 0.905 | Gen: etay arway onditiongcay
isway orkingway
Epoch:  24 | Train loss: 0.191 | Val loss: 0.637 | Gen: ethay arway onditiongcay
isway orkingway
Epoch:  25 | Train loss: 0.131 | Val loss: 0.686 | Gen: ehthay rway onditiongcay
isway orkingway
Epoch:  26 | Train loss: 0.122 | Val loss: 0.664 | Gen: ethay ariway
onditiongcincay isway orkingway
Epoch:  27 | Train loss: 0.092 | Val loss: 0.631 | Gen: ethay ariway
onditiongcay isway okrkingway
Epoch:  28 | Train loss: 0.072 | Val loss: 0.643 | Gen: ethay arway
onditioningcay isway orkingsway
Epoch:  29 | Train loss: 0.063 | Val loss: 0.598 | Gen: ethay ariway
onditiongcay isway okirngway
```

```
Epoch:  30 | Train loss: 0.054 | Val loss: 0.589 | Gen: ethay ariway
onditioningcay isway okirngway
Epoch:  31 | Train loss: 0.051 | Val loss: 0.647 | Gen: ethay ariway
onditiongcay isway okingray
Epoch:  32 | Train loss: 0.058 | Val loss: 0.642 | Gen: ethay arrway
onditioningcay isway okrkingway
Epoch:  33 | Train loss: 0.050 | Val loss: 0.655 | Gen: etay ariway
onditioningcay isway okrayngway
Epoch:  34 | Train loss: 0.050 | Val loss: 0.665 | Gen: ethay riway onditingcay
isway orkingway
Epoch:  35 | Train loss: 0.042 | Val loss: 0.593 | Gen: ethay arway
onditioningcay isway okrayngway
Epoch:  36 | Train loss: 0.073 | Val loss: 1.141 | Gen: ethay arway
onditiigonigcay isay orkinggway
Epoch:  37 | Train loss: 0.293 | Val loss: 0.876 | Gen: ehthay irway
onditiningnay isway okinringway
Epoch:  38 | Train loss: 0.220 | Val loss: 0.759 | Gen: ethay arway
ondioniongcay isway orkingway
Epoch:  39 | Train loss: 0.127 | Val loss: 0.555 | Gen: ethay arway onditionday
isway okingray
Epoch:  40 | Train loss: 0.079 | Val loss: 0.526 | Gen: ethay ariway
onditiongingcay isway okingray
Epoch:  41 | Train loss: 0.046 | Val loss: 0.571 | Gen: ethay ariway
onditiongcay isway okingray
Epoch:  42 | Train loss: 0.038 | Val loss: 0.441 | Gen: ethay arway onditiongcay
isway okingray
Epoch:  43 | Train loss: 0.024 | Val loss: 0.480 | Gen: ethay ariway
onditiongcay isway okingray
Epoch:  44 | Train loss: 0.020 | Val loss: 0.462 | Gen: ethay ariway
onditiongcay isway okingray
Epoch:  45 | Train loss: 0.017 | Val loss: 0.492 | Gen: ethay ariway
onditiongcay isway okingray
Epoch:  46 | Train loss: 0.015 | Val loss: 0.474 | Gen: ethay arway
onditioningcay isway okingwray
Epoch:  47 | Train loss: 0.013 | Val loss: 0.503 | Gen: ethay ariway
onditiongcay isway okingray
Epoch:  48 | Train loss: 0.012 | Val loss: 0.514 | Gen: ethay ariway
onditiongcay isway okingray
Epoch:  49 | Train loss: 0.013 | Val loss: 0.610 | Gen: ethay arway onditiongcay
way okrkingway
Obtained lowest validation loss of: 0.4412691276087317
source:         the air conditioning is working
translated:     ethay arway onditiongcay way okrkingway
```

[ ]:
```
TEST_SENTENCE = "the air conditioning is working"

trans64_args_l = AttrDict()
```

```python
args_dict = {
    "data_file_name": "pig_latin_large",  # Increased data set size
    "cuda": True,
    "nepochs": 50,
    "checkpoint_dir": "checkpoints",
    "learning_rate": 5e-4,
    "early_stopping_patience": 20,
    "lr_decay": 0.99,
    "batch_size": 512,
    "hidden_size": 64,  # Increased model size
    "num_transformer_layers": 3,
}
trans64_args_l.update(args_dict)
print_opts(trans64_args_l)

trans64_encoder_l, trans64_decoder_l, trans64_losses_l = train(trans64_args_l)

translated = translate_sentence(
    TEST_SENTENCE, trans64_encoder_l, trans64_decoder_l, None, trans64_args_l
)
print("source:\t\t{} \ntranslated:\t{}".format(TEST_SENTENCE, translated))
```

```
================================================================================
                                       Opts
--------------------------------------------------------------------------------
                         data_file_name: pig_latin_large
                                   cuda: 1
                                 nepochs: 50
                         checkpoint_dir: checkpoints
                          learning_rate: 0.0005
                early_stopping_patience: 20
                                lr_decay: 0.99
                              batch_size: 512
                             hidden_size: 64
                 num_transformer_layers: 3
================================================================================
================================================================================
                                    Data Stats
--------------------------------------------------------------------------------
('tribune', 'ibunetray')
('sheriff', 'eriffshay')
('sobbed', 'obbedsay')
('monte', 'ontemay')
('ere', 'ereway')
Num unique word pairs: 22402
Vocabulary: dict_keys(['-', 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j',
'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z',
```

```
'SOS', 'EOS', 'EOP'])
Vocab size: 30
================================================================================
Moved models to GPU!
Epoch:   0 | Train loss: 2.517 | Val loss: 2.084 | Gen: omay ay ooooonsay-
oooontay iisisisisisisssssss ooooooooooooooay
Epoch:   1 | Train loss: 1.770 | Val loss: 1.780 | Gen: otay ay-iririririray
ontinsay-ontay issay-isisay ootay-otay
Epoch:   2 | Train loss: 1.483 | Val loss: 1.729 | Gen: etete teday aiiririiway
ontintintintintitint isissisisssay oooutingray
Epoch:   3 | Train loss: 1.293 | Val loss: 1.786 | Gen: tay-eday airirararararay
oncicay-onincandiond issay-isisisay ooorrrrray-irrrrrrrr
Epoch:   4 | Train loss: 1.185 | Val loss: 1.587 | Gen: eteteay aiway onticay-
odingay isssay ooogrgay-ongray
Epoch:   5 | Train loss: 1.038 | Val loss: 1.354 | Gen: tay-tay away ondicay-
ongngngay issay ororgway-ay
Epoch:   6 | Train loss: 0.913 | Val loss: 1.347 | Gen: ethay away
ondgingingngngay isay orgry
Epoch:   7 | Train loss: 0.835 | Val loss: 1.314 | Gen: etay arway
onditingtingday isay orway-rgway
Epoch:   8 | Train loss: 0.760 | Val loss: 1.269 | Gen: ethay iwway
ondicininincay isway orwway
Epoch:   9 | Train loss: 0.687 | Val loss: 1.253 | Gen: ethay iray onitininday
isay orway-gway
Epoch:  10 | Train loss: 0.635 | Val loss: 1.212 | Gen: ethay aiway onciticincay
isay orkingay
Epoch:  11 | Train loss: 0.547 | Val loss: 0.993 | Gen: ethway arway
onditiningngngcay isay orkingngway
Epoch:  12 | Train loss: 0.520 | Val loss: 1.284 | Gen: etay arway
onititinntingay isay orkingnay
Epoch:  13 | Train loss: 0.518 | Val loss: 0.984 | Gen: ethay arway
onditinctingday isway orkingngway
Epoch:  14 | Train loss: 0.404 | Val loss: 0.836 | Gen: ethay arway
onditioniningcay isay orkingnay
Epoch:  15 | Train loss: 0.346 | Val loss: 0.741 | Gen: ethay arway
onditiningdingay isay orkingway
Epoch:  16 | Train loss: 0.319 | Val loss: 0.840 | Gen: ethay arway
onditioningcay iisway orkingway
Epoch:  17 | Train loss: 0.298 | Val loss: 0.657 | Gen: ethay airway
onditioningway isay orkingway
Epoch:  18 | Train loss: 0.261 | Val loss: 0.846 | Gen: ehay arway
onditiningdincay isway orkingway
Epoch:  19 | Train loss: 0.253 | Val loss: 0.654 | Gen: ehtay airway
oniditioningcay isway orkingway
Epoch:  20 | Train loss: 0.229 | Val loss: 0.647 | Gen: ethay airway
onditiningday isway orkingway
Epoch:  21 | Train loss: 0.195 | Val loss: 0.676 | Gen: ehay airway
onditioningcay isay orkingway
```

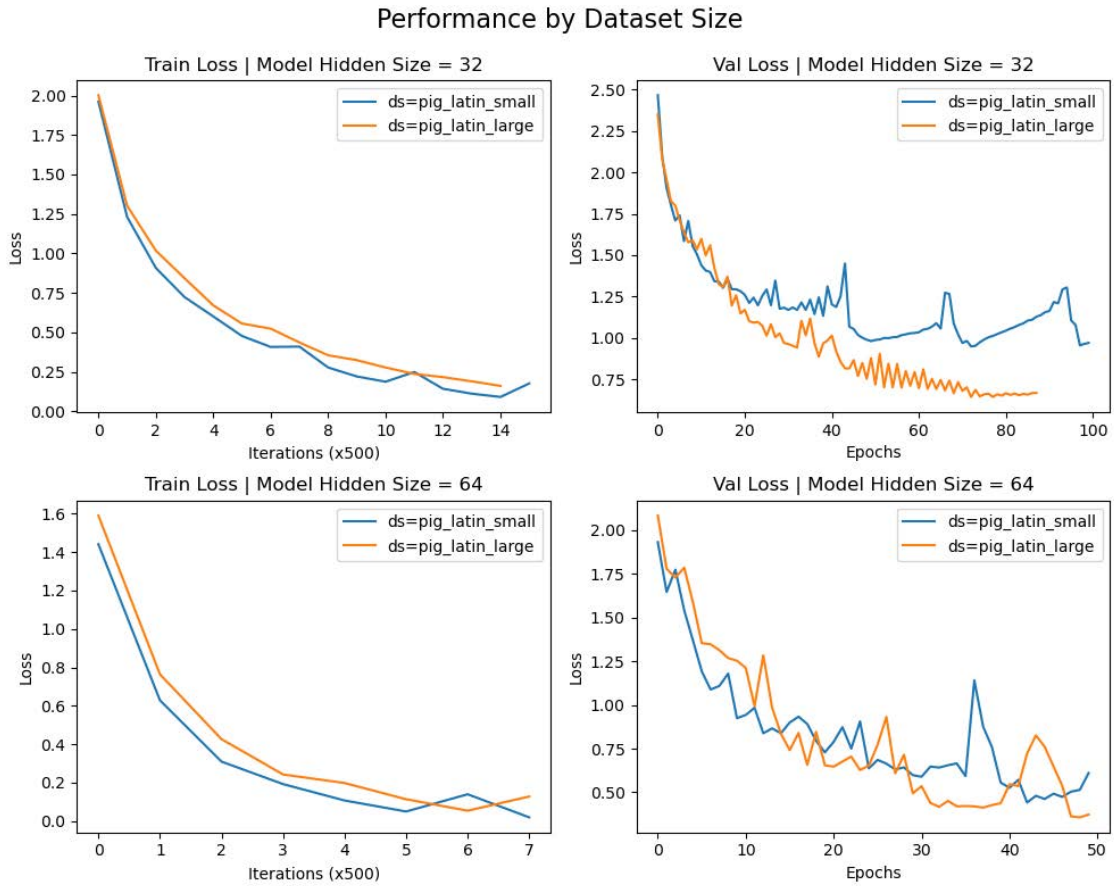
```
Epoch:  22 | Train loss: 0.202 | Val loss: 0.705 | Gen: ethay airway
onditiningcay isway okingingway
Epoch:  23 | Train loss: 0.207 | Val loss: 0.628 | Gen: ehthay airway
onditiningcay isway orkingway
Epoch:  24 | Train loss: 0.178 | Val loss: 0.652 | Gen: ehay airway
onditionngscay isway orkingway
Epoch:  25 | Train loss: 0.198 | Val loss: 0.773 | Gen: ehtay airway
onditiongincay isway orkingway
Epoch:  26 | Train loss: 0.201 | Val loss: 0.931 | Gen: ehay airway
onditiningdcccccay iisway orkingway
Epoch:  27 | Train loss: 0.232 | Val loss: 0.609 | Gen: ethay ariway
onditinongnaycay isway orkingway
Epoch:  28 | Train loss: 0.173 | Val loss: 0.714 | Gen: ethay irway
onditioningcayaaay isay orkingway
Epoch:  29 | Train loss: 0.168 | Val loss: 0.494 | Gen: ethay airway
onditioningcay isway orkingway
Epoch:  30 | Train loss: 0.116 | Val loss: 0.535 | Gen: ehtay airway
onditionningcay isway orkingway
Epoch:  31 | Train loss: 0.103 | Val loss: 0.439 | Gen: ehtay airway
onditinongcay isway orkingway
Epoch:  32 | Train loss: 0.083 | Val loss: 0.417 | Gen: ehthay airway
onditioningcay isway orkingway
Epoch:  33 | Train loss: 0.070 | Val loss: 0.451 | Gen: ehthay airway
onditioningcay isway orkingway
Epoch:  34 | Train loss: 0.066 | Val loss: 0.420 | Gen: ehtay airway
onditioningcay isway orkingway
Epoch:  35 | Train loss: 0.057 | Val loss: 0.421 | Gen: ehtay airway
onditioningcay isway orkingway
Epoch:  36 | Train loss: 0.056 | Val loss: 0.419 | Gen: ehtay airway
onditioningcay isway orkingway
Epoch:  37 | Train loss: 0.052 | Val loss: 0.412 | Gen: ehtay airway
onditioningcay isway orkingway
Epoch:  38 | Train loss: 0.049 | Val loss: 0.427 | Gen: ehtay airway
onditioningcay isway orkingway
Epoch:  39 | Train loss: 0.046 | Val loss: 0.437 | Gen: ehtay airwayway
onditiniongcay isway orkingway
Epoch:  40 | Train loss: 0.052 | Val loss: 0.546 | Gen: ethay airway
onditionginay isway orkingway
Epoch:  41 | Train loss: 0.057 | Val loss: 0.536 | Gen: ehtay airway
onditioningcay isway orkingway
Epoch:  42 | Train loss: 0.104 | Val loss: 0.722 | Gen: ehthay airway
onditiongncay isway orkingway
Epoch:  43 | Train loss: 0.213 | Val loss: 0.826 | Gen: ethay airway
onditionningscay isway oorkingway
Epoch:  44 | Train loss: 0.203 | Val loss: 0.761 | Gen: etay irway onditiongscay
isway orkingway
Epoch:  45 | Train loss: 0.167 | Val loss: 0.649 | Gen: ethay airway
onditinoningcay isay orkingway
```

```
Epoch:  46 | Train loss: 0.126 | Val loss: 0.538 | Gen: ethay airway
onditioningcay isway orkingway
Epoch:  47 | Train loss: 0.082 | Val loss: 0.362 | Gen: ehthay airway
onditioningcay isway orkingway
Epoch:  48 | Train loss: 0.063 | Val loss: 0.357 | Gen: ehthay airway
onditioningcay isway orkingway
Epoch:  49 | Train loss: 0.049 | Val loss: 0.373 | Gen: ethay airway
onditioningcay isway orkingway
Obtained lowest validation loss of: 0.35659549697967513
source:         the air conditioning is working
translated:     ethay airway onditioningcay isway orkingway
```

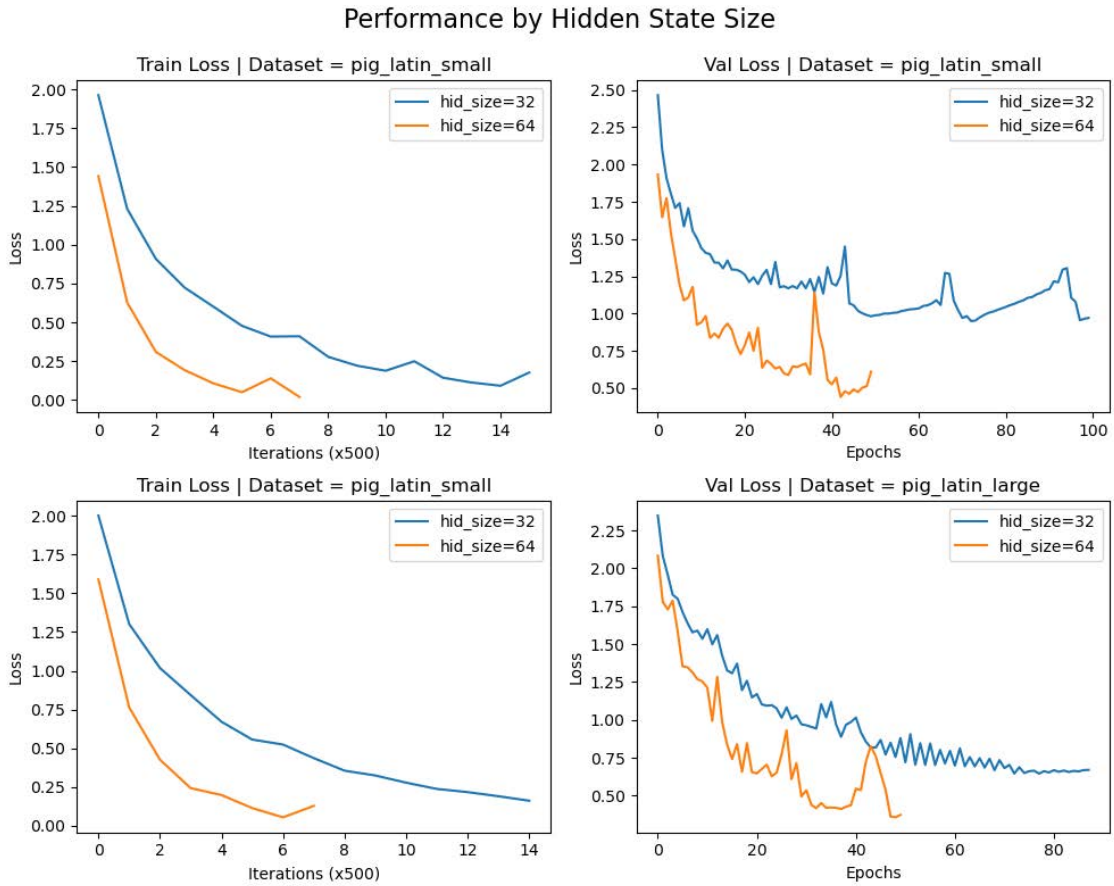
The following cell generates two loss plots. In the first plot, we compare the effects of increasing dataset size. In the second plot, we compare the effects of increasing model size. Include both plots in your report, and include your analysis of the results.

[ ]:
```
save_loss_comparison_by_dataset(
    trans32_losses_s,
    trans32_losses_l,
    trans64_losses_s,
    trans64_losses_l,
    trans32_args_s,
    trans32_args_l,
    trans64_args_s,
    trans64_args_l,
    "trans_by_dataset",
)
save_loss_comparison_by_hidden(
    trans32_losses_s,
    trans32_losses_l,
    trans64_losses_s,
    trans64_losses_l,
    trans32_args_s,
    trans32_args_l,
    trans64_args_s,
    trans64_args_l,
    "trans_by_hidden",
)
```

```
<Figure size 640x480 with 0 Axes>
```




```
<Figure size 640x480 with 0 Axes>
```




# 5 Part 2: Decoder-only NMT

## 5.1 Training and evaluation code

[ ]:
```
# Get Decoder Params
def transformer_params(vocab_size, hidden_size, num_layers, max_seq_len=1000):
    """ Given transformerEncoder config calculate total number of parameters """
    # token and position embeddings
    embeddings = vocab_size * hidden_size + max_seq_len * hidden_size
    # attention
    self_attention = 3 * hidden_size**2 + 3 * hidden_size
    attention_mlp = hidden_size**2 + hidden_size
    # out
    out = hidden_size * vocab_size + vocab_size
    # note: embeddings are not included in the param count!
    total_params = num_layers * (self_attention + attention_mlp) + out
    return total_params
```

```
# Let us use a factor of 2 to describe the multiply accumulate cost.
# See more details in https://arxiv.org/pdf/2203.15556.pdf Appendix F
# Get Decoder Flops
def transformer_flops(seq_len, vocab_size, hidden_size, num_layers):
    """ Given AttentionEncoder config calculate total number of parameters """
    # token and position embeddings
    embeddings = 2 * seq_len * vocab_size * hidden_size
    # attention
    # key, query, value projections
    attention = 2 * 3 * seq_len * hidden_size * hidden_size
    # key @ query logits
    attlogits = 2 * seq_len * seq_len * hidden_size
    # softmax
    attsoftmax = 3 * seq_len * seq_len # 3 * is for subtract (max), exp, divide␣
 ↪(?)
    # softmax @ value reductions
    attvalue = 2 * seq_len * seq_len * hidden_size
    # mlp
    attlinear = 2 * seq_len * hidden_size * hidden_size
    att = attention + attlogits + attsoftmax + attvalue + attlinear

    # out
    out = 2 * seq_len * vocab_size * hidden_size

    # note: we ignore the embedding for now
    forward_flops = num_layers * att + out
    backward_flops = 2 * forward_flops # as in Kaplan et al. 2020
    total_flops = forward_flops + backward_flops
    return total_flops

def string_to_index_list(s, char_to_index, end_token=None):
    """Converts a sentence into a list of indexes (for each character)."""
    # return [char_to_index[char] for char in s] + [
    #     end_token
    # ]  # Adds the end token to each index list

    if end_token is not None:
        return [char_to_index[char] for char in s] + [
            end_token
        ]  # Adds the end token to each index list
    else:
        return [char_to_index[char] for char in s]


def translate_sentence(sentence, decoder, idx_dict, opts):
    """Translates a sentence from English to Pig-Latin, by splitting the␣
 ↪sentence into
```

```
    words (whitespace-separated), running the decoder model to translate each
    word independently, and then stitching the words back together with spaces␣
↪between them.
    """
    if idx_dict is None:
        line_pairs, vocab_size, idx_dict = load_data(opts["data_file_name"])
    return " ".join(
        [translate(word, decoder, idx_dict, opts) for word in sentence.split()]
    )


def translate(input_string, decoder, idx_dict, opts):
    """Translates a given string from English to Pig-Latin."""
    char_to_index = idx_dict["char_to_index"]
    index_to_char = idx_dict["index_to_char"]
    start_token = idx_dict["start_token"]
    end_token = idx_dict["end_token"]
    end_prompt_token = idx_dict["end_prompt_token"]

    max_generated_chars = 20
    gen_string = ""

    indexes = string_to_index_list(input_string, char_to_index,␣
↪end_prompt_token)
    indexes = to_var(
        torch.LongTensor(indexes).unsqueeze(0), opts.cuda
    )  # Unsqueeze to make it like BS = 1

    # decoder_hidden = encoder_last_hidden
    decoder_input = to_var(torch.LongTensor([[start_token]]), opts.cuda)  # For␣
↪BS = 1
    decoder_input = torch.cat([decoder_input, indexes], dim=1)

    decoder_inputs = decoder_input
    out_start_idx = len(decoder_inputs[0]) - 1
    for i in range(max_generated_chars):
        ## slow decoding, recompute everything at each time
        decoder_outputs, attention_weights = decoder(
            decoder_inputs
        )

        generated_words_all = F.softmax(decoder_outputs, dim=2).max(2)[1]
        generated_words = generated_words_all[:, out_start_idx:]
        ni = generated_words.cpu().numpy().reshape(-1)  # LongTensor of size 1
        ni = ni[-1]  # latest output token

        decoder_inputs = torch.cat([decoder_input, generated_words], dim=1)
```

```
        if ni == end_token:
            break
        else:
            gen_string = "".join(
                [
                    index_to_char[int(item)]
                    for item in generated_words.cpu().numpy().reshape(-1)
                ]
            )

    return gen_string

def visualize_attention(input_string, encoder, decoder, idx_dict, opts):
    """Generates a heatmap to show where attention is focused in each decoder␣
 ↪step."""
    if idx_dict is None:
        line_pairs, vocab_size, idx_dict = load_data(opts["data_file_name"])
    char_to_index = idx_dict["char_to_index"]
    index_to_char = idx_dict["index_to_char"]
    start_token = idx_dict["start_token"]
    end_token = idx_dict["end_token"]

    max_generated_chars = 20
    gen_string = ""

    indexes = string_to_index_list(input_string, char_to_index, end_token)
    indexes = to_var(
        torch.LongTensor(indexes).unsqueeze(0), opts.cuda
    )  # Unsqueeze to make it like BS = 1

    encoder_annotations, encoder_hidden = encoder(indexes)

    decoder_hidden = encoder_hidden
    decoder_input = to_var(torch.LongTensor([[start_token]]), opts.cuda)  # For␣
 ↪BS = 1
    decoder_inputs = decoder_input

    produced_end_token = False

    for i in range(max_generated_chars):
        ## slow decoding, recompute everything at each time
        decoder_outputs, attention_weights = decoder(
            decoder_inputs, encoder_annotations, decoder_hidden
        )
        generated_words = F.softmax(decoder_outputs, dim=2).max(2)[1]
        ni = generated_words.cpu().numpy().reshape(-1)  # LongTensor of size 1
```

```
        ni = ni[-1]  # latest output token

        decoder_inputs = torch.cat([decoder_input, generated_words], dim=1)

        if ni == end_token:
            break
        else:
            gen_string = "".join(
                [
                    index_to_char[int(item)]
                    for item in generated_words.cpu().numpy().reshape(-1)
                ]
            )

    if isinstance(attention_weights, tuple):
        ## transformer's attention mweights
        attention_weights, self_attention_weights = attention_weights

    all_attention_weights = attention_weights.data.cpu().numpy()

    for i in range(len(all_attention_weights)):
        attention_weights_matrix = all_attention_weights[i].squeeze()
        fig = plt.figure()
        ax = fig.add_subplot(111)
        cax = ax.matshow(attention_weights_matrix, cmap="bone")
        fig.colorbar(cax)

        # Set up axes
        ax.set_yticklabels([""] + list(input_string) + ["EOS"], rotation=90)
        ax.set_xticklabels(
            [""] + list(gen_string) + (["EOS"] if produced_end_token else [])
        )

        # Show label at every tick
        ax.xaxis.set_major_locator(ticker.MultipleLocator(1))
        ax.yaxis.set_major_locator(ticker.MultipleLocator(1))
        # Add title
        plt.xlabel("Attention weights to the source sentence in layer {}".
 ↪format(i + 1))
        plt.tight_layout()
        plt.grid("off")
        plt.show()

    return gen_string

def compute_loss(data_dict, decoder, idx_dict, criterion, optimizer, opts):
    """Train/Evaluate the model on a dataset.
```

```
    Arguments:
        data_dict: The validation/test word pairs, organized by source and␣
↪target lengths.
        decoder: A decoder model (with or without attention) to generate output␣
↪tokens.
        idx_dict: Contains char-to-index and index-to-char mappings, and start␣
↪& end token indexes.
        criterion: Used to compute the CrossEntropyLoss for each decoder output.
        optimizer: Train the weights if an optimizer is given. None if only␣
↪evaluate the model.
        opts: The command-line arguments.

    Returns:
        mean_loss: The average loss over all batches from data_dict.
    """
    start_token = idx_dict["start_token"]
    end_token = idx_dict["end_token"]
    end_prompt_token = idx_dict["end_prompt_token"]
    char_to_index = idx_dict["char_to_index"]

    losses = []
    num_tokens = 0
    num_flops = 0
    num_example = 0
    for key in data_dict:
        input_strings, target_strings = zip(*data_dict[key])
        input_tensors = [
            torch.LongTensor(string_to_index_list(s, char_to_index,␣
↪end_prompt_token))
            for s in input_strings
        ]
        target_tensors = [
            torch.LongTensor(string_to_index_list(s, char_to_index, end_token))
            for s in target_strings
        ]

        num_tensors = len(input_tensors)
        num_batches = int(np.ceil(num_tensors / float(opts.batch_size)))

        for i in range(num_batches):
            start = i * opts.batch_size
            end = start + opts.batch_size

            src_EOP = to_var(torch.stack(input_tensors[start:end]), opts.cuda)
            tgt_EOS = to_var(torch.stack(target_tensors[start:end]), opts.cuda)
            tgt_EOS_len = tgt_EOS.size(1)
```

```
            SOS_src_EOP_tgt, src_EOP_tgt_EOS =␣
↪generate_tensors_for_training_decoder_nmt(src_EOP, tgt_EOS, start_token,␣
↪opts.cuda)

            predict_src_EOP_tgt_EOS, attention_weights =␣
↪decoder(SOS_src_EOP_tgt)

            predict_tgt_EOS = predict_src_EOP_tgt_EOS[:, -tgt_EOS_len:, :]
            predict_tgt_EOS_flatten = predict_tgt_EOS.reshape(-1,␣
↪predict_tgt_EOS.size(2))
            tgt_EOS_flatten = tgt_EOS.view(-1)
            loss = criterion(predict_tgt_EOS_flatten, tgt_EOS_flatten)

            losses.append(loss.item())

            # Collect some statistics
            BS = src_EOP.size(0)
            i_seq_len = SOS_src_EOP_tgt.size(1)
            num_tokens += BS * i_seq_len
            num_flops += BS * transformer_flops(i_seq_len, opts.vocab_size,␣
↪opts.hidden_size, opts.num_layers)
            num_example += BS

            ## training if an optimizer is provided
            if optimizer:
                # Zero gradients
                optimizer.zero_grad()
                # Compute gradients
                loss.backward()
                # Update the parameters of the encoder and decoder
                optimizer.step()

    return losses, num_tokens, num_flops, num_example


def training_loop(
    train_dict, val_dict, idx_dict, decoder, criterion, optimizer, opts
):
    """Runs the main training loop; evaluates the model on the val set every␣
↪epoch.
        * Prints training and val loss each epoch.
        * Prints qualitative translation results each epoch using TEST_SENTENCE
        * Saves an attention map for TEST_WORD_ATTN each epoch
        * Returns loss curves for comparison

    Arguments:
```

```
        train_dict: The training word pairs, organized by source and target␣
↪lengths.
        val_dict: The validation word pairs, organized by source and target␣
↪lengths.
        idx_dict: Contains char-to-index and index-to-char mappings, and start␣
↪& end token indexes.
        decoder: A decoder model (with or without attention) to generate output␣
↪tokens.
        criterion: Used to compute the CrossEntropyLoss for each decoder output.
        optimizer: Implements a step rule to update the parameters of the␣
↪encoder and decoder.
        opts: The command-line arguments.

    Returns:
        losses: Lists containing training and validation loss curves.
    """

    start_token = idx_dict["start_token"]
    end_token = idx_dict["end_token"]
    char_to_index = idx_dict["char_to_index"]

    loss_log = open(os.path.join(opts.checkpoint_path, "loss_log.txt"), "w")

    best_val_loss = 1e6
    train_losses = []
    val_losses = []

    mean_train_losses = []
    mean_val_losses = []

    early_stopping_counter = 0

    num_tokens = []
    num_flops = []
    num_examples = []
    num_params = []

    total_tokens = 0
    total_flops = 0
    total_examples = 0
    params_count = transformer_params(opts.vocab_size, opts.hidden_size, opts.
↪num_layers)


    for epoch in range(opts.nepochs):
        optimizer.param_groups[0]["lr"] *= opts.lr_decay
```

```
        train_loss, num_token, num_flop, num_example = compute_loss(
            train_dict, decoder, idx_dict, criterion, optimizer, opts
        )
        val_loss, _, _, _ = compute_loss(
            val_dict, decoder, idx_dict, criterion, None, opts
        )

        mean_train_loss = np.mean(train_loss)
        mean_val_loss = np.mean(val_loss)

        if mean_val_loss < best_val_loss:
            checkpoint(decoder, idx_dict, opts)
            best_val_loss = mean_val_loss
            early_stopping_counter = 0
        else:
            early_stopping_counter += 1

        if early_stopping_counter > opts.early_stopping_patience:
            print(
                "Validation loss has not improved in {} epochs, stopping early".
↪format(
                    opts.early_stopping_patience
                )
            )
            print("Obtained lowest validation loss of: {}".
↪format(best_val_loss))
            return (mean_train_losses, mean_val_losses, num_tokens, num_flops,␣
↪num_examples, num_params)

        gen_string = translate_sentence(TEST_SENTENCE, decoder, idx_dict, opts)
        total_tokens += num_token
        total_flops += num_flop
        total_examples += num_example

        print(
            "Epoch: {:3d} | Flops (T): {}, Tokens (M) {}, Train Loss {:.3f},␣
↪Val Loss {:.3f} | Gen: {:20s}".format(
                epoch, total_flops/1e12, total_tokens/1e6,
                mean_train_loss, mean_val_loss, gen_string)
        )

        loss_log.write("{} {} {}\n".format(epoch, train_loss, val_loss))
        loss_log.flush()

        train_losses += train_loss
        val_losses += val_loss
```

```
        mean_train_losses.append(mean_train_loss)
        mean_val_losses.append(mean_val_loss)
        num_tokens.append(total_tokens)
        num_flops.append(total_flops)
        num_examples.append(total_examples)
        num_params.append(params_count)

    print("Obtained lowest validation loss of: {}".format(best_val_loss))
    print("Avg tokens per example (Seq Len): {}, Avg flops per example: {}".
 ↪format(
        total_tokens/total_examples, total_flops/total_examples))
    return (mean_train_losses, mean_val_losses, num_tokens, num_flops,␣
 ↪num_examples, num_params)

def train(opts):
    line_pairs, vocab_size, idx_dict = load_data(opts["data_file_name"])
    print_data_stats(line_pairs, vocab_size, idx_dict)

    # Split the line pairs into an 80% train and 20% val split
    num_lines = len(line_pairs)
    num_train = int(0.8 * num_lines)
    train_pairs, val_pairs = line_pairs[:num_train], line_pairs[num_train:]

    # Group the data by the lengths of the source and target words, to form␣
 ↪batches
    train_dict = create_dict(train_pairs)
    val_dict = create_dict(val_pairs)
    print(train_dict)

    ##########################################################################
    ### Setup: Create Decoder, Learning Criterion, and Optimizers ###
    ##########################################################################
    decoder = TransformerDecoder(
        vocab_size=vocab_size,
        hidden_size=opts.hidden_size,
        num_layers=opts.num_transformer_layers,
        loss_seq_span=opts.loss_seq_span
    )

    #### setup checkpoint path
    model_name = "decoder-only-h{}-bs{}-{}".format(
        opts.hidden_size, opts.batch_size, opts.data_file_name
    )
    opts.checkpoint_path = model_name
    create_dir_if_not_exists(opts.checkpoint_path)
    ####
```

```
    if opts.cuda:
        decoder.cuda()
        print("Moved models to GPU!")

    criterion = nn.CrossEntropyLoss()
    optimizer = optim.Adam(
        list(decoder.parameters()), lr=opts.learning_rate
    )

    try:
        losses = training_loop(
            train_dict, val_dict, idx_dict, decoder, criterion, optimizer, opts
        )
    except KeyboardInterrupt:
        print("Exiting early from training.")
        return decoder, losses

    return decoder, losses, idx_dict


def print_data_stats(line_pairs, vocab_size, idx_dict):
    """Prints example word pairs, the number of data points, and the vocabulary.
 ↪"""
    print("=" * 80)
    print("Data Stats".center(80))
    print("-" * 80)
    for pair in line_pairs[:5]:
        print(pair)
    print("Num unique word pairs: {}".format(len(line_pairs)))
    print("Vocabulary: {}".format(idx_dict["char_to_index"].keys()))
    print("Vocab size: {}".format(vocab_size))
    print("=" * 80)


def print_opts(opts):
    """Prints the values of all command-line arguments."""
    print("=" * 80)
    print("Opts".center(80))
    print("-" * 80)
    for key in opts.__dict__:
        print("{:>30}: {:<30}".format(key, opts.__dict__[key]).center(80))
    print("=" * 80)


def checkpoint(decoder, idx_dict, opts):
    """Saves the decoder model, along with idx_dict, which
```

```
    contains the char_to_index and index_to_char mappings, and the start_token
    and end_token values.
    """
    with open(os.path.join(opts.checkpoint_path, "decoder.pt"), "wb") as f:
        torch.save(decoder, f)

    with open(os.path.join(opts.checkpoint_path, "idx_dict.pkl"), "wb") as f:
        pkl.dump(idx_dict, f)
```

### 5.2 Step 1: Construct the input tensors and the target tensors for training a decoder

Implement the function below by following the three steps mentioned.

```
[ ]: def generate_tensors_for_training_decoder_nmt(src_EOP, tgt_EOS, start_token,␣
 ↪cuda):

    # Step1: concatenate input_EOP, and target_EOS vectors to form a target␣
 ↪tensor.
    src_EOP_tgt_EOS = torch.cat((src_EOP, tgt_EOS), dim=1)

    # Step2: make a sos vector
    sos_vector = torch.full((src_EOP.size(0), 1), start_token, dtype=torch.long)

    # Check if GPU is available and move the tensor to GPU if specified
    sos_vector = to_var(sos_vector, cuda)

    # Step3: make a concatenated input tensor to the decoder-only NMT (format:␣
 ↪Start-of-token source end-of-prompt target)
    SOS_src_EOP_tgt = torch.cat((sos_vector, src_EOP, tgt_EOS), dim=1)

    return SOS_src_EOP_tgt, src_EOP_tgt_EOS
```

### 5.3 Step 2: Implement decoder forward pass

The following cell defines the decoder only transformer. You can check the forward function implementation from the encoder-decoder model ot get an idea of how to implement this.

```
[ ]: class TransformerDecoder(nn.Module):
    def __init__(self, vocab_size, hidden_size, num_layers, loss_seq_span):
        super(TransformerDecoder, self).__init__()
        self.vocab_size = vocab_size
        self.hidden_size = hidden_size

        self.embedding = nn.Embedding(vocab_size, hidden_size)
        self.num_layers = num_layers
```

```
        self.self_attentions = nn.ModuleList(
            [
                CausalScaledDotAttention(
                    hidden_size=hidden_size,
                )
                for i in range(self.num_layers)
            ]
        )

        self.attention_mlps = nn.ModuleList(
            [
                nn.Sequential(
                    nn.Linear(hidden_size, hidden_size),
                    nn.ReLU(),
                )
                for i in range(self.num_layers)
            ]
        )
        self.out = nn.Linear(hidden_size, vocab_size)

        self.positional_encodings = self.create_positional_encodings()
        self.loss_seq_span = loss_seq_span


    def forward(self, inputs):
        """Forward pass of the attention-based decoder RNN.

        Arguments:
            inputs: Input token indexes across a batch for all the time step.␣
↪(batch_size x decoder_seq_len)
        Returns:
            output: Un-normalized scores for each token in the vocabulary,␣
↪across a batch for all the decoding time steps. (batch_size x␣
↪decoder_seq_len x vocab_size)
            attentions: The stacked attention weights applied to the encoder␣
↪annotations (batch_size x encoder_seq_len x decoder_seq_len)
        """
        # Embed input tokens
        embedded_inputs = self.embedding(inputs)  # (batch_size x␣
↪decoder_seq_len x hidden_size)

        # Add positional encodings
        embedded_inputs += self.positional_encodings[: embedded_inputs.size(1),␣
↪:].unsqueeze(0)

        x = embedded_inputs
```

```
        self_attention_weights_list = []

        for i in range(self.num_layers):
            # Apply self-attention
            x, self_attention_weights = self.self_attentions[i](x, x, x)
            self_attention_weights_list.append(self_attention_weights)

            # Apply MLP
            x = x + self.attention_mlps[i](x)

        # Stack attention weights across all layers
        self_attention_weights = torch.stack(self_attention_weights_list,␣
↪dim=1)  # (batch_size x num_layers x decoder_seq_len x decoder_seq_len)

        # Output layer
        output = self.out(x)  # (batch_size x decoder_seq_len x vocab_size)

        return output, self_attention_weights


    def create_positional_encodings(self, max_seq_len=1000):
        """Creates positional encodings for the inputs.

        Arguments:
            max_seq_len: a number larger than the maximum string length we␣
↪expect to encounter during training

        Returns:
            pos_encodings: (max_seq_len, hidden_dim) Positional encodings for a␣
↪sequence with length max_seq_len.
        """
        pos_indices = torch.arange(max_seq_len)[..., None]
        dim_indices = torch.arange(self.hidden_size // 2)[None, ...]
        exponents = (2 * dim_indices).float() / (self.hidden_size)
        trig_args = pos_indices / (10000 ** exponents)
        sin_terms = torch.sin(trig_args)
        cos_terms = torch.cos(trig_args)

        pos_encodings = torch.zeros((max_seq_len, self.hidden_size))
        pos_encodings[:, 0::2] = sin_terms
        pos_encodings[:, 1::2] = cos_terms

        if torch.cuda.is_available():
            pos_encodings = pos_encodings.cuda()

        return pos_encodings
```

### 5.4 Step 3: Train the decoder-only model

Run the following block to see the training and validation loss of a decoder-only transformer. How is the quality of the generated translation?

[ ]:
```
TEST_SENTENCE = "the air conditioning is working"
trans_args_s = AttrDict()
args_dict = {
    "data_file_name": "pig_latin_small", #pig_latin_small debug
    "cuda": use_cuda,
    "nepochs": 100,
    "checkpoint_dir": "checkpoints",
    "learning_rate": 5e-4,
    "early_stopping_patience": 100,
    "lr_decay": 0.99,
    "batch_size": 64,
    "hidden_size": 128,
    "num_transformer_layers": 4,
    "loss_seq_span": "target_only",
    "num_layers": 4,
    "vocab_size": 30
}
trans_args_s.update(args_dict)
print_opts(trans_args_s)
trans_decoder_s, trans_losses_s, idx_dict = train(trans_args_s)
```

```
================================================================================
                                       Opts
--------------------------------------------------------------------------------
                         data_file_name: pig_latin_small
                                   cuda: 1
                               nepochs: 100
                          checkpoint_dir: checkpoints
                           learning_rate: 0.0005
                 early_stopping_patience: 100
                                lr_decay: 0.99
                              batch_size: 64
                             hidden_size: 128
                  num_transformer_layers: 4
                           loss_seq_span: target_only
                              num_layers: 4
                              vocab_size: 30
================================================================================
================================================================================
                                     Data Stats
--------------------------------------------------------------------------------
('irresistible', 'irresistibleway')
('repentance', 'epentanceray')
```

```
('pretty', 'ettypray')
('parcels', 'arcelspay')
('survey', 'urveysay')
Num unique word pairs: 3198
Vocabulary: dict_keys(['-', 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j',
'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z',
'SOS', 'EOS', 'EOP'])
Vocab size: 30
================================================================================
defaultdict(<class 'list'>, {(12, 15): [('irresistible', 'irresistibleway'),
('advantageous', 'advantageousway'), ('impartiality', 'impartialityway'),
('interruption', 'interruptionway'), ('anticipation', 'anticipationway'),
('impenetrable', 'impenetrableway'), ('excellencies', 'excellenciesway'),
('intermission', 'intermissionway'), ('interspersed', 'interspersedway'),
('unaffectedly', 'unaffectedlyway'), ('insinuations', 'insinuationsway'),
('unpardonable', 'unpardonableway'), ('arrangements', 'arrangementsway'),
('unexpectedly', 'unexpectedlyway'), ('introduction', 'introductionway'),
('occasionally', 'occasionallyway'), ('establishing', 'establishingway'),
('unanswerable', 'unanswerableway'), ('independence', 'independenceway'),
('inclinations', 'inclinationsway'), ('instructions', 'instructionsway'),
('undoubtingly', 'undoubtinglyway'), ('unfathomable', 'unfathomableway'),
('impoverished', 'impoverishedway'), ('outstretched', 'outstretchedway'),
('unrestrained', 'unrestrainedway'), ('ungraciously', 'ungraciouslyway'),
('imperfection', 'imperfectionway'), ('unreasonably', 'unreasonablyway'),
('interference', 'interferenceway'), ('inconvenient', 'inconvenientway'),
('enthusiastic', 'enthusiasticway'), ('extinguished', 'extinguishedway'),
('entanglement', 'entanglementway'), ('affectionate', 'affectionateway')], (10,
12): [('repentance', 'epentanceray'), ('capability', 'apabilitycay'),
('beneficial', 'eneficialbay'), ('retrenched', 'etrenchedray'), ('devonshire',
'evonshireday'), ('palanquins', 'alanquinspay'), ('contribute', 'ontributecay'),
('minuteness', 'inutenessmay'), ('dissimilar', 'issimilarday'), ('flattering',
'atteringflay'), ('confounded', 'onfoundedcay'), ('compromise', 'ompromisecay'),
('contrasted', 'ontrastedcay'), ('precaution', 'ecautionpray'), ('fastidious',
'astidiousfay'), ('blackguard', 'ackguardblay'), ('heroically', 'eroicallyhay'),
('lengthened', 'engthenedlay'), ('pocketbook', 'ocketbookpay'), ('companions',
'ompanionscay'), ('displeased', 'ispleasedday'), ('pianoforte', 'ianofortepay'),
('succession', 'uccessionsay'), ('composedly', 'omposedlycay'), ('pronounced',
'onouncedpray'), ('honourable', 'onourablehay'), ('solicitude', 'olicitudesay'),
('justifying', 'ustifyingjay'), ('collecting', 'ollectingcay'), ('headstrong',
'eadstronghay'), ('forbidding', 'orbiddingfay'), ('successful', 'uccessfulsay'),
('confidence', 'onfidencecay'), ('prescribed', 'escribedpray'), ('scrutinies',
'utiniesscray'), ('melancholy', 'elancholymay'), ('handsomest', 'andsomesthay'),
('prescience', 'esciencepray'), ('conclusion', 'onclusioncay'), ('remarkably',
'emarkablyray'), ('gratefully', 'atefullygray'), ('michaelmas', 'ichaelmasmay'),
('containing', 'ontainingcay'), ('discussion', 'iscussionday'), ('resentment',
'esentmentray'), ('restraints', 'estraintsray'), ('tablecloth', 'ableclothtay'),
('projecting', 'ojectingpray'), ('frightened', 'ightenedfray'), ('pardonable',
'ardonablepay'), ('variations', 'ariationsvay'), ('sensations', 'ensationssay'),
```

('rheumatism', 'eumatismrhay'), ('dispirited', 'ispiritedday'), ('politeness', 'olitenesspay'), ('suggesting', 'uggestingsay'), ('willoughby', 'illoughbyway'), ('reasonable', 'easonableray'), ('discerning', 'iscerningday'), ('previously', 'eviouslypray'), ('passionate', 'assionatepay'), ('prophecies', 'opheciespray'), ('possessing', 'ossessingpay'), ('perfection', 'erfectionpay'), ('heightened', 'eightenedhay'), ('generously', 'enerouslygay'), ('faithfully', 'aithfullyfay'), ('reconciled', 'econciledray'), ('distressed', 'istressedday'), ('respective', 'espectiveray'), ('liberality', 'iberalitylay'), ('disengaged', 'isengagedday'), ('honourably', 'onourablyhay'), ('persuading', 'ersuadingpay'), ('cultivated', 'ultivatedcay'), ('projection', 'ojectionpray'), ('disapprove', 'isapproveday'), ('conciliate', 'onciliatecay'), ('satisfying', 'atisfyingsay'), ('dispatches', 'ispatchesday'), ('repeatedly', 'epeatedlyray'), ('suspicions', 'uspicionssay'), ('perplexity', 'erplexitypay'), ('strengthen', 'engthenstray'), ('dependence', 'ependenceday'), ('dissipated', 'issipatedday'), ('conformity', 'onformitycay'), ('hysterical', 'ystericalhay'), ('throughout', 'oughoutthray'), ('sacrificed', 'acrificedsay'), ('dimensions', 'imensionsday'), ('flatteries', 'atteriesflay'), ('completely', 'ompletelycay'), ('reproaches', 'eproachesray'), ('contriving', 'ontrivingcay'), ('consequent', 'onsequentcay'), ('liveliness', 'ivelinesslay'), ('resistance', 'esistanceray'), ('caricature', 'aricaturecay'), ('stimulated', 'imulatedstay'), ('brightened', 'ightenedbray'), ('prevailing', 'evailingpray'), ('disclosure', 'isclosureday'), ('bequeathed', 'equeathedbay'), ('disappoint', 'isappointday'), ('complexion', 'omplexioncay'), ('dismounted', 'ismountedday'), ('remembered', 'ememberedray'), ('delineated', 'elineatedday'), ('subsequent', 'ubsequentsay'), ('persevered', 'erseveredpay'), ('doubtingly', 'oubtinglyday'), ('characters', 'aracterschay'), ('neighbours', 'eighboursnay'), ('witticisms', 'itticismsway'), ('materially', 'ateriallymay'), ('playthings', 'aythingsplay'), ('suspecting', 'uspectingsay'), ('conditions', 'onditionscay'), ('discovered', 'iscoveredday'), ('deficiency', 'eficiencyday'), ('regularity', 'egularityray'), ('subjection', 'ubjectionsay'), ('brightness', 'ightnessbray'), ('pleasantly', 'easantlyplay'), ('delicacies', 'elicaciesday'), ('preference', 'eferencepray'), ('difference', 'ifferenceday'), ('compassion', 'ompassioncay'), ('competence', 'ompetencecay'), ('generosity', 'enerositygay'), ('retirement', 'etirementray'), ('strikingly', 'ikinglystray'), ('subsisting', 'ubsistingsay'), ('conscience', 'onsciencecay'), ('diabolical', 'iabolicalday'), ('temptation', 'emptationtay'), ('formidable', 'ormidablefay'), ('reconsider', 'econsiderray'), ('prejudiced', 'ejudicedpray'), ('comprehend', 'omprehendcay'), ('compliment', 'omplimentcay'), ('sufferings', 'ufferingssay'), ('concluding', 'oncludingcay'), ('compliance', 'ompliancecay'), ('mosquitoes', 'osquitoesmay'), ('thoughtful', 'oughtfulthay'), ('bitterness', 'itternessbay'), ('mentioning', 'entioningmay'), ('toleration', 'olerationtay'), ('profession', 'ofessionpray'), ('discompose', 'iscomposeday'), ('waistcoats', 'aistcoatsway'), ('reflecting', 'eflectingray'), ('positively', 'ositivelypay'), ('diminished', 'iminishedday')], (6, 8): [('pretty', 'ettypray'), ('survey', 'urveysay'), ('demure', 'emureday'), ('secure', 'ecuresay'), ('sobbed', 'obbedsay'), ('moment', 'omentmay'), ('guilty', 'uiltygay'), ('combat', 'ombatcay'), ('modern', 'odernmay'), ('father', 'atherfay'), ('horses', 'orseshay'), ('closer', 'oserclay'), ('nicety', 'icetynay'), ('shaken', 'akenshay'), ('centre', 'entrecay'), ('powers', 'owerspay'), ('simple', 'implesay'), ('beings', 'eingsbay'), ('recent',

'ecentray'), ('sketch', 'etchskay'), ('loving', 'ovinglay'), ('dagger',
'aggerday'), ('hearts', 'eartshay'), ('copied', 'opiedcay'), ('gazing',
'azinggay'), ('sooner', 'oonersay'), ('disarm', 'isarmday'), ('suited',
'uitedsay'), ('doubts', 'oubtsday'), ('lifted', 'iftedlay'), ('vacant',
'acantvay'), ('palmer', 'almerpay'), ('grudge', 'udgegray'), ('folded',
'oldedfay'), ('remove', 'emoveray'), ('borrow', 'orrowbay'), ('scheme',
'emeschay'), ('solemn', 'olemnsay'), ('denote', 'enoteday'), ('bustle',
'ustlebay'), ('turned', 'urnedtay'), ('mutton', 'uttonmay'), ('tender',
'endertay'), ('hearer', 'earerhay'), ('begins', 'eginsbay'), ('taught',
'aughttay'), ('honour', 'onourhay'), ('robert', 'obertray'), ('gibson',
'ibsongay'), ('virtue', 'irtuevay'), ('vanish', 'anishvay'), ('writes',
'iteswray'), ('silent', 'ilentsay'), ('singly', 'inglysay'), ('jargon',
'argonjay'), ('grieve', 'ievegray'), ('design', 'esignday'), ('linger',
'ingerlay'), ('bathed', 'athedbay'), ('pardon', 'ardonpay'), ('rumour',
'umourray'), ('dances', 'ancesday'), ('remark', 'emarkray'), ('worded',
'ordedway'), ('honest', 'onesthay'), ('chosen', 'osenchay'), ('boyish',
'oyishbay'), ('stress', 'essstray'), ('caught', 'aughtcay'), ('potent',
'otentpay'), ('during', 'uringday'), ('stocks', 'ocksstay'), ('relics',
'elicsray'), ('tongue', 'onguetay'), ('street', 'eetstray'), ('whiled',
'iledwhay'), ('manner', 'annermay'), ('truths', 'uthstray'), ('travel',
'aveltray'), ('couple', 'ouplecay'), ('gaiety', 'aietygay'), ('giving',
'ivinggay'), ('county', 'ountycay'), ('cramps', 'ampscray'), ('wealth',
'ealthway'), ('divine', 'ivineday'), ('behold', 'eholdbay'), ('feared',
'earedfay'), ('sparks', 'arksspay'), ('finger', 'ingerfay'), ('talent',
'alenttay'), ('demand', 'emandday'), ('coffee', 'offeecay'), ('merest',
'erestmay'), ('detail', 'etailday'), ('casino', 'asinocay'), ('choose',
'oosechay'), ('warmed', 'armedway'), ('relate', 'elateray'), ('sashes',
'ashessay'), ('chance', 'ancechay'), ('rushed', 'ushedray'), ('health',
'ealthhay'), ('preyed', 'eyedpray'), ('mutual', 'utualmay'), ('months',
'onthsmay'), ('raises', 'aisesray'), ('things', 'ingsthay'), ('master',
'astermay'), ('thanks', 'anksthay'), ('cheeks', 'eekschay'), ('hazard',
'azardhay'), ('slight', 'ightslay'), ('guests', 'uestsgay'), ('halves',
'alveshay'), ('lessen', 'essenlay'), ('notice', 'oticenay'), ('higher',
'igherhay'), ('truest', 'uesttray'), ('porter', 'orterpay'), ('fickle',
'icklefay'), ('behave', 'ehavebay'), ('credit', 'editcray'), ('number',
'umbernay'), ('mother', 'othermay'), ('tricks', 'ickstray'), ('traced',
'acedtray'), ('subdue', 'ubduesay'), ('ruined', 'uinedray'), ('famous',
'amousfay'), ('danced', 'ancedday'), ('clever', 'everclay'), ('drives',
'ivesdray'), ('resist', 'esistray'), ('forget', 'orgetfay'), ('stable',
'ablestay'), ('bigger', 'iggerbay'), ('betray', 'etraybay'), ('senses',
'ensessay'), ('smiled', 'iledsmay'), ('hating', 'atinghay'), ('bolder',
'olderbay'), ('banish', 'anishbay'), ('remain', 'emainray'), ('period',
'eriodpay'), ('gossip', 'ossipgay'), ('basket', 'asketbay'), ('genius',
'eniusgay'), ('seized', 'eizedsay'), ('widely', 'idelyway'), ('pursue',
'ursuepay'), ('favour', 'avourfay'), ('sighed', 'ighedsay'), ('plants',
'antsplay'), ('proved', 'ovedpray'), ('season', 'easonsay'), ('circle',
'irclecay'), ('taking', 'akingtay'), ('screen', 'eenscray'), ('talker',
'alkertay'), ('teazed', 'eazedtay'), ('suffer', 'uffersay'), ('curate',

```
'uratecay'), ('traits', 'aitstray'), ('defect', 'efectday'), ('called',
'alledcay'), ('dining', 'iningday'), ('direct', 'irectday'), ('dinner',
'innerday'), ('joking', 'okingjay'), ('whiter', 'iterwhay'), ('limits',
'imitslay'), ('papers', 'aperspay'), ('public', 'ublicpay'), ('losing',
'osinglay'), ('handed', 'andedhay'), ('wisest', 'isestway'), ('delays',
'elaysday'), ('leaves', 'eaveslay'), ('points', 'ointspay'), ('hither',
'itherhay'), ('horrid', 'orridhay'), ('middle', 'iddlemay'), ('vastly',
'astlyvay'), ('luxury', 'uxurylay'), ('failed', 'ailedfay'), ('lodges',
'odgeslay'), ('female', 'emalefay'), ('walked', 'alkedway'), ('nipped',
'ippednay'), ('nephew', 'ephewnay'), ('settle', 'ettlesay'), ('kinder',
'inderkay'), ('bowing', 'owingbay'), ('smooth', 'oothsmay'), ('roared',
'oaredray'), ('wiping', 'ipingway'), ('jewels', 'ewelsjay'), ('breath',
'eathbray'), ('steady', 'eadystay'), ('breach', 'eachbray'), ('polite',
'olitepay'), ('colder', 'oldercay'), ('though', 'oughthay'), ('poured',
'ouredpay'), ('paused', 'ausedpay'), ('nought', 'oughtnay'), ('shrewd',
'ewdshray'), ('fourth', 'ourthfay'), ('parish', 'arishpay'), ('traded',
'adedtray'), ('glance', 'anceglay'), ('lesser', 'esserlay'), ('curacy',
'uracycay'), ('labour', 'abourlay'), ('became', 'ecamebay'), ('decide',
'ecideday'), ('kisses', 'isseskay'), ('styled', 'yledstay'), ('sprung',
'ungspray'), ('priory', 'iorypray'), ('derive', 'eriveday'), ('school',
'oolschay'), ('liking', 'ikinglay'), ('dawned', 'awnedday'), ('weaken',
'eakenway'), ('winter', 'interway'), ('rivals', 'ivalsray'), ('sallow',
'allowsay'), ('defend', 'efendday'), ('seated', 'eatedsay'), ('penury',
'enurypay'), ('repair', 'epairray'), ('chaise', 'aisechay'), ('church',
'urchchay'), ('beyond', 'eyondbay'), ('wisdom', 'isdomway'), ('retain',
'etainray'), ('toilet', 'oilettay'), ('talked', 'alkedtay'), ('dearly',
'earlyday'), ('reward', 'ewardray'), ('safely', 'afelysay'), ('making',
'akingmay'), ('debate', 'ebateday'), ('future', 'uturefay'), ('broken',
'okenbray'), ('begged', 'eggedbay'), ('thirty', 'irtythay'), ('muslin',
'uslinmay'), ('marked', 'arkedmay'), ('hinder', 'inderhay'), ('beheld',
'eheldbay'), ('maxims', 'aximsmay'), ('theirs', 'eirsthay'), ('corner',
'ornercay'), ('piqued', 'iquedpay'), ('rested', 'estedray'), ('needed',
'eedednay'), ('fitted', 'ittedfay'), ('candid', 'andidcay'), ('laughs',
'aughslay'), ('window', 'indowway'), ('tastes', 'astestay'), ('spread',
'eadspray'), ('hollow', 'ollowhay'), ('sorrow', 'orrowsay'), ('partly',
'artlypay'), ('served', 'ervedsay'), ('flowed', 'owedflay'), ('remedy',
'emedyray'), ('trying', 'yingtray'), ('latest', 'atestlay'), ('manage',
'anagemay'), ('gained', 'ainedgay'), ('plenty', 'entyplay'), ('garden',
'ardengay'), ('frosts', 'ostsfray'), ('thirds', 'irdsthay'), ('handle',
'andlehay'), ('verbal', 'erbalvay'), ('sheath', 'eathshay'), ('covert',
'overtcay'), ('darker', 'arkerday'), ('harris', 'arrishay'), ('houses',
'ouseshay'), ('nature', 'aturenay'), ('thomas', 'omasthay'), ('submit',
'ubmitsay'), ('danger', 'angerday'), ('lawyer', 'awyerlay'), ('matter',
'attermay'), ('rugged', 'uggedray'), ('highly', 'ighlyhay'), ('whence',
'encewhay'), ('martha', 'arthamay'), ('hearty', 'eartyhay'), ('recall',
'ecallray'), ('raised', 'aisedray'), ('follow', 'ollowfay'), ('nearer',
'earernay'), ('gentle', 'entlegay'), ('pitied', 'itiedpay'), ('nurses',
'ursesnay'), ('punish', 'unishpay')], (7, 9): [('parcels', 'arcelspay'),
```

```
('welcome', 'elcomeway'), ('posture', 'osturepay'), ('mothers', 'othersmay'),
('tempers', 'emperstay'), ('carries', 'arriescay'), ('neglect', 'eglectnay'),
('prevail', 'evailpray'), ('declare', 'eclareday'), ('crowned', 'ownedcray'),
('mindful', 'indfulmay'), ('blessed', 'essedblay'), ('kitchen', 'itchenkay'),
('younger', 'youngeray'), ('hanover', 'anoverhay'), ('lowered', 'oweredlay'),
('summons', 'ummonssay'), ('suffice', 'ufficesay'), ('rousing', 'ousingray'),
('feeling', 'eelingfay'), ('parlors', 'arlorspay'), ('gravely', 'avelygray'),
('connect', 'onnectcay'), ('species', 'eciesspay'), ('revived', 'evivedray'),
('tribute', 'ibutetray'), ('pitying', 'ityingpay'), ('brother', 'otherbray'),
('retired', 'etiredray'), ('suffers', 'ufferssay'), ('spurned', 'urnedspay'),
('dropped', 'oppeddray'), ('galling', 'allinggay'), ('seducer', 'educersay'),
('foibles', 'oiblesfay'), ('tuesday', 'uesdaytay'), ('patient', 'atientpay'),
('worsted', 'orstedway'), ('trifled', 'ifledtray'), ('waiting', 'aitingway'),
('widower', 'idowerway'), ('hardily', 'ardilyhay'), ('glimpse', 'impseglay'),
('whether', 'etherwhay'), ('reverie', 'everieray'), ('resolve', 'esolveray'),
('propose', 'oposepray'), ('tallest', 'allesttay'), ('nominal', 'ominalnay'),
('languid', 'anguidlay'), ('wronged', 'ongedwray'), ('nothing', 'othingnay'),
('promote', 'omotepray'), ('novelty', 'oveltynay'), ('dooming', 'oomingday'),
('display', 'isplayday'), ('correct', 'orrectcay'), ('feigned', 'eignedfay'),
('schemes', 'emesschay'), ('deepest', 'eepestday'), ('flannel', 'annelflay'),
('hauteur', 'auteurhay'), ('sharing', 'aringshay'), ('getting', 'ettinggay'),
('screams', 'eamsscray'), ('variety', 'arietyvay'), ('yielded', 'yieldeday'),
('brought', 'oughtbray'), ('haughty', 'aughtyhay'), ('bristol', 'istolbray'),
('silence', 'ilencesay'), ('staying', 'ayingstay'), ('confess', 'onfesscay'),
('placing', 'acingplay'), ('noisier', 'oisiernay'), ('nursing', 'ursingnay'),
('tuition', 'uitiontay'), ('welfare', 'elfareway'), ('hunters', 'untershay'),
('poplars', 'oplarspay'), ('deprive', 'epriveday'), ('mansion', 'ansionmay'),
('reasons', 'easonsray'), ('conceit', 'onceitcay'), ('burgess', 'urgessbay'),
('receipt', 'eceiptray'), ('befalls', 'efallsbay'), ('fetches', 'etchesfay'),
('ranging', 'angingray'), ('passion', 'assionpay'), ('smirked', 'irkedsmay'),
('honours', 'onourshay'), ('covered', 'overedcay'), ('dancing', 'ancingday'),
('hundred', 'undredhay'), ('forgave', 'orgavefay'), ('fifteen', 'ifteenfay'),
('lasting', 'astinglay'), ('minuter', 'inutermay'), ('tremour', 'emourtray'),
('passing', 'assingpay'), ('streets', 'eetsstray'), ('resists', 'esistsray'),
('cleared', 'earedclay'), ('miracle', 'iraclemay'), ('several', 'everalsay'),
('sighing', 'ighingsay'), ('resumed', 'esumedray'), ('merited', 'eritedmay'),
('without', 'ithoutway'), ('removed', 'emovedray'), ('behaved', 'ehavedbay'),
('succour', 'uccoursay'), ('bribing', 'ibingbray'), ('charity', 'aritychay'),
('shortly', 'ortlyshay'), ('tempted', 'emptedtay'), ('briefly', 'ieflybray'),
('parties', 'artiespay'), ('touched', 'ouchedtay'), ('content', 'ontentcay'),
('proceed', 'oceedpray'), ('windows', 'indowsway'), ('ribbons', 'ibbonsray'),
('stepped', 'eppedstay'), ('pursuit', 'ursuitpay'), ('quarter', 'arterquay'),
('prudent', 'udentpray'), ('dignity', 'ignityday'), ('payment', 'aymentpay'),
('grounds', 'oundsgray'), ('bequest', 'equestbay'), ('donavan', 'onavanday'),
('renewed', 'enewedray'), ('smaller', 'allersmay'), ('fearful', 'earfulfay'),
('civilly', 'ivillycay'), ('romance', 'omanceray'), ('females', 'emalesfay'),
('grieved', 'ievedgray'), ('created', 'eatedcray'), ('command', 'ommandcay'),
('defined', 'efinedday'), ('bluntly', 'untlyblay'), ('recital', 'ecitalray'),
```

```
('sorrows', 'orrowssay'), ('whither', 'itherwhay'), ('spotted', 'ottedspay'),
('justify', 'ustifyjay'), ('conduit', 'onduitcay'), ('revenge', 'evengeray'),
('scenery', 'eneryscay'), ('started', 'artedstay'), ('degrees', 'egreesday'),
('whisper', 'isperwhay'), ('strange', 'angestray'), ('bidding', 'iddingbay'),
('musical', 'usicalmay'), ('happens', 'appenshay'), ('compare', 'omparecay'),
('presume', 'esumepray'), ('sitting', 'ittingsay'), ('regards', 'egardsray'),
('violins', 'iolinsvay'), ('rightly', 'ightlyray'), ('memento', 'ementomay'),
('courted', 'ourtedcay'), ('servant', 'ervantsay'), ('possess', 'ossesspay'),
('herself', 'erselfhay'), ('present', 'esentpray'), ('natural', 'aturalnay'),
('managed', 'anagedmay'), ('hostess', 'ostesshay'), ('leaving', 'eavinglay'),
('mention', 'entionmay'), ('circuit', 'ircuitcay'), ('counsel', 'ounselcay'),
('guineas', 'uineasgay'), ('solidly', 'olidlysay'), ('dashing', 'ashingday'),
('station', 'ationstay'), ('written', 'ittenwray'), ('writing', 'itingwray'),
('country', 'ountrycay'), ('delight', 'elightday'), ('westons', 'estonsway'),
('debated', 'ebatedday'), ('lengths', 'engthslay'), ('torment', 'ormenttay'),
('reverse', 'everseray'), ('bedroom', 'edroombay'), ('forbade', 'orbadefay'),
('comings', 'omingscay'), ('tonight', 'onighttay'), ('similar', 'imilarsay'),
('calling', 'allingcay'), ('cassino', 'assinocay'), ('strains', 'ainsstray'),
('scratch', 'atchscray'), ('gardens', 'ardensgay'), ('freedom', 'eedomfray'),
('compact', 'ompactcay'), ('message', 'essagemay'), ('heavily', 'eavilyhay'),
('manners', 'annersmay'), ('systems', 'ystemssay'), ('forbore', 'orborefay'),
('wrapped', 'appedwray'), ('defects', 'efectsday'), ('villain', 'illainvay'),
('studied', 'udiedstay'), ('playing', 'ayingplay'), ('diction', 'ictionday'),
('tending', 'endingtay'), ('seeking', 'eekingsay'), ('rapture', 'aptureray'),
('dawdled', 'awdledday'), ('wearing', 'earingway'), ('comfort', 'omfortcay'),
('finally', 'inallyfay'), ('circles', 'irclescay'), ('putting', 'uttingpay'),
('twisted', 'istedtway'), ('destroy', 'estroyday'), ('whereas', 'ereaswhay'),
('dismiss', 'ismissday'), ('visibly', 'isiblyvay'), ('talents', 'alentstay'),
('finding', 'indingfay'), ('removes', 'emovesray'), ('replied', 'epliedray'),
('bluster', 'usterblay'), ('raising', 'aisingray'), ('plainly', 'ainlyplay'),
('worried', 'orriedway'), ('meaning', 'eaningmay'), ('blushes', 'ushesblay'),
('relapse', 'elapseray'), ('careful', 'arefulcay'), ('beloved', 'elovedbay'),
('degrade', 'egradeday'), ('foresaw', 'oresawfay'), ('procure', 'ocurepray'),
('kindest', 'indestkay'), ('secures', 'ecuressay'), ('hanging', 'anginghay'),
('persons', 'ersonspay'), ('taverns', 'avernstay'), ('walking', 'alkingway'),
('quicker', 'ickerquay'), ('guessed', 'uessedgay'), ('confirm', 'onfirmcay'),
('venture', 'enturevay'), ('project', 'ojectpray'), ('prayers', 'ayerspray'),
('heights', 'eightshay'), ('subject', 'ubjectsay'), ('summits', 'ummitssay'),
('towards', 'owardstay'), ('closing', 'osingclay'), ('journey', 'ourneyjay'),
('pointer', 'ointerpay'), ('chained', 'ainedchay'), ('clothes', 'othesclay'),
('require', 'equireray'), ('chusing', 'usingchay'), ('reproof', 'eproofray'),
('reckons', 'eckonsray'), ('pressed', 'essedpray'), ('talking', 'alkingtay'),
('harbour', 'arbourhay'), ('garrets', 'arretsgay'), ('foresee', 'oreseefay'),
('lustily', 'ustilylay'), ('leagued', 'eaguedlay'), ('judging', 'udgingjay'),
('rejoice', 'ejoiceray'), ('honiton', 'onitonhay'), ('shorten', 'ortenshay'),
('stating', 'atingstay'), ('privacy', 'ivacypray'), ('steeles', 'eelesstay'),
('thickly', 'icklythay'), ('shocked', 'ockedshay'), ('limited', 'imitedlay'),
('blights', 'ightsblay'), ('seclude', 'ecludesay'), ('cordial', 'ordialcay'),
```

```
('request', 'equestray'), ('denoted', 'enotedday'), ('darling', 'arlingday'),
('coxcomb', 'oxcombcay'), ('however', 'oweverhay'), ('perform', 'erformpay'),
('society', 'ocietysay'), ('deserts', 'esertsday'), ('palmers', 'almerspay'),
('manager', 'anagermay'), ('rupture', 'uptureray'), ('concise', 'oncisecay'),
('scolded', 'oldedscay'), ('seduced', 'educedsay'), ('doubted', 'oubtedday'),
('forward', 'orwardfay'), ('leisure', 'eisurelay'), ('robbing', 'obbingray'),
('horizon', 'orizonhay'), ('blunder', 'underblay'), ('flushed', 'ushedflay'),
('divided', 'ividedday'), ('charged', 'argedchay'), ('disturb', 'isturbday'),
('smitten', 'ittensmay'), ('seconds', 'econdssay'), ('letters', 'etterslay'),
('provide', 'ovidepray'), ('conduct', 'onductcay'), ('plagued', 'aguedplay'),
('stretch', 'etchstray'), ('hoarded', 'oardedhay'), ('blushed', 'ushedblay'),
('nodding', 'oddingnay'), ('mankind', 'ankindmay'), ('caprice', 'apricecay'),
('quitted', 'ittedquay')], (5, 7): [('china', 'inachay'), ('deter', 'eterday'),
('woman', 'omanway'), ('clock', 'ockclay'), ('borne', 'ornebay'), ('noise',
'oisenay'), ('midst', 'idstmay'), ('dwell', 'elldway'), ('smith', 'ithsmay'),
('gowns', 'ownsgay'), ('named', 'amednay'), ('glass', 'assglay'), ('price',
'icepray'), ('close', 'oseclay'), ('guide', 'uidegay'), ('newly', 'ewlynay'),
('sweep', 'eepsway'), ('swept', 'eptsway'), ('those', 'osethay'), ('lower',
'owerlay'), ('plump', 'umpplay'), ('downs', 'ownsday'), ('purse', 'ursepay'),
('linen', 'inenlay'), ('later', 'aterlay'), ('broad', 'oadbray'), ('miles',
'ilesmay'), ('facts', 'actsfay'), ('sleep', 'eepslay'), ('steps', 'epsstay'),
('wants', 'antsway'), ('forth', 'orthfay'), ('drove', 'ovedray'), ('mouth',
'outhmay'), ('worry', 'orryway'), ('dried', 'ieddray'), ('stood', 'oodstay'),
('brush', 'ushbray'), ('sting', 'ingstay'), ('mirth', 'irthmay'), ('heads',
'eadshay'), ('sight', 'ightsay'), ('whose', 'osewhay'), ('drawn', 'awndray'),
('score', 'orescay'), ('party', 'artypay'), ('tears', 'earstay'), ('heart',
'earthay'), ('words', 'ordsway'), ('hunts', 'untshay'), ('trace', 'acetray'),
('trial', 'ialtray'), ('swoon', 'oonsway'), ('knees', 'eesknay'), ('bring',
'ingbray'), ('solid', 'olidsay'), ('sharp', 'arpshay'), ('basis', 'asisbay'),
('cavil', 'avilcay'), ('rated', 'atedray'), ('grave', 'avegray'), ('serve',
'ervesay'), ('betty', 'ettybay'), ('judge', 'udgejay'), ('boast', 'oastbay'),
('march', 'archmay'), ('built', 'uiltbay'), ('clear', 'earclay'), ('happy',
'appyhay'), ('fruit', 'uitfray'), ('witty', 'ittyway'), ('total', 'otaltay'),
('dared', 'aredday'), ('defer', 'eferday'), ('tried', 'iedtray'), ('vexed',
'exedvay'), ('walls', 'allsway'), ('liked', 'ikedlay'), ('prove', 'ovepray'),
('sneer', 'eersnay'), ('merit', 'eritmay'), ('scott', 'ottscay'), ('madam',
'adammay'), ('truth', 'uthtray'), ('minor', 'inormay'), ('stare', 'arestay'),
('calls', 'allscay'), ('lucky', 'uckylay'), ('hence', 'encehay'), ('press',
'esspray'), ('comes', 'omescay'), ('dined', 'inedday'), ('think', 'inkthay'),
('plate', 'ateplay'), ('grief', 'iefgray'), ('names', 'amesnay'), ('stock',
'ockstay'), ('short', 'ortshay'), ('moved', 'ovedmay'), ('teach', 'eachtay'),
('fears', 'earsfay'), ('tiled', 'iledtay'), ('cried', 'iedcray'), ('mound',
'oundmay'), ('small', 'allsmay'), ('twice', 'icetway'), ('lines', 'ineslay'),
('looks', 'ookslay'), ('rival', 'ivalray'), ('first', 'irstfay'), ('wound',
'oundway'), ('saves', 'avessay'), ('nurse', 'ursenay'), ('treat', 'eattray'),
('could', 'ouldcay'), ('rider', 'iderray'), ('lobby', 'obbylay'), ('table',
'abletay'), ('touch', 'ouchtay'), ('topic', 'opictay'), ('stand', 'andstay'),
('leant', 'eantlay'), ('grass', 'assgray'), ('check', 'eckchay'), ('quite',
```

'itequay'), ('wives', 'ivesway'), ('walks', 'alksway'), ('labor', 'aborlay'), ('house', 'ousehay'), ('drops', 'opsdray'), ('loose', 'ooselay'), ('tempt', 'empttay'), ('place', 'aceplay'), ('local', 'ocallay'), ('seats', 'eatssay'), ('gates', 'atesgay'), ('dropt', 'optdray'), ('cares', 'arescay'), ('falls', 'allsfay'), ('speak', 'eakspay'), ('shops', 'opsshay'), ('groom', 'oomgray'), ('sugar', 'ugarsay'), ('front', 'ontfray'), ('visit', 'isitvay'), ('raise', 'aiseray'), ('fetch', 'etchfay'), ('silks', 'ilkssay'), ('beset', 'esetbay'), ('rocks', 'ocksray'), ('start', 'artstay'), ('winks', 'inksway'), ('sends', 'endssay'), ('piano', 'ianopay'), ('fully', 'ullyfay'), ('widow', 'idowway'), ('wiser', 'iserway'), ('dirty', 'irtyday'), ('below', 'elowbay'), ('manor', 'anormay'), ('scale', 'alescay'), ('whole', 'olewhay'), ('minds', 'indsmay'), ('glebe', 'ebeglay'), ('fifty', 'iftyfay'), ('years', 'yearsay'), ('story', 'orystay'), ('fresh', 'eshfray'), ('begun', 'egunbay'), ('dance', 'anceday'), ('jokes', 'okesjay'), ('turns', 'urnstay'), ('honor', 'onorhay'), ('spite', 'itespay'), ('green', 'eengray'), ('saint', 'aintsay'), ('hated', 'atedhay'), ('doors', 'oorsday'), ('shown', 'ownshay'), ('power', 'owerpay'), ('today', 'odaytay'), ('floor', 'oorflay'), ('force', 'orcefay'), ('right', 'ightray'), ('trick', 'icktray'), ('dread', 'eaddray'), ('final', 'inalfay'), ('shake', 'akeshay'), ('lamps', 'ampslay'), ('dying', 'yingday'), ('black', 'ackblay'), ('grand', 'andgray'), ('field', 'ieldfay'), ('plead', 'eadplay'), ('loved', 'ovedlay'), ('terms', 'ermstay'), ('spend', 'endspay'), ('mohrs', 'ohrsmay'), ('young', 'youngay'), ('great', 'eatgray'), ('guess', 'uessgay'), ('ninny', 'innynay'), ('hasty', 'astyhay'), ('nerve', 'ervenay'), ('meals', 'ealsmay'), ('three', 'eethray'), ('shade', 'adeshay'), ('store', 'orestay'), ('daily', 'ailyday'), ('taste', 'astetay'), ('gross', 'ossgray'), ('dares', 'aresday'), ('grate', 'ategray'), ('stuff', 'uffstay'), ('sheet', 'eetshay')], (14, 18): [('breakfast-room', 'eakfastbray-oomray'), ('lavender-water', 'avenderlay-aterway'), ('broken-hearted', 'okenbray-eartedhay'), ('spunging-house', 'ungingspay-ousehay')], (9, 12): [('unwelcome', 'unwelcomeway'), ('exhausted', 'exhaustedway'), ('intrusion', 'intrusionway'), ('untouched', 'untouchedway'), ('interview', 'interviewway'), ('effusions', 'effusionsway'), ('impartial', 'impartialway'), ('inventive', 'inventiveway'), ('emergency', 'emergencyway'), ('officious', 'officiousway'), ('everybody', 'everybodyway'), ('observant', 'observantway'), ('emptiness', 'emptinessway'), ('occasions', 'occasionsway'), ('assembled', 'assembledway'), ('admirable', 'admirableway'), ('uselessly', 'uselesslyway'), ('excellent', 'excellentway'), ('associate', 'associateway'), ('affecting', 'affectingway'), ('absurdity', 'absurdityway'), ('allowable', 'allowableway'), ('appointed', 'appointedway'), ('overheard', 'overheardway'), ('improving', 'improvingway'), ('eagerness', 'eagernessway'), ('unluckily', 'unluckilyway'), ('allowance', 'allowanceway'), ('estimable', 'estimableway'), ('obviating', 'obviatingway'), ('entrusted', 'entrustedway'), ('arranging', 'arrangingway'), ('exclaimed', 'exclaimedway'), ('ingenuity', 'ingenuityway'), ('outwardly', 'outwardlyway'), ('expensive', 'expensiveway'), ('entreated', 'entreatedway'), ('unconcern', 'unconcernway'), ('obtaining', 'obtainingway'), ('interests', 'interestsway'), ('assiduous', 'assiduousway'), ('adjusting', 'adjustingway'), ('ungenteel', 'ungenteelway'), ('encourage', 'encourageway'), ('apartment', 'apartmentway'), ('inheritor', 'inheritorway'), ('unceasing', 'unceasingway'), ('insulting', 'insultingway'), ('unchanged', 'unchangedway'),

```
('animation', 'animationway'), ('enclosure', 'enclosureway'), ('apologize',
'apologizeway'), ('earnestly', 'earnestlyway'), ('exquisite', 'exquisiteway'),
('attending', 'attendingway'), ('invention', 'inventionway'), ('enquiries',
'enquiriesway'), ('appetites', 'appetitesway'), ('additions', 'additionsway'),
('impudence', 'impudenceway'), ('insincere', 'insincereway'), ('abundance',
'abundanceway'), ('exchanged', 'exchangedway'), ('extremely', 'extremelyway'),
('inflicted', 'inflictedway'), ('attracted', 'attractedway'), ('inelegant',
'inelegantway'), ('undoubted', 'undoubtedway'), ('etiquette', 'etiquetteway'),
('unfeigned', 'unfeignedway'), ('agreeable', 'agreeableway'), ('unsubdued',
'unsubduedway'), ('inspiring', 'inspiringway'), ('unguarded', 'unguardedway'),
('advancing', 'advancingway'), ('undressed', 'undressedway'), ('excursive',
'excursiveway'), ('extending', 'extendingway'), ('extolling', 'extollingway'),
('offending', 'offendingway'), ('eloquence', 'eloquenceway'), ('ornaments',
'ornamentsway'), ('intruding', 'intrudingway'), ('incessant', 'incessantway'),
('unwilling', 'unwillingway'), ('annamaria', 'annamariaway'), ('unbounded',
'unboundedway'), ('incurable', 'incurableway'), ('unnoticed', 'unnoticedway'),
('expressly', 'expresslyway'), ('impetuous', 'impetuousway'), ('appearing',
'appearingway'), ('ingenious', 'ingeniousway'), ('amazement', 'amazementway'),
('animating', 'animatingway'), ('interfere', 'interfereway'), ('ascending',
'ascendingway'), ('affection', 'affectionway'), ('undergone', 'undergoneway'),
('unstudied', 'unstudiedway'), ('attempted', 'attemptedway'), ('infirmity',
'infirmityway'), ('apologies', 'apologiesway'), ('elsewhere', 'elsewhereway'),
('expecting', 'expectingway'), ('informing', 'informingway'), ('admission',
'admissionway'), ('extremity', 'extremityway')], (6, 9): [('unread',
'unreadway'), ('owning', 'owningway'), ('odious', 'odiousway'), ('either',
'eitherway'), ('insult', 'insultway'), ('awhile', 'awhileway'), ('assist',
'assistway'), ('injure', 'injureway'), ('eluded', 'eludedway'), ('others',
'othersway'), ('untied', 'untiedway'), ('excess', 'excessway'), ('urgent',
'urgentway'), ('appear', 'appearway'), ('aspect', 'aspectway'), ('issued',
'issuedway'), ('evince', 'evinceway'), ('eating', 'eatingway'), ('easter',
'easterway'), ('arisen', 'arisenway'), ('earned', 'earnedway'), ('esteem',
'esteemway'), ('accuse', 'accuseway'), ('owners', 'ownersway'), ('events',
'eventsway'), ('energy', 'energyway'), ('affair', 'affairway'), ('effect',
'effectway'), ('unlike', 'unlikeway'), ('unison', 'unisonway'), ('enough',
'enoughway'), ('invite', 'inviteway'), ('income', 'incomeway'), ('extent',
'extentway'), ('office', 'officeway'), ('ankles', 'anklesway'), ('admire',
'admireway'), ('induce', 'induceway'), ('infant', 'infantway'), ('expand',
'expandway'), ('orders', 'ordersway'), ('occupy', 'occupyway'), ('attend',
'attendway'), ('unable', 'unableway'), ('olives', 'olivesway'), ('awaken',
'awakenway'), ('engage', 'engageway'), ('eldest', 'eldestway'), ('eyeing',
'eyeingway'), ('advice', 'adviceway'), ('always', 'alwaysway'), ('insist',
'insistway'), ('accept', 'acceptway'), ('indies', 'indiesway'), ('excite',
'exciteway'), ('openly', 'openlyway'), ('oblige', 'obligeway'), ('amidst',
'amidstway'), ('across', 'acrossway'), ('afraid', 'afraidway'), ('uproar',
'uproarway')], (8, 10): [('villages', 'illagesvay'), ('doubtful', 'oubtfulday'),
('deceived', 'eceivedday'), ('foreseen', 'oreseenfay'), ('dennison',
'ennisonday'), ('steadily', 'eadilystay'), ('startled', 'artledstay'),
('drawings', 'awingsdray'), ('rejected', 'ejectedray'), ('deserves',
```

'eservesday'), ('curricle', 'urriclecay'), ('domestic', 'omesticday'),
('believed', 'elievedbay'), ('packages', 'ackagespay'), ('kneeling',
'eelingknay'), ('degraded', 'egradedday'), ('medicine', 'edicinemay'),
('promoted', 'omotedpray'), ('wondered', 'onderedway'), ('reproach',
'eproachray'), ('somerset', 'omersetsay'), ('contents', 'ontentscay'),
('children', 'ildrenchay'), ('politics', 'oliticspay'), ('hastened',
'astenedhay'), ('relative', 'elativeray'), ('henshawe', 'enshawehay'),
('material', 'aterialmay'), ('welcomed', 'elcomedway'), ('weddings',
'eddingsway'), ('diverted', 'ivertedday'), ('patience', 'atiencepay'),
('delivery', 'eliveryday'), ('hazarded', 'azardedhay'), ('happiest',
'appiesthay'), ('trifling', 'iflingtray'), ('casement', 'asementcay'),
('reformed', 'eformedray'), ('demanded', 'emandedday'), ('ceremony',
'eremonycay'), ('vigorous', 'igorousvay'), ('mornings', 'orningsmay'),
('hilarity', 'ilarityhay'), ('daughter', 'aughterday'), ('vexation',
'exationvay'), ('shocking', 'ockingshay'), ('previous', 'eviouspray'),
('pleasant', 'easantplay'), ('candidly', 'andidlycay'), ('cheerful',
'eerfulchay'), ('festival', 'estivalfay'), ('changing', 'angingchay'),
('conveyed', 'onveyedcay'), ('lombardy', 'ombardylay'), ('militate',
'ilitatemay'), ('humanity', 'umanityhay'), ('giggling', 'igglinggay'),
('standard', 'andardstay'), ('returned', 'eturnedray'), ('proclaim',
'oclaimpray'), ('ridicule', 'idiculeray'), ('pointers', 'ointerspay'),
('grounded', 'oundedgray'), ('contrast', 'ontrastcay'), ('feelings',
'eelingsfay'), ('mistakes', 'istakesmay'), ('prepared', 'eparedpray'),
('stranger', 'angerstray'), ('habitual', 'abitualhay'), ('firmness',
'irmnessfay'), ('rational', 'ationalray'), ('regulate', 'egulateray'),
('claiming', 'aimingclay'), ('knocking', 'ockingknay'), ('servants',
'ervantssay'), ('goodness', 'oodnessgay'), ('mutually', 'utuallymay'),
('legacies', 'egacieslay'), ('sisterly', 'isterlysay'), ('punctual',
'unctualpay'), ('deciding', 'ecidingday'), ('suitable', 'uitablesay'),
('guarding', 'uardinggay'), ('converse', 'onversecay'), ('slighter',
'ighterslay'), ('dissuade', 'issuadeday'), ('despatch', 'espatchday'),
('blackest', 'ackestblay'), ('recalled', 'ecalledray'), ('loitered',
'oiteredlay'), ('dreading', 'eadingdray'), ('relieved', 'elievedray'),
('westward', 'estwardway'), ('thistles', 'istlesthay'), ('defended',
'efendedday'), ('contempt', 'ontemptcay'), ('sensibly', 'ensiblysay'),
('romantic', 'omanticray'), ('comforts', 'omfortscay'), ('jealousy',
'ealousyjay'), ('repeated', 'epeatedray'), ('banditti', 'andittibay'),
('greatest', 'eatestgray'), ('mistrust', 'istrustmay'), ('forwards',
'orwardsfay'), ('rewarded', 'ewardedray'), ('watching', 'atchingway'),
('negative', 'egativenay'), ('sympathy', 'ympathysay'), ('helpless',
'elplesshay'), ('quickest', 'ickestquay'), ('portrait', 'ortraitpay'),
('consoled', 'onsoledcay'), ('fourteen', 'ourteenfay'), ('retarded',
'etardedray'), ('comments', 'ommentscay'), ('jennings', 'enningsjay'),
('securing', 'ecuringsay'), ('withdrew', 'ithdrewway'), ('youthful',
'youthfulay'), ('bachelor', 'achelorbay'), ('convince', 'onvincecay'),
('measured', 'easuredmay'), ('goodwill', 'oodwillgay'), ('branched',
'anchedbray'), ('purchase', 'urchasepay'), ('covenant', 'ovenantcay'),
('reseated', 'eseatedray'), ('concerns', 'oncernscay'), ('hardship',

'ardshiphay'), ('lessened', 'essenedlay'), ('delaford', 'elafordday'),
('governed', 'overnedgay'), ('listened', 'istenedlay'), ('constant',
'onstantcay'), ('fatigued', 'atiguedfay'), ('persuade', 'ersuadepay'),
('critique', 'itiquecray'), ('remedies', 'emediesray'), ('stepping',
'eppingstay'), ('kindness', 'indnesskay'), ('sociable', 'ociablesay'),
('meetings', 'eetingsmay'), ('deserved', 'eservedday'), ('thoughts',
'oughtsthay'), ('becoming', 'ecomingbay'), ('heartily', 'eartilyhay'),
('promised', 'omisedpray'), ('shutting', 'uttingshay'), ('reproved',
'eprovedray'), ('ladyship', 'adyshiplay'), ('pursuing', 'ursuingpay'),
('residing', 'esidingray'), ('sentence', 'entencesay'), ('revealed',
'evealedray'), ('decrease', 'ecreaseday'), ('speedily', 'eedilyspay'),
('commands', 'ommandscay'), ('mistaken', 'istakenmay'), ('settling',
'ettlingsay'), ('polished', 'olishedpay'), ('stanhill', 'anhillstay'),
('retailed', 'etailedray'), ('pointing', 'ointingpay'), ('strongly',
'onglystray'), ('subjects', 'ubjectssay'), ('calmness', 'almnesscay'),
('doctrine', 'octrineday'), ('handsome', 'andsomehay'), ('profited',
'ofitedpray'), ('planning', 'anningplay'), ('denoting', 'enotingday'),
('commonly', 'ommonlycay'), ('bestowed', 'estowedbay'), ('marriage',
'arriagemay'), ('presents', 'esentspray'), ('complete', 'ompletecay'),
('striking', 'ikingstray'), ('needless', 'eedlessnay'), ('plighted',
'ightedplay'), ('tranquil', 'anquiltray'), ('spacious', 'aciousspay'),
('nonsense', 'onsensenay'), ('likeness', 'ikenesslay'), ('valuable',
'aluablevay'), ('laudable', 'audablelay'), ('delirium', 'eliriumday'),
('believes', 'elievesbay'), ('premises', 'emisespray'), ('presumed',
'esumedpray'), ('coldness', 'oldnesscay'), ('metaphor', 'etaphormay'),
('thankful', 'ankfulthay'), ('suffered', 'ufferedsay'), ('preserve',
'eservepray'), ('resorted', 'esortedray'), ('felicity', 'elicityfay'),
('betrayed', 'etrayedbay'), ('throwing', 'owingthray'), ('powerful',
'owerfulpay'), ('humoured', 'umouredhay'), ('practise', 'actisepray'),
('coloured', 'olouredcay'), ('regiment', 'egimentray'), ('stubborn',
'ubbornstay'), ('huswifes', 'uswifeshay'), ('painting', 'aintingpay'),
('friendly', 'iendlyfray'), ('happened', 'appenedhay'), ('moderate',
'oderatemay'), ('solitary', 'olitarysay'), ('wildness', 'ildnessway'),
('marianne', 'ariannemay'), ('delicacy', 'elicacyday'), ('progress',
'ogresspray'), ('coachman', 'oachmancay'), ('grieving', 'ievinggray'),
('troubles', 'oublestray'), ('coolness', 'oolnesscay'), ('possibly',
'ossiblypay'), ('relating', 'elatingray'), ('patterns', 'atternspay'),
('consists', 'onsistscay'), ('creating', 'eatingcray'), ('confused',
'onfusedcay'), ('critical', 'iticalcray'), ('laboured', 'abouredlay'),
('minutely', 'inutelymay'), ('tendency', 'endencytay'), ('distrust',
'istrustday'), ('distress', 'istressday'), ('conclude', 'oncludecay'),
('sixpence', 'ixpencesay'), ('somewhat', 'omewhatsay'), ('coincide',
'oincidecay'), ('harmless', 'armlesshay'), ('juvenile', 'uvenilejay'),
('pleasing', 'easingplay'), ('precious', 'eciouspray'), ('restrain',
'estrainray'), ('produced', 'oducedpray'), ('renewing', 'enewingray'),
('pinching', 'inchingpay'), ('severely', 'everelysay'), ('blossoms',
'ossomsblay'), ('heighten', 'eightenhay'), ('wandered', 'anderedway'),
('properly', 'operlypray'), ('veracity', 'eracityvay'), ('released',

'eleasedray'), ('transact', 'ansacttray'), ('helpmate', 'elpmatehay'), ('business', 'usinessbay'), ('concerto', 'oncertocay'), ('reserved', 'eservedray'), ('reliance', 'elianceray'), ('headache', 'eadachehay'), ('families', 'amiliesfay'), ('centered', 'enteredcay'), ('mulberry', 'ulberrymay'), ('sensible', 'ensiblesay'), ('steepest', 'eepeststay'), ('compared', 'omparedcay'), ('speaking', 'eakingspay'), ('censured', 'ensuredcay'), ('screened', 'eenedscray'), ('pictures', 'icturespay'), ('reaching', 'eachingray'), ('wonderer', 'ondererway'), ('whenever', 'eneverwhay'), ('detested', 'etestedday'), ('dispatch', 'ispatchday'), ('shutters', 'uttersshay'), ('disgrace', 'isgraceday'), ('dejected', 'ejectedday'), ('conquest', 'onquestcay'), ('foretell', 'oretellfay'), ('weakened', 'eakenedway'), ('brooding', 'oodingbray'), ('shedding', 'eddingshay'), ('poignant', 'oignantpay'), ('trembled', 'embledtray'), ('resolute', 'esoluteray'), ('reverted', 'evertedray'), ('silenced', 'ilencedsay'), ('jointure', 'ointurejay')], (10, 13): [('illaudable', 'illaudableway'), ('incivility', 'incivilityway'), ('oppression', 'oppressionway'), ('acquainted', 'acquaintedway'), ('economical', 'economicalway'), ('ebullition', 'ebullitionway'), ('affronting', 'affrontingway'), ('unforeseen', 'unforeseenway'), ('auspicious', 'auspiciousway'), ('indecorous', 'indecorousway'), ('attractive', 'attractiveway'), ('exercising', 'exercisingway'), ('unsatiable', 'unsatiableway'), ('individual', 'individualway'), ('enthusiasm', 'enthusiasmway'), ('announcing', 'announcingway'), ('indistinct', 'indistinctway'), ('affliction', 'afflictionway'), ('imaginable', 'imaginableway'), ('obstructed', 'obstructedway'), ('improperly', 'improperlyway'), ('illiterate', 'illiterateway'), ('interested', 'interestedway'), ('infamously', 'infamouslyway'), ('amusements', 'amusementsway'), ('endeavours', 'endeavoursway'), ('inhabiting', 'inhabitingway'), ('indisposed', 'indisposedway'), ('acquiesced', 'acquiescedway'), ('anticipate', 'anticipateway'), ('accurately', 'accuratelyway'), ('influenced', 'influencedway'), ('assemblies', 'assembliesway'), ('approached', 'approachedway'), ('accusation', 'accusationway'), ('apparently', 'apparentlyway'), ('intimately', 'intimatelyway'), ('uneasiness', 'uneasinessway'), ('alteration', 'alterationway'), ('artificial', 'artificialway'), ('admiration', 'admirationway'), ('accidental', 'accidentalway'), ('unexpected', 'unexpectedway'), ('insensible', 'insensibleway'), ('expressing', 'expressingway'), ('accelerate', 'accelerateway'), ('everything', 'everythingway'), ('increasing', 'increasingway'), ('undervalue', 'undervalueway'), ('invaluable', 'invaluableway'), ('affections', 'affectionsway'), ('infinitely', 'infinitelyway'), ('unsuitable', 'unsuitableway'), ('authorised', 'authorisedway'), ('impudently', 'impudentlyway'), ('unpardoned', 'unpardonedway'), ('inevitable', 'inevitableway'), ('ungenerous', 'ungenerousway'), ('impression', 'impressionway'), ('indelicacy', 'indelicacyway'), ('inducement', 'inducementway'), ('unaffected', 'unaffectedway'), ('instigated', 'instigatedway'), ('inquietude', 'inquietudeway'), ('attempting', 'attemptingway'), ('experience', 'experienceway'), ('impatience',

'impatienceway'), ('indulgence', 'indulgenceway'), ('affability',
'affabilityway'), ('unavailing', 'unavailingway'), ('incommoded',
'incommodedway'), ('altogether', 'altogetherway')], (4, 6): [('done', 'oneday'),
('wife', 'ifeway'), ('plea', 'eaplay'), ('mall', 'allmay'), ('sang', 'angsay'),
('clue', 'ueclay'), ('bell', 'ellbay'), ('bind', 'indbay'), ('join', 'oinjay'),
('went', 'entway'), ('vile', 'ilevay'), ('hens', 'enshay'), ('tore', 'oretay'),
('rage', 'ageray'), ('flow', 'owflay'), ('gate', 'ategay'), ('palm', 'almpay'),
('give', 'ivegay'), ('lips', 'ipslay'), ('sink', 'inksay'), ('rent', 'entray'),
('gave', 'avegay'), ('joke', 'okejay'), ('miss', 'issmay'), ('mere', 'eremay'),
('made', 'ademay'), ('dogs', 'ogsday'), ('bent', 'entbay'), ('duly', 'ulyday'),
('when', 'enwhay'), ('rate', 'ateray'), ('sure', 'uresay'), ('land', 'andlay'),
('pink', 'inkpay'), ('hint', 'inthay'), ('cast', 'astcay'), ('park', 'arkpay'),
('bold', 'oldbay'), ('door', 'oorday'), ('bulk', 'ulkbay'), ('doat', 'oatday'),
('coat', 'oatcay'), ('date', 'ateday'), ('foot', 'ootfay'), ('hand', 'andhay'),
('damp', 'ampday'), ('pall', 'allpay'), ('goes', 'oesgay'), ('rung', 'ungray'),
('none', 'onenay'), ('game', 'amegay'), ('nods', 'odsnay'), ('read', 'eadray'),
('gold', 'oldgay'), ('sing', 'ingsay'), ('puts', 'utspay'), ('cats', 'atscay'),
('many', 'anymay'), ('dare', 'areday'), ('noon', 'oonnay'), ('said', 'aidsay'),
('mark', 'arkmay'), ('case', 'asecay'), ('hate', 'atehay'), ('word', 'ordway'),
('beat', 'eatbay'), ('lose', 'oselay'), ('fish', 'ishfay'), ('laid', 'aidlay'),
('fine', 'inefay'), ('snug', 'ugsnay'), ('moss', 'ossmay'), ('heir', 'eirhay'),
('sigh', 'ighsay'), ('mend', 'endmay'), ('sake', 'akesay'), ('pass', 'asspay'),
('debt', 'ebtday'), ('lane', 'anelay'), ('days', 'aysday'), ('deep', 'eepday'),
('song', 'ongsay'), ('spot', 'otspay'), ('took', 'ooktay'), ('both', 'othbay'),
('year', 'yearay'), ('mama', 'amamay'), ('hill', 'illhay'), ('beds', 'edsbay'),
('less', 'esslay'), ('gown', 'owngay'), ('kill', 'illkay'), ('free', 'eefray'),
('hall', 'allhay'), ('room', 'oomray'), ('gray', 'aygray'), ('beef', 'eefbay'),
('kept', 'eptkay'), ('what', 'atwhay'), ('blue', 'ueblay'), ('fell', 'ellfay'),
('draw', 'awdray'), ('play', 'ayplay'), ('baby', 'abybay'), ('rude', 'uderay'),
('trim', 'imtray'), ('chat', 'atchay'), ('bely', 'elybay'), ('fled', 'edflay'),
('past', 'astpay'), ('wait', 'aitway'), ('meal', 'ealmay'), ('wont', 'ontway'),
('week', 'eekway'), ('best', 'estbay'), ('line', 'inelay'), ('lawn', 'awnlay'),
('tree', 'eetray'), ('tend', 'endtay'), ('find', 'indfay'), ('five', 'ivefay'),
('gone', 'onegay'), ('need', 'eednay'), ('drew', 'ewdray'), ('more', 'oremay'),
('cool', 'oolcay'), ('suit', 'uitsay'), ('from', 'omfray'), ('cure', 'urecay'),
('will', 'illway'), ('slow', 'owslay'), ('they', 'eythay'), ('name', 'amenay'),
('shot', 'otshay'), ('come', 'omecay'), ('tied', 'iedtay'), ('came', 'amecay'),
('boys', 'oysbay'), ('wish', 'ishway'), ('hour', 'ourhay'), ('kind', 'indkay'),
('mile', 'ilemay'), ('quit', 'itquay'), ('john', 'ohnjay'), ('hazy', 'azyhay'),
('fall', 'allfay'), ('sell', 'ellsay'), ('tale', 'aletay'), ('grew', 'ewgray'),
('seek', 'eeksay'), ('muff', 'uffmay'), ('soft', 'oftsay'), ('same', 'amesay'),
('body', 'odybay'), ('head', 'eadhay'), ('like', 'ikelay'), ('here', 'erehay'),
('help', 'elphay'), ('sail', 'ailsay'), ('pope', 'opepay'), ('deny', 'enyday'),
('bond', 'ondbay'), ('bath', 'athbay'), ('blow', 'owblay'), ('hold', 'oldhay'),
('safe', 'afesay'), ('path', 'athpay'), ('feed', 'eedfay'), ('thus', 'usthay'),
('four', 'ourfay'), ('lest', 'estlay'), ('thin', 'inthay')], (11, 14):
[('approaching', 'approachingway'), ('unequivocal', 'unequivocalway'),
('exceedingly', 'exceedinglyway'), ('application', 'applicationway'),

```
('observation', 'observationway'), ('unhappiness', 'unhappinessway'),
('interrupted', 'interruptedway'), ('enlightened', 'enlightenedway'),
('appearances', 'appearancesway'), ('unconnected', 'unconnectedway'),
('unconcerned', 'unconcernedway'), ('indignation', 'indignationway'),
('accommodate', 'accommodateway'), ('inattention', 'inattentionway'),
('undoubtedly', 'undoubtedlyway'), ('intercourse', 'intercourseway'),
('imperfectly', 'imperfectlyway'), ('imprudently', 'imprudentlyway'),
('affectation', 'affectationway'), ('instigation', 'instigationway'),
('inclination', 'inclinationway'), ('undervalued', 'undervaluedway'),
('affirmative', 'affirmativeway'), ('inconstancy', 'inconstancyway'),
('universally', 'universallyway'), ('engagements', 'engagementsway'),
('exclamation', 'exclamationway'), ('elucidation', 'elucidationway'),
('accustomary', 'accustomaryway'), ('approbation', 'approbationway'),
('unnecessary', 'unnecessaryway'), ('associating', 'associatingway'),
('attributing', 'attributingway'), ('unobtrusive', 'unobtrusiveway'),
('impropriety', 'improprietyway'), ('examination', 'examinationway'),
('alterations', 'alterationsway'), ('unexpressed', 'unexpressedway'),
('abridgement', 'abridgementway'), ('exhilarated', 'exhilaratedway'),
('unsuspected', 'unsuspectedway'), ('independent', 'independentway'),
('alternately', 'alternatelyway'), ('enforcement', 'enforcementway'),
('undertaking', 'undertakingway'), ('outstripped', 'outstrippedway'),
('appointment', 'appointmentway'), ('explanation', 'explanationway'),
('experienced', 'experiencedway'), ('unknowingly', 'unknowinglyway'),
('undesirable', 'undesirableway'), ('ascertained', 'ascertainedway'),
('enumeration', 'enumerationway'), ('alleviation', 'alleviationway'),
('instruction', 'instructionway'), ('undeserving', 'undeservingway'),
('earnestness', 'earnestnessway'), ('insincerity', 'insincerityway'),
('accompanied', 'accompaniedway'), ('employments', 'employmentsway'),
('inferiority', 'inferiorityway'), ('inexcusable', 'inexcusableway'),
('introducing', 'introducingway'), ('extravagant', 'extravagantway'),
('impatiently', 'impatientlyway'), ('attentively', 'attentivelyway'),
('eligibility', 'eligibilityway'), ('uncertainty', 'uncertaintyway')], (7, 10):
[('anxiety', 'anxietyway'), ('attacks', 'attacksway'), ('ailment',
'ailmentway'), ('awkward', 'awkwardway'), ('effects', 'effectsway'), ('induced',
'inducedway'), ('uncivil', 'uncivilway'), ('against', 'againstway'), ('apology',
'apologyway'), ('endless', 'endlessway'), ('alluded', 'alludedway'), ('ability',
'abilityway'), ('address', 'addressway'), ('engaged', 'engagedway'), ('eagerly',
'eagerlyway'), ('expects', 'expectsway'), ('elliott', 'elliottway'), ('oneself',
'oneselfway'), ('anybody', 'anybodyway'), ('augment', 'augmentway'), ('utility',
'utilityway'), ('unquiet', 'unquietway'), ('infancy', 'infancyway'), ('altered',
'alteredway'), ('unequal', 'unequalway'), ('adapted', 'adaptedway'), ('useless',
'uselessway'), ('opposed', 'opposedway'), ('artless', 'artlessway'), ('imagery',
'imageryway'), ('amiably', 'amiablyway'), ('uncouth', 'uncouthway'), ('affixed',
'affixedway'), ('impulse', 'impulseway'), ('opinion', 'opinionway'), ('imitate',
'imitateway'), ('admires', 'admiresway'), ('affront', 'affrontway'), ('amusing',
'amusingway'), ('accents', 'accentsway'), ('implied', 'impliedway'), ('ecstasy',
'ecstasyway'), ('illness', 'illnessway'), ('entreat', 'entreatway'), ('another',
'anotherway'), ('opening', 'openingway'), ('england', 'englandway'), ('ensured',
```

'ensuredway'), ('imputed', 'imputedway'), ('enabled', 'enabledway'), ('offence', 'offenceway'), ('insured', 'insuredway'), ('urgency', 'urgencyway'), ('amazing', 'amazingway'), ('uniform', 'uniformway'), ('avarice', 'avariceway'), ('ellison', 'ellisonway'), ('endowed', 'endowedway'), ('ashamed', 'ashamedway'), ('eloping', 'elopingway'), ('inhabit', 'inhabitway'), ('advised', 'advisedway'), ('attract', 'attractway'), ('already', 'alreadyway'), ('enquire', 'enquireway'), ('affects', 'affectsway'), ('invalid', 'invalidway')], (9, 11): [('genuinely', 'enuinelygay'), ('columella', 'olumellacay'), ('vouchsafe', 'ouchsafevay'), ('forwarded', 'orwardedfay'), ('scattered', 'atteredscay'), ('christmas', 'istmaschray'), ('crimsoned', 'imsonedcray'), ('contrives', 'ontrivescay'), ('miserable', 'iserablemay'), ('preceding', 'ecedingpray'), ('healthful', 'ealthfulhay'), ('confiding', 'onfidingcay'), ('perfectly', 'erfectlypay'), ('prospects', 'ospectspray'), ('withdrawn', 'ithdrawnway'), ('lingering', 'ingeringlay'), ('contained', 'ontainedcay'), ('depressed', 'epressedday'), ('disgusted', 'isgustedday'), ('promoting', 'omotingpray'), ('supported', 'upportedsay'), ('gradually', 'aduallygray'), ('disturbed', 'isturbedday'), ('perpetual', 'erpetualpay'), ('pronounce', 'onouncepray'), ('sincerely', 'incerelysay'), ('seventeen', 'eventeensay'), ('presented', 'esentedpray'), ('medicines', 'edicinesmay'), ('syllables', 'yllablessay'), ('behaviour', 'ehaviourbay'), ('niggardly', 'iggardlynay'), ('commonest', 'ommonestcay'), ('necessary', 'ecessarynay'), ('prettiest', 'ettiestpray'), ('condemned', 'ondemnedcay'), ('conquests', 'onquestscay'), ('fashioned', 'ashionedfay'), ('hysterics', 'ystericshay'), ('smallness', 'allnesssmay'), ('professed', 'ofessedpray'), ('conceited', 'onceitedcay'), ('compliant', 'ompliantcay'), ('frankness', 'anknessfray'), ('merriment', 'errimentmay'), ('resembled', 'esembledray'), ('leisurely', 'eisurelylay'), ('dispersed', 'ispersedday'), ('departure', 'epartureday'), ('pecuniary', 'ecuniarypay'), ('frivolous', 'ivolousfray'), ('situation', 'ituationsay'), ('discourse', 'iscourseday'), ('resettled', 'esettledray'), ('beholding', 'eholdingbay'), ('gentleman', 'entlemangay'), ('chagrined', 'agrinedchay'), ('struggled', 'uggledstray'), ('preparing', 'eparingpray'), ('respected', 'espectedray'), ('newspaper', 'ewspapernay'), ('justified', 'ustifiedjay'), ('sacrifice', 'acrificesay'), ('stammered', 'ammeredstay'), ('smartness', 'artnesssmay'), ('continued', 'ontinuedcay'), ('contrived', 'ontrivedcay'), ('reconcile', 'econcileray'), ('modestest', 'odestestmay'), ('cordially', 'ordiallycay'), ('sentiment', 'entimentsay'), ('shameless', 'amelessshay'), ('reflected', 'eflectedray'), ('feelingly', 'eelinglyfay'), ('confident', 'onfidentcay'), ('proposals', 'oposalspray'), ('warehouse', 'arehouseway'), ('deviation', 'eviationday'), ('statement', 'atementstay'), ('submitted', 'ubmittedsay'), ('believing', 'elievingbay'), ('gentility', 'entilitygay'), ('resources', 'esourcesray'), ('composure', 'omposurecay'), ('deserving', 'eservingday'), ('vehemence', 'ehemencevay'), ('disgraced', 'isgracedday'), ('perplexed', 'erplexedpay'), ('prejudice', 'ejudicepray'), ('quickened', 'ickenedquay'), ('blundered', 'underedblay'), ('strongest', 'ongeststray'), ('prevailed', 'evailedpray'), ('languages', 'anguageslay'), ('reanimate', 'eanimateray'), ('stationer', 'ationerstay'), ('reminding', 'emindingray'), ('blameless', 'amelessblay'), ('principle', 'inciplepray'), ('hardening', 'ardeninghay'), ('suggested', 'uggestedsay'), ('desperate', 'esperateday'), ('landscape', 'andscapelay'),

```
('paragraph', 'aragraphpay'), ('provision', 'ovisionpray'), ('relations',
'elationsray'), ('fruitless', 'uitlessfray'), ('christian', 'istianchray'),
('stipulate', 'ipulatestay'), ('constancy', 'onstancycay'), ('gentlemen',
'entlemengay'), ('venturing', 'enturingvay'), ('northward', 'orthwardnay'),
('neighbour', 'eighbournay'), ('weakening', 'eakeningway'), ('persisted',
'ersistedpay'), ('malignant', 'alignantmay'), ('wandering', 'anderingway'),
('witnesses', 'itnessesway'), ('dangerous', 'angerousday'), ('safeguard',
'afeguardsay'), ('treasured', 'easuredtray'), ('cleveland', 'evelandclay'),
('patronage', 'atronagepay'), ('despising', 'espisingday'), ('foolishly',
'oolishlyfay'), ('seriously', 'eriouslysay'), ('restraint', 'estraintray'),
('september', 'eptembersay'), ('distrusts', 'istrustsday'), ('temperate',
'emperatetay'), ('finishing', 'inishingfay'), ('persuaded', 'ersuadedpay'),
('stupified', 'upifiedstay'), ('declaring', 'eclaringday'), ('dejection',
'ejectionday'), ('soothings', 'oothingssay'), ('conveying', 'onveyingcay'),
('possessed', 'ossessedpay'), ('described', 'escribedday'), ('spreading',
'eadingspray'), ('foresight', 'oresightfay'), ('political', 'oliticalpay'),
('concluded', 'oncludedcay'), ('doubtless', 'oubtlessday'), ('confusion',
'onfusioncay'), ('disliking', 'islikingday'), ('carriages', 'arriagescay'),
('favourite', 'avouritefay'), ('brilliant', 'illiantbray'), ('hastening',
'asteninghay'), ('precisely', 'eciselypray'), ('wondering', 'onderingway'),
('shoulders', 'oulderssshay'), ('honorably', 'onorablyhay'), ('privilege',
'ivilegepray'), ('deference', 'eferenceday'), ('vulgarity', 'ulgarityvay'),
('reprimand', 'eprimandray'), ('surveying', 'urveyingsay'), ('handsomer',
'andsomerhay'), ('shivering', 'iveringshay'), ('tolerable', 'olerabletay'),
('requiring', 'equiringray'), ('precision', 'ecisionpray'), ('courtland',
'ourtlandcay'), ('sportsman', 'ortsmanspay'), ('happiness', 'appinesshay'),
('pointedly', 'ointedlypay'), ('benefited', 'enefitedbay'), ('substance',
'ubstancesay'), ('beginning', 'eginningbay'), ('gratitude', 'atitudegray'),
('negligent', 'egligentnay'), ('belonging', 'elongingbay'), ('mortified',
'ortifiedmay'), ('neglected', 'eglectednay'), ('shillings', 'illingsshay'),
('warranted', 'arrantedway'), ('performed', 'erformedpay'), ('necessity',
'ecessitynay'), ('slightest', 'ightestslay'), ('guardians', 'uardiansgay'),
('preferred', 'eferredpray'), ('forgotten', 'orgottenfay'), ('refreshed',
'efreshedray'), ('barouches', 'arouchesbay'), ('grievance', 'ievancegray'),
('seduction', 'eductionsay'), ('comprised', 'omprisedcay'), ('listening',
'isteninglay'), ('detection', 'etectionday'), ('narration', 'arrationnay'),
('deceiving', 'eceivingday'), ('provoking', 'ovokingpray'), ('cessation',
'essationcay'), ('questions', 'estionsquay'), ('continual', 'ontinualcay'),
('faintness', 'aintnessfay'), ('diffident', 'iffidentday'), ('referring',
'eferringray'), ('blessings', 'essingsblay'), ('therefore', 'ereforethay'),
('frugality', 'ugalityfray'), ('reprobate', 'eprobateray')], (11, 13):
[('tranquilize', 'anquilizetray'), ('fashionable', 'ashionablefay'),
('respectable', 'espectableray'), ('publication', 'ublicationpay'),
('superintend', 'uperintendsay'), ('recollected', 'ecollectedray'),
('cheerfuller', 'eerfullerchay'), ('concurrence', 'oncurrencecay'),
('restorative', 'estorativeray'), ('reproaching', 'eproachingray'),
('declaration', 'eclarationday'), ('discharging', 'ischargingday'),
('proclaiming', 'oclaimingpray'), ('discovering', 'iscoveringday'),
```

```
('possibility', 'ossibilitypay'), ('contributed', 'ontributedcay'),
('captivating', 'aptivatingcay'), ('misinformed', 'isinformedmay'),
('carefulness', 'arefulnesscay'), ('philosophic', 'ilosophicphay'),
('calculation', 'alculationcay'), ('consumption', 'onsumptioncay'),
('confinement', 'onfinementcay'), ('certainties', 'ertaintiescay'),
('destination', 'estinationday'), ('separations', 'eparationssay'),
('voluntarily', 'oluntarilyvay'), ('connoisseur', 'onnoisseurcay'),
('refinements', 'efinementsray'), ('subservient', 'ubservientsay'),
('marlborough', 'arlboroughmay'), ('sagaciously', 'agaciouslysay'),
('fascinating', 'ascinatingfay'), ('comfortable', 'omfortablecay'),
('reflections', 'eflectionsray'), ('sensibility', 'ensibilitysay'),
('forbearance', 'orbearancefay'), ('mischievous', 'ischievousmay'),
('disgraceful', 'isgracefulday'), ('temporizing', 'emporizingtay'),
('sacrificing', 'acrificingsay'), ('spontaneous', 'ontaneousspay'),
('flourishing', 'ourishingflay'), ('lamentation', 'amentationlay'),
('meditations', 'editationsmay'), ('quarrelling', 'arrellingquay'),
('discussions', 'iscussionsday'), ('supposition', 'uppositionsay'),
('comfortably', 'omfortablycay'), ('disappeared', 'isappearedday'),
('preparation', 'eparationpray'), ('consistency', 'onsistencycay'),
('compliments', 'omplimentscay'), ('promptitude', 'omptitudepray'),
('determining', 'eterminingday'), ('countenance', 'ountenancecay'),
('maintenance', 'aintenancemay'), ('contraction', 'ontractioncay'),
('dissipation', 'issipationday'), ('comparative', 'omparativecay'),
('desperation', 'esperationday'), ('displeasure', 'ispleasureday'),
('despatching', 'espatchingday'), ('playfellows', 'ayfellowsplay'),
('plantations', 'antationsplay'), ('vindication', 'indicationvay'),
('significant', 'ignificantsay'), ('discontents', 'iscontentsday'),
('housekeeper', 'ousekeeperhay'), ('playfulness', 'ayfulnessplay'),
('threatening', 'eateningthray'), ('meritorious', 'eritoriousmay'),
('tediousness', 'ediousnesstay'), ('convenience', 'onveniencecay'),
('possessions', 'ossessionspay'), ('consolation', 'onsolationcay'),
('composition', 'ompositioncay'), ('proposition', 'opositionpray'),
('correctness', 'orrectnesscay'), ('consequence', 'onsequencecay'),
('barbarously', 'arbarouslybay'), ('booksellers', 'ooksellersbay'),
('conjectures', 'onjecturescay'), ('violoncello', 'ioloncellovay'),
('dissembling', 'issemblingday'), ('rapturously', 'apturouslyray'),
('speculative', 'eculativespay'), ('remembering', 'ememberingray'),
('probability', 'obabilitypray'), ('punctuality', 'unctualitypay'),
('forgiveness', 'orgivenessfay'), ('disapproves', 'isapprovesday'),
('troublesome', 'oublesometray')], (3, 5): [('say', 'aysay'), ('for', 'orfay'),
('get', 'etgay'), ('pin', 'inpay'), ('not', 'otnay'), ('fit', 'itfay'), ('rap',
'apray'), ('lip', 'iplay'), ('pry', 'ypray'), ('rob', 'obray'), ('fir',
'irfay'), ('can', 'ancay'), ('two', 'otway'), ('him', 'imhay'), ('son',
'onsay'), ('let', 'etlay'), ('try', 'ytray'), ('hot', 'othay'), ('sum',
'umsay'), ('got', 'otgay'), ('law', 'awlay'), ('how', 'owhay'), ('far',
'arfay'), ('the', 'ethay'), ('sky', 'yskay'), ('sly', 'yslay'), ('cut',
'utcay'), ('hop', 'ophay'), ('had', 'adhay'), ('was', 'asway'), ('bow',
'owbay'), ('hue', 'uehay'), ('now', 'ownay'), ('vex', 'exvay'), ('yet',
```

'yetay'), ('sit', 'itsay'), ('due', 'ueday'), ('cap', 'apcay'), ('sat',
'atsay'), ('hit', 'ithay'), ('see', 'eesay'), ('pay', 'aypay'), ('nor',
'ornay'), ('fly', 'yflay'), ('dog', 'ogday'), ('too', 'ootay'), ('six',
'ixsay'), ('lap', 'aplay'), ('new', 'ewnay'), ('led', 'edlay'), ('sin',
'insay'), ('ten', 'entay'), ('his', 'ishay'), ('day', 'ayday')], (14, 17):
[('insurmountable', 'insurmountableway'), ('inconveniences',
'inconveniencesway'), ('aggrandizement', 'aggrandizementway'),
('irreconcilable', 'irreconcilableway'), ('unacknowledged',
'unacknowledgedway'), ('unpleasantness', 'unpleasantnessway')], (13, 16):
[('understanding', 'understandingway'), ('intentionally', 'intentionallyway'),
('embarrassment', 'embarrassmentway'), ('unembarrassed', 'unembarrassedway'),
('administering', 'administeringway'), ('instantaneous', 'instantaneousway'),
('intrinsically', 'intrinsicallyway'), ('indispensable', 'indispensableway'),
('impoverishing', 'impoverishingway'), ('opportunities', 'opportunitiesway'),
('impertinently', 'impertinentlyway'), ('independently', 'independentlyway'),
('encroachments', 'encroachmentsway'), ('uninteresting', 'uninterestingway'),
('investigation', 'investigationway'), ('unaccountable', 'unaccountableway'),
('irrepressible', 'irrepressibleway'), ('impossibility', 'impossibilityway')],
(10, 14): [('thirty-six', 'irtythay-ixsay')], (8, 11): [('increase',
'increaseway'), ('agitated', 'agitatedway'), ('unkindly', 'unkindlyway'),
('occurred', 'occurredway'), ('arranged', 'arrangedway'), ('appetite',
'appetiteway'), ('inclined', 'inclinedway'), ('indolent', 'indolentway'),
('inquired', 'inquiredway'), ('evenings', 'eveningsway'), ('obliging',
'obligingway'), ('equalled', 'equalledway'), ('accepted', 'acceptedway'),
('extended', 'extendedway'), ('entreaty', 'entreatyway'), ('exigence',
'exigenceway'), ('election', 'electionway'), ('examined', 'examinedway'),
('inviting', 'invitingway'), ('embitter', 'embitterway'), ('idleness',
'idlenessway'), ('admirers', 'admirersway'), ('editions', 'editionsway'),
('extorted', 'extortedway'), ('outweigh', 'outweighway'), ('attended',
'attendedway'), ('addition', 'additionway'), ('entitled', 'entitledway'),
('awakened', 'awakenedway'), ('attacked', 'attackedway'), ('arriving',
'arrivingway'), ('overlook', 'overlookway'), ('eminence', 'eminenceway'),
('adhering', 'adheringway'), ('obtained', 'obtainedway'), ('openness',
'opennessway'), ('allusion', 'allusionway'), ('anything', 'anythingway'),
('overcome', 'overcomeway'), ('announce', 'announceway'), ('enjoying',
'enjoyingway'), ('outlived', 'outlivedway'), ('emphasis', 'emphasisway'),
('insulted', 'insultedway'), ('esteemed', 'esteemedway'), ('alphabet',
'alphabetway'), ('enquired', 'enquiredway'), ('accident', 'accidentway'),
('occupied', 'occupiedway'), ('ellisons', 'ellisonsway'), ('avoiding',
'avoidingway'), ('attained', 'attainedway'), ('unwarily', 'unwarilyway'),
('employed', 'employedway'), ('altering', 'alteringway'), ('intimate',
'intimateway'), ('effected', 'effectedway'), ('offences', 'offencesway'),
('entirely', 'entirelyway'), ('abounded', 'aboundedway'), ('actually',
'actuallyway'), ('identify', 'identifyway'), ('applying', 'applyingway'),
('exterior', 'exteriorway'), ('auditors', 'auditorsway'), ('unfairly',
'unfairlyway'), ('eighteen', 'eighteenway'), ('embraced', 'embracedway'),
('unfolded', 'unfoldedway'), ('injuries', 'injuriesway'), ('elevated',
'elevatedway'), ('alarming', 'alarmingway'), ('adequate', 'adequateway'),

```
('unabated', 'unabatedway'), ('incurred', 'incurredway'), ('uncommon',
'uncommonway'), ('unawares', 'unawaresway'), ('unsuited', 'unsuitedway'),
('instance', 'instanceway'), ('interval', 'intervalway'), ('audacity',
'audacityway')], (5, 8): [('other', 'otherway'), ('ideas', 'ideasway'),
('agony', 'agonyway'), ('avoid', 'avoidway'), ('awful', 'awfulway'), ('april',
'aprilway'), ('enter', 'enterway'), ('infer', 'inferway'), ('abuse',
'abuseway'), ('ivory', 'ivoryway'), ('adopt', 'adoptway'), ('image',
'imageway'), ('agree', 'agreeway'), ('alike', 'alikeway'), ('earth',
'earthway'), ('ended', 'endedway'), ('owner', 'ownerway'), ('using',
'usingway'), ('ought', 'oughtway'), ('abhor', 'abhorway'), ('altar',
'altarway'), ('often', 'oftenway'), ('atone', 'atoneway'), ('angel',
'angelway'), ('arise', 'ariseway'), ('under', 'underway'), ('abode',
'abodeway'), ('exert', 'exertway'), ('usual', 'usualway'), ('aches',
'achesway'), ('enjoy', 'enjoyway'), ('urged', 'urgedway'), ('admit',
'admitway'), ('aside', 'asideway'), ('equal', 'equalway'), ('alive',
'aliveway'), ('arose', 'aroseway'), ('eight', 'eightway'), ('opera',
'operaway'), ('ennui', 'ennuiway'), ('utter', 'utterway'), ('ankle',
'ankleway')], (2, 5): [('is', 'isway'), ('ay', 'ayway'), ('ah', 'ahway'), ('ii',
'iiway'), ('or', 'orway'), ('am', 'amway'), ('if', 'ifway'), ('an', 'anway')],
(12, 14): [('tranquillity', 'anquillitytray'), ('humiliations',
'umiliationshay'), ('contributing', 'ontributingcay'), ('significancy',
'ignificancysay'), ('wretchedness', 'etchednesswray'), ('prodigiously',
'odigiouslypray'), ('particularly', 'articularlypay'), ('carelessness',
'arelessnesscay'), ('refreshments', 'efreshmentsray'), ('satisfaction',
'atisfactionsay'), ('construction', 'onstructioncay'), ('contribution',
'ontributioncay'), ('gracefulness', 'acefulnessgray'), ('consequences',
'onsequencescay'), ('grandmothers', 'andmothersgray'), ('proportioned',
'oportionedpray'), ('perverseness', 'erversenesspay'), ('conciliation',
'onciliationcay'), ('representing', 'epresentingray'), ('watchfulness',
'atchfulnessway'), ('recollecting', 'ecollectingray'), ('hesitatingly',
'esitatinglyhay'), ('pleasantness', 'easantnessplay'), ('benefactress',
'enefactressbay'), ('distractedly', 'istractedlyday'), ('perseverance',
'erseverancepay'), ('performances', 'erformancespay'), ('disapproving',
'isapprovingday'), ('recollection', 'ecollectionray'), ('comprehended',
'omprehendedcay'), ('relinquished', 'elinquishedray'), ('conjecturing',
'onjecturingcay'), ('reservedness', 'eservednessray'), ('promontories',
'omontoriespray'), ('discontented', 'iscontentedday'), ('persecutions',
'ersecutionspay'), ('relationship', 'elationshipray'), ('triumphantly',
'iumphantlytray'), ('deliberation', 'eliberationday'), ('supplication',
'upplicationsay'), ('dishonorable', 'ishonorableday'), ('propensities',
'opensitiespray'), ('contemptuous', 'ontemptuouscay'), ('transporting',
'ansportingtray')], (13, 15): [('distinguished', 'istinguishedday'),
('resuscitation', 'esuscitationray'), ('particularity', 'articularitypay'),
('superannuated', 'uperannuatedsay'), ('disagreements', 'isagreementsday'),
('gentlemanlike', 'entlemanlikegay'), ('prepossessing', 'epossessingpray'),
('conscientious', 'onscientiouscay'), ('considerately', 'onsideratelycay'),
('felicitations', 'elicitationsfay'), ('prepossession', 'epossessionpray'),
('consternation', 'onsternationcay'), ('disproportion', 'isproportionday'),
```

('commiseration', 'ommiserationcay'), ('mortification', 'ortificationmay'), ('solicitations', 'olicitationssay'), ('qualification', 'alificationquay'), ('philanthropic', 'ilanthropicphay'), ('disappointing', 'isappointingday'), ('prescriptions', 'escriptionspray'), ('strengthening', 'engtheningstray')], (13, 17): [('mulberry-tree', 'ulberrymay-eetray'), ('good-humoured', 'oodgay-umouredhay'), ('mean-spirited', 'eanmay-iritedspay'), ('self-reproach', 'elfsay-eproachray'), ('drawing-table', 'awingdray-abletay'), ('well-disposed', 'ellway-isposedday'), ('proud-looking', 'oudpray-ookinglay'), ('flower-garden', 'owerflay-ardengay'), ('mansion-house', 'ansionmay-ousehay'), ('landing-place', 'andinglay-aceplay'), ('music-sellers', 'usicmay-ellerssay')], (4, 7): [('acts', 'actsway'), ('anne', 'anneway'), ('upon', 'uponway'), ('able', 'ableway'), ('inch', 'inchway'), ('ages', 'agesway'), ('envy', 'envyway'), ('idea', 'ideaway'), ('ends', 'endsway'), ('-and', '-andway'), ('eyed', 'eyedway'), ('arms', 'armsway'), ('over', 'overway'), ('easy', 'easyway'), ('ever', 'everway'), ('ours', 'oursway'), ('ears', 'earsway'), ('away', 'awayway')], (16, 20): [('pleasure-grounds', 'easureplay-oundsgray'), ('self-provocation', 'elfsay-ovocationpray')], (11, 15): [('bed-chamber', 'edbay-amberchay'), ('high-minded', 'ighhay-indedmay'), ('before-hand', 'eforebay-andhay'), ('watch-tower', 'atchway-owertay'), ('print-shops', 'intpray-opsshay'), ('card-tables', 'ardcay-ablestay'), ('post-horses', 'ostpay-orseshay'), ('good-nature', 'oodgay-aturenay'), ('dining-room', 'iningday-oomray'), ('high-church', 'ighhay-urchchay'), ('well-wisher', 'ellway-isherway'), ('bond-street', 'ondbay-eetstray'), ('carpet-work', 'arpetcay-orkway')], (2, 4): [('to', 'otay'), ('he', 'ehay'), ('no', 'onay'), ('me', 'emay')], (12, 16): [('good-hearted', 'oodgay-eartedhay'), ('well-judging', 'ellway-udgingjay'), ('drawing-room', 'awingdray-oomray'), ('common-place', 'ommoncay-aceplay'), ('self-control', 'elfsay-ontrolcay'), ('hand-writing', 'andhay-itingwray'), ('well-behaved', 'ellway-ehavedbay'), ('mantel-piece', 'antelmay-iecepay')], (10, 15): [('ill-humour', 'illway-umourhay'), ('ill-judged', 'illway-udgedjay'), ('ill-suited', 'illway-uitedsay')], (1, 3): [('c', 'cay')], (8, 13): [('ill-wind', 'illway-indway'), ('ill-will', 'illway-illway'), ('over-set', 'overway-etsay')], (17, 19): [('disqualifications', 'isqualificationsday')], (9, 16): [('by-and-by', 'ybay-andway-ybay')], (3, 6): [('aye', 'ayeway'), ('use', 'useway'), ('owe', 'oweway'), ('our', 'ourway'), ('old', 'oldway'), ('ear', 'earway'), ('its', 'itsway'), ('ill', 'illway')], (12, 17): [('ill-breeding', 'illway-eedingbray'), ('re-assurance', 'eray-assuranceway'), ('over-powered', 'overway-oweredpay'), ('ill-disposed', 'illway-isposedday')], (11, 16): [('ill-natured', 'illway-aturednay')], (14, 16): [('reconciliation', 'econciliationray'), ('recommendation', 'ecommendationray'), ('reasonableness', 'easonablenessray'), ('disinclination', 'isinclinationday'), ('respectability', 'espectabilityray'), ('circumspection', 'ircumspectioncay'), ('considerations', 'onsiderationscay'), ('qualifications', 'alificationsquay'), ('congratulating', 'ongratulatingcay')], (15, 17): [('dissatisfaction', 'issatisfactionday'), ('disrespectfully', 'isrespectfullyday'), ('conscientiously', 'onscientiouslycay'), ('misconstruction', 'isconstructionmay')], (15, 18): [('instantaneously', 'instantaneouslyway')], (15, 19): [('shooting-jacket', 'ootingshay-acketjay')], (14, 21): [('sisters-in-law', 'isterssay-inway-awlay')], (17, 21): [('self-condemnation', 'elfsay-ondemnationcay')], (13, 20): [('sister-in-law',

```
'istersay-inway-awlay')], (9, 14): [('ill-timed', 'illway-imedtay')], (13, 18):
[('ill-treatment', 'illway-eatmenttray'), ('old-fashioned', 'oldway-
ashionedfay'), ('self-interest', 'elfsay-interestway')], (16, 22): [('good-for-
nothing', 'oodgay-orfay-othingnay')], (9, 13): [('dove-cote', 'oveday-otecay'),
('tea-table', 'eatay-abletay'), ('work-bags', 'orkway-agsbay')], (1, 4): [('i',
'iway')], (16, 19): [('incomprehensible', 'incomprehensibleway')], (8, 12):
[('to-night', 'otay-ightnay'), ('land-tax', 'andlay-axtay')], (11, 18): [('tete-
a-tete', 'etetay-away-etetay')], (5, 9): [('to-do', 'otay-oday')]})
Moved models to GPU!
Epoch:   0 | Flops (T): 0.091202679072, Tokens (M) 0.052678, Train Loss 2.940,
Val Loss 2.960 | Gen: aaaaaaaaaaaaaaaaaaaa aaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaa aaaaaaaaaaaaaaaaaaaa aaaaaaaaaaaaaaaaaaaa
Epoch:   1 | Flops (T): 0.182405358144, Tokens (M) 0.105356, Train Loss 2.859,
Val Loss 2.842 | Gen: aaaaaaaaaaaaaaaaaaaa yyyyyyyyyyyyyyyyyyyy
aaaaaaaaaaaaaaaaaaaa aaaaaaaaaaaaaaaaaaaa yyyyyyyyyyyyyyyyyyyy
Epoch:   2 | Flops (T): 0.273608037216, Tokens (M) 0.158034, Train Loss 2.744,
Val Loss 2.766 | Gen: hhhhhhhhhhhhhhhhhhh yyyyyyyyyaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaa aaaaaaaaaaaaaaaaaaaa aaaaayyyyaaaaaaaaaaa
Epoch:   3 | Flops (T): 0.364810716288, Tokens (M) 0.210712, Train Loss 2.367,
Val Loss 2.211 | Gen: eeeeeaaaaaaaaaaaaaaa iiiiiiiiiiiiiiiiiiii
oooooooooooooooooooo iiiiiiiiiiiiiiiiiiii iiiiiiiiiiiiiiiiiiii
Epoch:   4 | Flops (T): 0.45601339536, Tokens (M) 0.26339, Train Loss 2.056, Val
Loss 2.126 | Gen: eeeeeeeaaaaaaaaaaaaa wwrrrraaaaaaaaaaaaaa iiiiiiiiiiiiiiiiiiii
wwwisssssssssssssss rrrrrwwwwwwwwwwwwwww
Epoch:   5 | Flops (T): 0.547216074432, Tokens (M) 0.316068, Train Loss 1.985,
Val Loss 2.058 | Gen: teeeeeeaaaaaaaaaaaaa iiiiiiiwwwwwwwwwwaa
iiiiiiiiiiiiiiiiiiii iiiiiiiiiiiaaaaaaaaa iiiiiiiiiiiiiiiiiiiw
Epoch:   6 | Flops (T): 0.638418753504, Tokens (M) 0.368746, Train Loss 1.926,
Val Loss 2.024 | Gen: eeeeeeayyyyyyyyyyyyy wwwwwwyyyyyyyyyyyyy
iiiiiiiiiiiiiiiinnnn iisssssyyyyyyyyyyyy iiiggggggggggggggay
Epoch:   7 | Flops (T): 0.729621432576, Tokens (M) 0.421424, Train Loss 1.873,
Val Loss 1.977 | Gen: aaaaaaaaaaaaaaaaaaaa wwwwwyyyyyyyyyyyyyy
iiiiiiiiiiinnnnnnnnn iissssyyyyyyyyyyyyy rrrrwgggwwwwwwwwwwww
Epoch:   8 | Flops (T): 0.820824111648, Tokens (M) 0.474102, Train Loss 1.786,
Val Loss 1.949 | Gen: teeeeeeyyyyyyyyyyyy awwwwwwwwwwwwwwwwwww
iiiinnnnnnnnnnnnnnn wwwwwwwwwwwwwwwwwwww wwwwwwwwwwwwwaaayyyy
Epoch:   9 | Flops (T): 0.91202679072, Tokens (M) 0.52678, Train Loss 1.666, Val
Loss 1.831 | Gen: hhhhhhhhhhhhhhhhhhh iiiiwwwwwwwwwwwwwwww oooonnnnnnnnnnnnnnnn
ttttttttttttttttttt oo-iiggggggggggggggg
Epoch:  10 | Flops (T): 1.003229469792, Tokens (M) 0.579458, Train Loss 1.637,
Val Loss 2.067 | Gen: yyyyyyyyyyyyyyyyyyyy wwwwwwwwwwwwwwwwwwww
yyyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyyy
Epoch:  11 | Flops (T): 1.094432148864, Tokens (M) 0.632136, Train Loss 1.629,
Val Loss 1.632 | Gen: eeeeeeeeeeeeeeeeeeee wwwwwwwwwwwwwwwwwwww
onnnnnnnnnnnnnnnnnnn wwwwwwwwwwwwwwwwwwww wwwwwwwwwwwwwwwwwwww
Epoch:  12 | Flops (T): 1.185634827936, Tokens (M) 0.684814, Train Loss 1.451,
Val Loss 1.552 | Gen: eeeeeeeeeeeeeeeeeeee wwwwwwwwwwwwwwwwwwww
oiiinnnnnnnnnnnnnnn iiiiiiiiwiyyyyyyyyyy owwwwwwwwwwwwwwwwwww
```

Epoch: 13 | Flops (T): 1.276837507008, Tokens (M) 0.737492, Train Loss 1.600, Val Loss 1.997 | Gen: tttttttttttttttttttt wwwwwwwwwyyyyyyyyyy ooonnnnnnnnnnnnnnnnn ssssssssssysssssssss yyyyyyyyyyyyyyyyyyy
Epoch: 14 | Flops (T): 1.36804018608, Tokens (M) 0.79017, Train Loss 1.670, Val Loss 1.645 | Gen: hhhhhhhhhhhhhhhhhhh rrrrrrwwwwwwwwwwwwww oooiiiiiiiiiiiiiiinnn ssssswwwwwwwwwwwwwww oiiiiiiwwwwwwwwwwwww
Epoch: 15 | Flops (T): 1.459242865152, Tokens (M) 0.842848, Train Loss 1.433, Val Loss 1.472 | Gen: eeeeeeeeyyyyyyyyyyy iiiwwwwwwwwwwwwwwwww oiiiiiiiiiiiiiiiiiii iiiiiwwwyyyyyyyyyyy oooiiiiiiiiiwwwwwwww
Epoch: 16 | Flops (T): 1.550445544224, Tokens (M) 0.895526, Train Loss 1.352, Val Loss 4.309 | Gen: tttttttttttttttttttt tttttttttttttttttttt tttttttttttttttttttt tttttttttttttttttttt tttttttttttttttttttt
Epoch: 17 | Flops (T): 1.641648223296, Tokens (M) 0.948204, Train Loss 1.847, Val Loss 5.150 | Gen: hhhhhhhhhhhhhhhhhhh qaaaaaaaaaaaaaaaaaa ccccccccccccccccccccc ssssssssssjjjjjjjjj w----------
Epoch: 18 | Flops (T): 1.732850902368, Tokens (M) 1.000882, Train Loss 2.437, Val Loss 2.082 | Gen: hhhhhhhhhhhhhhhhhhh yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy
Epoch: 19 | Flops (T): 1.82405358144, Tokens (M) 1.05356, Train Loss 2.025, Val Loss 2.000 | Gen: heeehwwwwwwwwwwwwwww yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy
Epoch: 20 | Flops (T): 1.915256260512, Tokens (M) 1.106238, Train Loss 1.948, Val Loss 1.840 | Gen: hhhhhhhhhhhhhhhhhhh yyyyyyyyyyyyyyyyyyy onnnnnnnnnnnnnnnnnn yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy
Epoch: 21 | Flops (T): 2.006458939584, Tokens (M) 1.158916, Train Loss 1.717, Val Loss 1.689 | Gen: eeeeehhhhhhhhhhhhhh yyyyyyyyyyyyyyyyyyy onnnnnnnnnnnnnnnnnn yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy
Epoch: 22 | Flops (T): 2.097661618656, Tokens (M) 1.211594, Train Loss 1.589, Val Loss 1.628 | Gen: ehhhhhhhhhhhhhhhhhh yyyyyyyyyyyyyyyyyyy nnnnnnnnnnnnnnnnnnn yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy
Epoch: 23 | Flops (T): 2.188864297728, Tokens (M) 1.264272, Train Loss 1.546, Val Loss 1.572 | Gen: ewwwwwwwwwwwwwwwwwww yyyyyyyyyyyyyyyyyyy onnnnnnnnnnnnnnnnnn yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy
Epoch: 24 | Flops (T): 2.2800669768, Tokens (M) 1.31695, Train Loss 1.464, Val Loss 1.442 | Gen: eeeeeeehhhhhhhhhhhh yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy oonnnnnnnnnnnnnnnnn
Epoch: 25 | Flops (T): 2.371269655872, Tokens (M) 1.369628, Train Loss 1.345, Val Loss 1.383 | Gen: eeeeeehhhhhhhhhhhhh yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy
Epoch: 26 | Flops (T): 2.462472334944, Tokens (M) 1.422306, Train Loss 1.324, Val Loss 1.484 | Gen: eeeeeeeeeeeeeewwwwww yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy oooooooooooooooooooo
Epoch: 27 | Flops (T): 2.553675014016, Tokens (M) 1.474984, Train Loss 1.338, Val Loss 1.368 | Gen: eeeeeeeeeeeeeeeeeeee yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy owwwwwwwwwwwwwwwwwww
Epoch: 28 | Flops (T): 2.644877693088, Tokens (M) 1.527662, Train Loss 1.381, Val Loss 1.514 | Gen: eeeeeeeeeeeeeeeeeeee yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy

Epoch: 29 | Flops (T): 2.73608037216, Tokens (M) 1.58034, Train Loss 1.451, Val Loss 1.483 | Gen: eeeeeeeeeeeeeeeeeeee yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy
Epoch: 30 | Flops (T): 2.827283051232, Tokens (M) 1.633018, Train Loss 1.437, Val Loss 1.467 | Gen: eeeeeeeeeeeeeeeeeeee yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy
Epoch: 31 | Flops (T): 2.918485730304, Tokens (M) 1.685696, Train Loss 1.323, Val Loss 1.377 | Gen: eeeeeeeeeeeeeeeeeeee yyyyyyyyyyyyyyyyyyy ooonnnnnnnnnnnnnnnnn yyyyyyyyyyyyyyyyyyy ooooooonnnnnnnnnnnn
Epoch: 32 | Flops (T): 3.009688409376, Tokens (M) 1.738374, Train Loss 1.278, Val Loss 1.342 | Gen: eeeeeeeeeeeeeeeeeeee yyyyyyyyyyyyyyyyyyy onnnnnnnnnnnnnnnnnnn yyyyyyyyyyyyyyyyyyy okkkiiiiiiiiiiiiiiiii
Epoch: 33 | Flops (T): 3.100891088448, Tokens (M) 1.791052, Train Loss 1.261, Val Loss 1.330 | Gen: eeeeeeeeeeeeeeeeeeee yyyyyyyyyyyyyyyyyyy oonnnnnnnnnnnnnnnnnn yyyyyyyyyyyyyyyyyyy ooongggggggggggggggg
Epoch: 34 | Flops (T): 3.19209376752, Tokens (M) 1.84373, Train Loss 1.246, Val Loss 1.322 | Gen: eeeeeeeeeeeeeeeeeeee iwwwwwwwwwwwwwwwwww ooonnnnnnnnnnnnnnnnn yyyyyyyyyyyyyyyyyyy ooooggggggggggggggggg
Epoch: 35 | Flops (T): 3.283296446592, Tokens (M) 1.896408, Train Loss 1.236, Val Loss 1.311 | Gen: eeeeeeeeeeeeeeeeeeee yyyyyyyyyyyyyyyyyyy oooonnnnnnnnnnnnnnnn yyyyyyyyyyyyyyyyyyy oooookknniiiiiiiiggg
Epoch: 36 | Flops (T): 3.374499125664, Tokens (M) 1.949086, Train Loss 1.226, Val Loss 1.299 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii oooonnnnnnnnnnnnnnnn yyyyyyyyyyyyyyyyyyy ooookkgggggggggggggg
Epoch: 37 | Flops (T): 3.465701804736, Tokens (M) 2.001764, Train Loss 1.227, Val Loss 1.295 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii onnnnnnnnnnnnnnnnnn yyyyyyyyyyyyyyyyyyy ooonnnggggggggggggggg
Epoch: 38 | Flops (T): 3.556904483808, Tokens (M) 2.054442, Train Loss 1.243, Val Loss 1.314 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii ooooooooooiiiiiiiii yyyyyyyyyyyyyyyyyyy okkkkkkkkkkkkkkkkkk
Epoch: 39 | Flops (T): 3.64810716288, Tokens (M) 2.10712, Train Loss 1.221, Val Loss 1.293 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii ooonnnnnnnnnniiiiiii yyyyyyyyyyyyyyyyyyy oooggggggggggggggggg
Epoch: 40 | Flops (T): 3.739309841952, Tokens (M) 2.159798, Train Loss 1.205, Val Loss 1.283 | Gen: eeeeeeeeeeeeeeeeeeee wwwwwwwwwwwwwwwwwww oonnnnnnnnnnnnggggg yyyyyyyyyyyyyyyyyyy ookgggggggggggggggggg
Epoch: 41 | Flops (T): 3.830512521024, Tokens (M) 2.212476, Train Loss 1.195, Val Loss 1.281 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii ooooiiiiiiiiiiiiiiii yyyyyyyyyyyyyyyyyyy oooiiiiiiiiiggggggg
Epoch: 42 | Flops (T): 3.921715200096, Tokens (M) 2.265154, Train Loss 1.184, Val Loss 1.274 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii ooooiiiiiiiiiiiiiiii yyyyyyyyyyyyyyyyyyy ooooiiggggggggggggggg
Epoch: 43 | Flops (T): 4.012917879168, Tokens (M) 2.317832, Train Loss 1.177, Val Loss 1.256 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii oonnnnnnnnnnnnnnnnnn yyyyyyyyyyyyyyyyyyy ooiiiiiggggggggggggg
Epoch: 44 | Flops (T): 4.10412055824, Tokens (M) 2.37051, Train Loss 1.175, Val Loss 1.281 | Gen: eeeeeeeeeeeeeeeeeeee iwwwwwwwwwwwwwwwwww oooiiiiiiiiiiiiiiiiii yyyyyyyyyyyyyyyyyyy okkkkgggggggggggggggg

Epoch: 45 | Flops (T): 4.195323237312, Tokens (M) 2.423188, Train Loss 1.181, Val Loss 1.276 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii oooiiiiiiiiiiiiiiiii yyyyyyyyyyyyyyyyyyyy ooiiiiiiiiggggggggggg
Epoch: 46 | Flops (T): 4.286525916384, Tokens (M) 2.475866, Train Loss 1.172, Val Loss 1.269 | Gen: eeeeeeeeeeeeeeeeeeee iwwwwwwwwwwwwwwwwwww ooooooiiiiiiiiggggggg yyyyyyyyyyyyyyyyyyyy ookkkgggggggggggggggg
Epoch: 47 | Flops (T): 4.377728595456, Tokens (M) 2.528544, Train Loss 1.157, Val Loss 1.237 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii oooiiiinnnnnnnnnnnn yyyyyyyyyyyyyyyyyyyy ooiiiiiiinnnnnnnnnng
Epoch: 48 | Flops (T): 4.468931274528, Tokens (M) 2.581222, Train Loss 1.150, Val Loss 1.260 | Gen: eeeeeeeeeeeeeeeeeeee iwwwwwwwwwwwwwwwwwww oooiiiiggggggggggggg yyyyyyyyyyyyyyyyyyyy ookkgggggggggggggggg
Epoch: 49 | Flops (T): 4.5601339536, Tokens (M) 2.6339, Train Loss 1.154, Val Loss 1.231 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii oooonnnnnnnnnnnnnnnn yyyyyyyyyyyyyyyyyyyy oooiiinnnnnnnnnnnnn
Epoch: 50 | Flops (T): 4.651336632672, Tokens (M) 2.686578, Train Loss 1.153, Val Loss 1.238 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiwwwwwwwwwwww oooooiiiiiiiiiggggggg yyyyyyyyyyyyyyyyyyyy ooiiiggggggggggggggg
Epoch: 51 | Flops (T): 4.742539311744, Tokens (M) 2.739256, Train Loss 1.149, Val Loss 1.257 | Gen: eeeeeeeeeeeeeeeeeeee iwwwwwwwwwwwwwwwwwww ooiiiinnggggggggggggg yyyyyyyyyyyyyyyyyyyy oiiggggggggggggggggg
Epoch: 52 | Flops (T): 4.833741990816, Tokens (M) 2.791934, Train Loss 1.138, Val Loss 1.203 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii oooooinnnnnnnnnnnnnn yyyyyyyyyyyyyyyyyyyy ookiinnnnnnnnnnnnnnn
Epoch: 53 | Flops (T): 4.924944669888, Tokens (M) 2.844612, Train Loss 1.123, Val Loss 1.232 | Gen: eeeeeeeeeeeeeeeeeeee wwwwwwwwwwwwwwwwwwww oooonnnnnnnnnggggggg yyyyyyyyyyyyyyyyyyyy ooiggggggggggggggggg
Epoch: 54 | Flops (T): 5.01614734896, Tokens (M) 2.89729, Train Loss 1.120, Val Loss 1.227 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii oooooinnnggggggggggg yyyyyyyyyyyyyyyyyyyy ookkgggggggggggggggg
Epoch: 55 | Flops (T): 5.107350028032, Tokens (M) 2.949968, Train Loss 1.124, Val Loss 1.245 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii oooiiiggggggggggggg yyyyyyyyyyyyyyyyyyyy okkgggggggggggggggg
Epoch: 56 | Flops (T): 5.198552707104, Tokens (M) 3.002646, Train Loss 1.120, Val Loss 1.214 | Gen: eeeeeeeeeeeeeeeeeeee wwwwwwwwwwwwwwwwwwww ooonnnnnnnnnnnnnnnn yyyyyyyyyyyyyyyyyyyy oonnnnnnnnnnnnnnnnn
Epoch: 57 | Flops (T): 5.289755386176, Tokens (M) 3.055324, Train Loss 1.124, Val Loss 1.268 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii ooooonnggggggggggggg yyyyyyyyyyyyyyyyyyyy okkkkgggggggggggggg
Epoch: 58 | Flops (T): 5.380958065248, Tokens (M) 3.108002, Train Loss 1.132, Val Loss 2.113 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii yyyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyyy ooknnnnnnnnnnnnnnnn
Epoch: 59 | Flops (T): 5.47216074432, Tokens (M) 3.16068, Train Loss 1.231, Val Loss 1.295 | Gen: eeeeeeeeeeeeeeeeeeee iwwwwwwwwwwwwwwwwwww ooonnnnnnnccccccccccc yyyyyyyyyyyyyyyyyyyy oknnnnnnnnggggggggg
Epoch: 60 | Flops (T): 5.563363423392, Tokens (M) 3.213358, Train Loss 1.168, Val Loss 1.247 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii ooonnnnccccccccccccc yyyyyyyyyyyyyyyyyyyy oooiiiiiiiiiiiiiiiii

Epoch: 61 | Flops (T): 5.654566102464, Tokens (M) 3.266036, Train Loss 1.146, Val Loss 1.244 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiwwwwwwwww ooonnnnnnnnnnnnggggg yyyyyyyyyyyyyyyyyyy ookiiiiggggggggggggg
Epoch: 62 | Flops (T): 5.745768781536, Tokens (M) 3.318714, Train Loss 1.168, Val Loss 1.735 | Gen: eeeeeeeeeeeeeeeeeeee iwwwwwwwwwwwwwwwwwww yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy kkkgggggggggggggggggg
Epoch: 63 | Flops (T): 5.836971460608, Tokens (M) 3.371392, Train Loss 1.209, Val Loss 1.271 | Gen: eeeeeeeeeeeeeewwwwww yyyyyyyyyyyyyyyyyyy ooooiiiiiiiiiiiiiiii yyyyyyyyyyyyyyyyyyy ooiiiiiiiiiiiiiiiiii
Epoch: 64 | Flops (T): 5.92817413968, Tokens (M) 3.42407, Train Loss 1.156, Val Loss 1.232 | Gen: eeeeeeeeeeeeeeeeeeee yyyyyyyyyyyyyyyyyyy ooiiiiiiiiiiiiiiiigg yyyyyyyyyyyyyyyyyyy oiiiggggggggggggggg
Epoch: 65 | Flops (T): 6.019376818752, Tokens (M) 3.476748, Train Loss 1.130, Val Loss 1.205 | Gen: eeeeeeeeeeeeeeeeeeee yyyyyyyyyyyyyyyyyyy ooooiiiiiiiiiiiiiiii yyyyyyyyyyyyyyyyyyy okiiiiggggggggggggg
Epoch: 66 | Flops (T): 6.110579497824, Tokens (M) 3.529426, Train Loss 1.114, Val Loss 1.188 | Gen: eeeeeeeeeeeeeeeeeeee yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy kkkiinggggggggggggg
Epoch: 67 | Flops (T): 6.201782176896, Tokens (M) 3.582104, Train Loss 1.102, Val Loss 1.203 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy ooiiiggggggggggggggg
Epoch: 68 | Flops (T): 6.292984855968, Tokens (M) 3.634782, Train Loss 1.097, Val Loss 1.186 | Gen: eeeeeeeeeeeeeeeeeeee yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy kkkinggggggggggggggg
Epoch: 69 | Flops (T): 6.38418753504, Tokens (M) 3.68746, Train Loss 1.101, Val Loss 1.268 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii oooiiiiiiiiiiiiiiiii yyyyyyyyyyyyyyyyyyy kkkkkkkkkkknnnnnnnnn
Epoch: 70 | Flops (T): 6.475390214112, Tokens (M) 3.740138, Train Loss 1.115, Val Loss 1.238 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy oinnnnnnnnnggggggggg
Epoch: 71 | Flops (T): 6.566592893184, Tokens (M) 3.792816, Train Loss 1.114, Val Loss 1.187 | Gen: eeeeeeeeeeeeeeeeeeee iwwwwwwwwwwwwwwwwwww yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy okkkggggggggggggggggg
Epoch: 72 | Flops (T): 6.657795572256, Tokens (M) 3.845494, Train Loss 1.104, Val Loss 1.168 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy ookkiinggggggggggggg
Epoch: 73 | Flops (T): 6.748998251328, Tokens (M) 3.898172, Train Loss 1.085, Val Loss 1.192 | Gen: eeeeeeeeeeeeeeeeeeee iwwwwwwwwwwwwwwwwwww yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy kkkknnggggggggggggggg
Epoch: 74 | Flops (T): 6.8402009304, Tokens (M) 3.95085, Train Loss 1.082, Val Loss 1.207 | Gen: wwwwwwwwwwwwwwwwwwww wwwwwwwwwwwwwwwwwwww oooiinnnnnggggggggg yyyyyyyyyyyyyyyyyyy kkkgggggggggggggggggg
Epoch: 75 | Flops (T): 6.931403609472, Tokens (M) 4.003528, Train Loss 1.095, Val Loss 1.178 | Gen: eeeeeeeeeeeeeeeeeeee iwwwwwwwwwwwwwwwwwww yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy okkkkggggggggggggggg
Epoch: 76 | Flops (T): 7.022606288544, Tokens (M) 4.056206, Train Loss 1.089, Val Loss 1.232 | Gen: eeeeeeeeeeeeeeeeeeee iiiiiiiiiiiiiiiiiiii yyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyy ookkiggggggggggggggg

Epoch: 77 | Flops (T): 7.113808967616, Tokens (M) 4.108884, Train Loss 1.070, Val Loss 1.220 | Gen: eeeeeeeeeeeeeeeeeeee iwwwwwwwwwwwwwwwwww yyyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyyy ooooinnnnnnnnnnnnnn
Epoch: 78 | Flops (T): 7.205011646688, Tokens (M) 4.161562, Train Loss 1.063, Val Loss 1.147 | Gen: eeeeeeeeeeeeeeeeeeew iiiiiiiiiiiiiiiiiiii yyyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyyy okkiggggggggggggggg
Epoch: 79 | Flops (T): 7.29621432576, Tokens (M) 4.21424, Train Loss 1.047, Val Loss 1.161 | Gen: wwwwwwwwwwwwwwwwwwww iwwwwwwwwwwwwwwwwwww yyyyyyyyyyyyyyyyyyyy yyyyyyyyyyyyyyyyyyyy okkggggggggggggggggg
Epoch: 80 | Flops (T): 7.387417004832, Tokens (M) 4.266918, Train Loss 1.042, Val Loss 1.170 | Gen: ewwwwwwwwwwwwwwwwwww iiiiiiiiiiiiiiiiiwww ooooiiiiiiiiiiiiiinc yyyyyyyyyyyyyyyyyyyy ookkiiggggggggggggg
Epoch: 81 | Flops (T): 7.478619683904, Tokens (M) 4.319596, Train Loss 1.074, Val Loss 2.344 | Gen: ewwwwwwwwwwwwwwwwwww iwwwwwwwwwwwwwwwwwww ooooooiiiinnnnnnnnnn iiiiiiiiiiiiiiiiiii okkkkwwggggggggggg
Epoch: 82 | Flops (T): 7.569822362976, Tokens (M) 4.372274, Train Loss 1.266, Val Loss 1.304 | Gen: eeeeeeeeeeeeeeeeeeee wwwwwwwwwwwwwwwwwwww oonnnnnnnnnnnnnnnnnn yyyyyyyyyyyyyyyyyyyy owggggggggggggggggg
Epoch: 83 | Flops (T): 7.661025042048, Tokens (M) 4.424952, Train Loss 1.164, Val Loss 1.210 | Gen: eeeeeeeeeeeeeeeeeeee iwwwwwwwwwwwwwwwwwww oonnnnnnnnggggggggg yyyyyyyyyyyyyyyyyyyy oonnnnnggggggggggggg
Epoch: 84 | Flops (T): 7.75222772112, Tokens (M) 4.47763, Train Loss 1.098, Val Loss 1.154 | Gen: eeewwwwwwwwwwwwwwwww iwwwwwwwwwwwwwwwwhh ooooiiiinnnnnnnnnnnn iiisssssssssssssssss ooiiggggggggggggggg
Epoch: 85 | Flops (T): 7.843430400192, Tokens (M) 4.530308, Train Loss 1.071, Val Loss 1.175 | Gen: eeeeeeeeeeeeeeeeeeee iiiwwwwwwwwwwwwwwwww ooooiiiiiiiiiiiiiiin iiiiisssssssssssssss ookkkiiggggggggggggg
Epoch: 86 | Flops (T): 7.934633079264, Tokens (M) 4.582986, Train Loss 1.055, Val Loss 1.156 | Gen: eeeeeeeeeeeeewwwwwww iiiiwwwwwwwwwwwwwwww oooonnnnnnggggggggg yyyyyyyyyyyyyyyyyyyy ookkgggggggggggggggg
Epoch: 87 | Flops (T): 8.025835758336, Tokens (M) 4.635664, Train Loss 1.046, Val Loss 1.176 | Gen: eeewwwwwwwwwwwwwwwww iwwwwwwwwwwwwwwwwwww ooiiiinnnnnnccccccc iiwwwwwwwwwwwwwwwww okkggggggggggggggggg
Epoch: 88 | Flops (T): 8.117038437408, Tokens (M) 4.688342, Train Loss 1.038, Val Loss 1.204 | Gen: eeeeeewwwwwwwwwwwww iiiwwwwwwwwwwwwwwwww ooooiiiiinnggggggggg iiiiiiiiiiiiiiiiwwww oookkgggggggggggggggg
Epoch: 89 | Flops (T): 8.20824111648, Tokens (M) 4.74102, Train Loss 1.035, Val Loss 1.116 | Gen: eeewwwwwwwwwwwwwwwww iiwwwwwwwwwwwwwwwwww oooonnnnnnnccccccccc yyyyyyyyyyyyyyyyyyyy ookknggggggggggggggg
Epoch: 90 | Flops (T): 8.299443795552, Tokens (M) 4.793698, Train Loss 1.025, Val Loss 1.163 | Gen: ewwwwwwwwwwwwwwwwwww iwwwwwwwwwwwwwwwwwww oooonnnnnnnggggggggg yyyyyyyyyyyyyyyyyyyy okkgggggggggggggwwww
Epoch: 91 | Flops (T): 8.390646474624, Tokens (M) 4.846376, Train Loss 1.042, Val Loss 1.132 | Gen: eewwwwwwwwwwwwwwwwww iwwwwwwwwwwwwwwwwwww ooooiiiiiiiiiiiiiicc iiiiswwwwwwwwwwwwww okkkknnnggggggggnnw
Epoch: 92 | Flops (T): 8.481849153696, Tokens (M) 4.899054, Train Loss 1.039, Val Loss 1.147 | Gen: eeeeewwwwwwwwwwwwww iiiiiwwwwwwwwwwwwww iiiiiiiiiiiiiiiiiiii iiiiiiiiiiiwwwwwwww okkkgggggggggggggggg

```
Epoch:  93 | Flops (T): 8.573051832768, Tokens (M) 4.951732, Train Loss 1.040,
Val Loss 1.145 | Gen: eeeewwwwwwwwwwwwwwww iwwwwwwwwwwwhhhhhhhh
ooonnnggggggggggggg yyyyyyyyyyyyyyyyyyy okgggggggggggwwwwwww
Epoch:  94 | Flops (T): 8.66425451184, Tokens (M) 5.00441, Train Loss 1.013, Val
Loss 1.145 | Gen: eeeeeewwwwwwwwwwwwww iiiiiiiiiwwwwwwwwwww ooooinnnnnnnggggggg
iiisssssswwwwwwwwwww ookkgggggggggggggggg
Epoch:  95 | Flops (T): 8.755457190912, Tokens (M) 5.057088, Train Loss 0.988,
Val Loss 1.073 | Gen: eeeewwwwwwwwwwwwwwww iiwwwwwwwwwwwwwwwhhh
oooiinnnnnnggggccccc iisswwwwwwwwwwwwwwww ookkgggggggwwwwwwwwww
Epoch:  96 | Flops (T): 8.846659869984, Tokens (M) 5.109766, Train Loss 0.992,
Val Loss 1.144 | Gen: eewwwwwwwwwwwwwwwwww iwwwwwwwwwwwwwwwwwww
ooooooiinnnggggggggg iiswwwwwwwwwwwwwwwww okkgggggggggggggggwww
Epoch:  97 | Flops (T): 8.937862549056, Tokens (M) 5.162444, Train Loss 1.009,
Val Loss 1.094 | Gen: eeeeeeeewwwwwwwwwwww iiiwwwwwwwwwwwwwwwww
oooiiiiinnnnnnggggggg iiiiswwwwwwwwwwwwwww ookkgggggggggggggwwww
Epoch:  98 | Flops (T): 9.029065228128, Tokens (M) 5.215122, Train Loss 0.967,
Val Loss 1.058 | Gen: eewwwwwwwwwwwwwwwwww iiwwwwwwwwwwwwwwwwww
oooiiinnnnnnnnggggggg iiiswwwwwwwwwwwwwwww ookkgggggggwwwwwwwww
Epoch:  99 | Flops (T): 9.1202679072, Tokens (M) 5.2678, Train Loss 0.963, Val
Loss 1.155 | Gen: ewwwwwwwwwwwwwwwwwww wwwwwwwwwwwwwhhhhhh oodonnnnnnnnnnnnnnnn
wwwwwwwwwwwwwwwwwww ookgggwwwwwwwwwwwwww
Obtained lowest validation loss of: 1.0575359900330388
Avg tokens per example (Seq Len): 20.59343236903831, Avg flops per example:
35653901.12275215
```

[ ]:
```python
TEST_SENTENCE = "the air conditioning is working"
translated = translate_sentence(
    TEST_SENTENCE,  trans_decoder_s, None, trans_args_s
)
print("source:\t\t{} \ntranslated:\t{}".format(TEST_SENTENCE, translated))
```

```
source:         the air conditioning is working
translated:     ewwwwwwwwwwwwwwwwwww wwwwwwwwwwwwwhhhhhh oodonnnnnnnnnnnnnnnn
wwwwwwwwwwwwwwwwwww ookgggwwwwwwwwwwwwww
```

# 6 Part 3: Compute Optimal Model

Now we will train a set of decoder only transformer models to investigate the compute optimal model.

## 6.1 Step 1: Model training

- Let's train six models of different model sizes.

[ ]:
```python
TEST_SENTENCE = "floor"
trans16_args_s = AttrDict()
args_dict_16 = {
    "data_file_name": "pig_latin_small", #pig_latin_small debug
```

```
    "cuda": use_cuda,
    "nepochs": 100,
    "checkpoint_dir": "checkpoints",
    "learning_rate": 5e-4,
    "early_stopping_patience": 100,
    "lr_decay": 0.99,
    "batch_size": 64,
    "hidden_size": 16,
    "num_transformer_layers": 4,
    "loss_seq_span": "target_only",
    "num_layers": 4,
    "vocab_size": 30
}
trans16_args_s.update(args_dict_16)
print_opts(trans16_args_s)
trans16_decoder_s, trans16_losses_s, idx_dict = train(trans16_args_s)
```

```
================================================================================
                                       Opts
--------------------------------------------------------------------------------
                         data_file_name: pig_latin_small
                                   cuda: 1
                               nepochs: 100
                         checkpoint_dir: checkpoints
                          learning_rate: 0.0005
                early_stopping_patience: 100
                                lr_decay: 0.99
                             batch_size: 64
                            hidden_size: 16
                 num_transformer_layers: 4
                          loss_seq_span: target_only
                             num_layers: 4
                             vocab_size: 30
================================================================================
================================================================================
                                   Data Stats
--------------------------------------------------------------------------------
('irresistible', 'irresistibleway')
('repentance', 'epentanceray')
('pretty', 'ettypray')
('parcels', 'arcelspay')
('survey', 'urveysay')
Num unique word pairs: 3198
Vocabulary: dict_keys(['-', 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j',
'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z',
'SOS', 'EOS', 'EOP'])
Vocab size: 30
```

```
================================================================================
defaultdict(<class 'list'>, {(12, 15): [('irresistible', 'irresistibleway'),
('advantageous', 'advantageousway'), ('impartiality', 'impartialityway'),
('interruption', 'interruptionway'), ('anticipation', 'anticipationway'),
('impenetrable', 'impenetrableway'), ('excellencies', 'excellenciesway'),
('intermission', 'intermissionway'), ('interspersed', 'interspersedway'),
('unaffectedly', 'unaffectedlyway'), ('insinuations', 'insinuationsway'),
('unpardonable', 'unpardonableway'), ('arrangements', 'arrangementsway'),
('unexpectedly', 'unexpectedlyway'), ('introduction', 'introductionway'),
('occasionally', 'occasionallyway'), ('establishing', 'establishingway'),
('unanswerable', 'unanswerableway'), ('independence', 'independenceway'),
('inclinations', 'inclinationsway'), ('instructions', 'instructionsway'),
('undoubtingly', 'undoubtinglyway'), ('unfathomable', 'unfathomableway'),
('impoverished', 'impoverishedway'), ('outstretched', 'outstretchedway'),
('unrestrained', 'unrestrainedway'), ('ungraciously', 'ungraciouslyway'),
('imperfection', 'imperfectionway'), ('unreasonably', 'unreasonablyway'),
('interference', 'interferenceway'), ('inconvenient', 'inconvenientway'),
('enthusiastic', 'enthusiasticway'), ('extinguished', 'extinguishedway'),
('entanglement', 'entanglementway'), ('affectionate', 'affectionateway')], (10,
12): [('repentance', 'epentanceray'), ('capability', 'apabilitycay'),
('beneficial', 'eneficialbay'), ('retrenched', 'etrenchedray'), ('devonshire',
'evonshireday'), ('palanquins', 'alanquinspay'), ('contribute', 'ontributecay'),
('minuteness', 'inutenessmay'), ('dissimilar', 'issimilarday'), ('flattering',
'atteringflay'), ('confounded', 'onfoundedcay'), ('compromise', 'ompromisecay'),
('contrasted', 'ontrastedcay'), ('precaution', 'ecautionpray'), ('fastidious',
'astidiousfay'), ('blackguard', 'ackguardblay'), ('heroically', 'eroicallyhay'),
('lengthened', 'engthenedlay'), ('pocketbook', 'ocketbookpay'), ('companions',
'ompanionscay'), ('displeased', 'ispleasedday'), ('pianoforte', 'ianofortepay'),
('succession', 'uccessionsay'), ('composedly', 'omposedlycay'), ('pronounced',
'onouncedpray'), ('honourable', 'onourablehay'), ('solicitude', 'olicitudesay'),
('justifying', 'ustifyingjay'), ('collecting', 'ollectingcay'), ('headstrong',
'eadstronghay'), ('forbidding', 'orbiddingfay'), ('successful', 'uccessfulsay'),
('confidence', 'onfidencecay'), ('prescribed', 'escribedpray'), ('scrutinies',
'utiniesscray'), ('melancholy', 'elancholymay'), ('handsomest', 'andsomesthay'),
('prescience', 'esciencepray'), ('conclusion', 'onclusioncay'), ('remarkably',
'emarkablyray'), ('gratefully', 'atefullygray'), ('michaelmas', 'ichaelmasmay'),
('containing', 'ontainingcay'), ('discussion', 'iscussionday'), ('resentment',
'esentmentray'), ('restraints', 'estraintsray'), ('tablecloth', 'ableclothtay'),
('projecting', 'ojectingpray'), ('frightened', 'ightenedfray'), ('pardonable',
'ardonablepay'), ('variations', 'ariationsvay'), ('sensations', 'ensationssay'),
('rheumatism', 'eumatismrhay'), ('dispirited', 'ispiritedday'), ('politeness',
'olitenesspay'), ('suggesting', 'uggestingsay'), ('willoughby', 'illoughbyway'),
('reasonable', 'easonableray'), ('discerning', 'iscerningday'), ('previously',
'eviouslypray'), ('passionate', 'assionatepay'), ('prophecies', 'opheciespray'),
('possessing', 'ossessingpay'), ('perfection', 'erfectionpay'), ('heightened',
'eightenedhay'), ('generously', 'enerouslygay'), ('faithfully', 'aithfullyfay'),
('reconciled', 'econciledray'), ('distressed', 'istressedday'), ('respective',
'espectiveray'), ('liberality', 'iberalitylay'), ('disengaged', 'isengagedday'),
```

('honourably', 'onourablyhay'), ('persuading', 'ersuadingpay'), ('cultivated',
'ultivatedcay'), ('projection', 'ojectionpray'), ('disapprove', 'isapproveday'),
('conciliate', 'onciliatecay'), ('satisfying', 'atisfyingsay'), ('dispatches',
'ispatchesday'), ('repeatedly', 'epeatedlyray'), ('suspicions', 'uspicionssay'),
('perplexity', 'erplexitypay'), ('strengthen', 'engthenstray'), ('dependence',
'ependenceday'), ('dissipated', 'issipatedday'), ('conformity', 'onformitycay'),
('hysterical', 'ystericalhay'), ('throughout', 'oughoutthray'), ('sacrificed',
'acrificedsay'), ('dimensions', 'imensionsday'), ('flatteries', 'atteriesflay'),
('completely', 'ompletelycay'), ('reproaches', 'eproachesray'), ('contriving',
'ontrivingcay'), ('consequent', 'onsequentcay'), ('liveliness', 'ivelinesslay'),
('resistance', 'esistanceray'), ('caricature', 'aricaturecay'), ('stimulated',
'imulatedstay'), ('brightened', 'ightenedbray'), ('prevailing', 'evailingpray'),
('disclosure', 'isclosureday'), ('bequeathed', 'equeathedbay'), ('disappoint',
'isappointday'), ('complexion', 'omplexioncay'), ('dismounted', 'ismountedday'),
('remembered', 'ememberedray'), ('delineated', 'elineatedday'), ('subsequent',
'ubsequentsay'), ('persevered', 'erseveredpay'), ('doubtingly', 'oubtinglyday'),
('characters', 'aracterschay'), ('neighbours', 'eighboursnay'), ('witticisms',
'itticismsway'), ('materially', 'ateriallymay'), ('playthings', 'aythingsplay'),
('suspecting', 'uspectingsay'), ('conditions', 'onditionscay'), ('discovered',
'iscoveredday'), ('deficiency', 'eficiencyday'), ('regularity', 'egularityray'),
('subjection', 'ubjectionsay'), ('brightness', 'ightnessbray'), ('pleasantly',
'easantlyplay'), ('delicacies', 'elicaciesday'), ('preference', 'eferencepray'),
('difference', 'ifferenceday'), ('compassion', 'ompassioncay'), ('competence',
'ompetencecay'), ('generosity', 'enerositygay'), ('retirement', 'etirementray'),
('strikingly', 'ikinglystray'), ('subsisting', 'ubsistingsay'), ('conscience',
'onsciencecay'), ('diabolical', 'iabolicalday'), ('temptation', 'emptationtay'),
('formidable', 'ormidablefay'), ('reconsider', 'econsiderray'), ('prejudiced',
'ejudicedpray'), ('comprehend', 'omprehendcay'), ('compliment', 'omplimentcay'),
('sufferings', 'ufferingssay'), ('concluding', 'oncludingcay'), ('compliance',
'ompliancecay'), ('mosquitoes', 'osquitoesmay'), ('thoughtful', 'oughtfulthay'),
('bitterness', 'itternessbay'), ('mentioning', 'entioningmay'), ('toleration',
'olerationtay'), ('profession', 'ofessionpray'), ('discompose', 'iscomposeday'),
('waistcoats', 'aistcoatsway'), ('reflecting', 'eflectingray'), ('positively',
'ositivelypay'), ('diminished', 'iminishedday')], (6, 8): [('pretty',
'ettypray'), ('survey', 'urveysay'), ('demure', 'emureday'), ('secure',
'ecuresay'), ('sobbed', 'obbedsay'), ('moment', 'omentmay'), ('guilty',
'uiltygay'), ('combat', 'ombatcay'), ('modern', 'odernmay'), ('father',
'atherfay'), ('horses', 'orseshay'), ('closer', 'oserclay'), ('nicety',
'icetynay'), ('shaken', 'akenshay'), ('centre', 'entrecay'), ('powers',
'owerspay'), ('simple', 'implesay'), ('beings', 'eingsbay'), ('recent',
'ecentray'), ('sketch', 'etchskay'), ('loving', 'ovinglay'), ('dagger',
'aggerday'), ('hearts', 'eartshay'), ('copied', 'opiedcay'), ('gazing',
'azinggay'), ('sooner', 'oonersay'), ('disarm', 'isarmday'), ('suited',
'uitedsay'), ('doubts', 'oubtsday'), ('lifted', 'iftedlay'), ('vacant',
'acantvay'), ('palmer', 'almerpay'), ('grudge', 'udgegray'), ('folded',
'oldedfay'), ('remove', 'emoveray'), ('borrow', 'orrowbay'), ('scheme',
'emeschay'), ('solemn', 'olemnsay'), ('denote', 'enoteday'), ('bustle',
'ustlebay'), ('turned', 'urnedtay'), ('mutton', 'uttonmay'), ('tender',

'endertay'), ('hearer', 'earerhay'), ('begins', 'eginsbay'), ('taught',
'aughttay'), ('honour', 'onourhay'), ('robert', 'obertray'), ('gibson',
'ibsongay'), ('virtue', 'irtuevay'), ('vanish', 'anishvay'), ('writes',
'iteswray'), ('silent', 'ilentsay'), ('singly', 'inglysay'), ('jargon',
'argonjay'), ('grieve', 'ievegray'), ('design', 'esignday'), ('linger',
'ingerlay'), ('bathed', 'athedbay'), ('pardon', 'ardonpay'), ('rumour',
'umourray'), ('dances', 'ancesday'), ('remark', 'emarkray'), ('worded',
'ordedway'), ('honest', 'onesthay'), ('chosen', 'osenchay'), ('boyish',
'oyishbay'), ('stress', 'essstray'), ('caught', 'aughtcay'), ('potent',
'otentpay'), ('during', 'uringday'), ('stocks', 'ocksstay'), ('relics',
'elicsray'), ('tongue', 'onguetay'), ('street', 'eetstray'), ('whiled',
'iledwhay'), ('manner', 'annermay'), ('truths', 'uthstray'), ('travel',
'aveltray'), ('couple', 'ouplecay'), ('gaiety', 'aietygay'), ('giving',
'ivinggay'), ('county', 'ountycay'), ('cramps', 'ampscray'), ('wealth',
'ealthway'), ('divine', 'ivineday'), ('behold', 'eholdbay'), ('feared',
'earedfay'), ('sparks', 'arksspay'), ('finger', 'ingerfay'), ('talent',
'alenttay'), ('demand', 'emandday'), ('coffee', 'offeecay'), ('merest',
'erestmay'), ('detail', 'etailday'), ('casino', 'asinocay'), ('choose',
'oosechay'), ('warmed', 'armedway'), ('relate', 'elateray'), ('sashes',
'ashessay'), ('chance', 'ancechay'), ('rushed', 'ushedray'), ('health',
'ealthhay'), ('preyed', 'eyedpray'), ('mutual', 'utualmay'), ('months',
'onthsmay'), ('raises', 'aisesray'), ('things', 'ingsthay'), ('master',
'astermay'), ('thanks', 'anksthay'), ('cheeks', 'eekschay'), ('hazard',
'azardhay'), ('slight', 'ightslay'), ('guests', 'uestsgay'), ('halves',
'alveshay'), ('lessen', 'essenlay'), ('notice', 'oticenay'), ('higher',
'igherhay'), ('truest', 'uesttray'), ('porter', 'orterpay'), ('fickle',
'icklefay'), ('behave', 'ehavebay'), ('credit', 'editcray'), ('number',
'umbernay'), ('mother', 'othermay'), ('tricks', 'ickstray'), ('traced',
'acedtray'), ('subdue', 'ubduesay'), ('ruined', 'uinedray'), ('famous',
'amousfay'), ('danced', 'ancedday'), ('clever', 'everclay'), ('drives',
'ivesdray'), ('resist', 'esistray'), ('forget', 'orgetfay'), ('stable',
'ablestay'), ('bigger', 'iggerbay'), ('betray', 'etraybay'), ('senses',
'ensessay'), ('smiled', 'iledsmay'), ('hating', 'atinghay'), ('bolder',
'olderbay'), ('banish', 'anishbay'), ('remain', 'emainray'), ('period',
'eriodpay'), ('gossip', 'ossipgay'), ('basket', 'asketbay'), ('genius',
'eniusgay'), ('seized', 'eizedsay'), ('widely', 'idelyway'), ('pursue',
'ursuepay'), ('favour', 'avourfay'), ('sighed', 'ighedsay'), ('plants',
'antsplay'), ('proved', 'ovedpray'), ('season', 'easonsay'), ('circle',
'irclecay'), ('taking', 'akingtay'), ('screen', 'eenscray'), ('talker',
'alkertay'), ('teazed', 'eazedtay'), ('suffer', 'uffersay'), ('curate',
'uratecay'), ('traits', 'aitstray'), ('defect', 'efectday'), ('called',
'alledcay'), ('dining', 'iningday'), ('direct', 'irectday'), ('dinner',
'innerday'), ('joking', 'okingjay'), ('whiter', 'iterwhay'), ('limits',
'imitslay'), ('papers', 'aperspay'), ('public', 'ublicpay'), ('losing',
'osinglay'), ('handed', 'andedhay'), ('wisest', 'isestway'), ('delays',
'elaysday'), ('leaves', 'eaveslay'), ('points', 'ointspay'), ('hither',
'itherhay'), ('horrid', 'orridhay'), ('middle', 'iddlemay'), ('vastly',
'astlyvay'), ('luxury', 'uxurylay'), ('failed', 'ailedfay'), ('lodges',

```
'odgeslay'), ('female', 'emalefay'), ('walked', 'alkedway'), ('nipped',
'ippednay'), ('nephew', 'ephewnay'), ('settle', 'ettlesay'), ('kinder',
'inderkay'), ('bowing', 'owingbay'), ('smooth', 'oothsmay'), ('roared',
'oaredray'), ('wiping', 'ipingway'), ('jewels', 'ewelsjay'), ('breath',
'eathbray'), ('steady', 'eadystay'), ('breach', 'eachbray'), ('polite',
'olitepay'), ('colder', 'oldercay'), ('though', 'oughthay'), ('poured',
'ouredpay'), ('paused', 'ausedpay'), ('nought', 'oughtnay'), ('shrewd',
'ewdshray'), ('fourth', 'ourthfay'), ('parish', 'arishpay'), ('traded',
'adedtray'), ('glance', 'anceglay'), ('lesser', 'esserlay'), ('curacy',
'uracycay'), ('labour', 'abourlay'), ('became', 'ecamebay'), ('decide',
'ecideday'), ('kisses', 'isseskay'), ('styled', 'yledstay'), ('sprung',
'ungspray'), ('priory', 'iorypray'), ('derive', 'eriveday'), ('school',
'oolschay'), ('liking', 'ikinglay'), ('dawned', 'awnedday'), ('weaken',
'eakenway'), ('winter', 'interway'), ('rivals', 'ivalsray'), ('sallow',
'allowsay'), ('defend', 'efendday'), ('seated', 'eatedsay'), ('penury',
'enurypay'), ('repair', 'epairray'), ('chaise', 'aisechay'), ('church',
'urchchay'), ('beyond', 'eyondbay'), ('wisdom', 'isdomway'), ('retain',
'etainray'), ('toilet', 'oilettay'), ('talked', 'alkedtay'), ('dearly',
'earlyday'), ('reward', 'ewardray'), ('safely', 'afelysay'), ('making',
'akingmay'), ('debate', 'ebateday'), ('future', 'uturefay'), ('broken',
'okenbray'), ('begged', 'eggedbay'), ('thirty', 'irtythay'), ('muslin',
'uslinmay'), ('marked', 'arkedmay'), ('hinder', 'inderhay'), ('beheld',
'eheldbay'), ('maxims', 'aximsmay'), ('theirs', 'eirsthay'), ('corner',
'ornercay'), ('piqued', 'iquedpay'), ('rested', 'estedray'), ('needed',
'eedednay'), ('fitted', 'ittedfay'), ('candid', 'andidcay'), ('laughs',
'aughslay'), ('window', 'indowway'), ('tastes', 'astestay'), ('spread',
'eadspray'), ('hollow', 'ollowhay'), ('sorrow', 'orrowsay'), ('partly',
'artlypay'), ('served', 'ervedsay'), ('flowed', 'owedflay'), ('remedy',
'emedyray'), ('trying', 'yingtray'), ('latest', 'atestlay'), ('manage',
'anagemay'), ('gained', 'ainedgay'), ('plenty', 'entyplay'), ('garden',
'ardengay'), ('frosts', 'ostsfray'), ('thirds', 'irdsthay'), ('handle',
'andlehay'), ('verbal', 'erbalvay'), ('sheath', 'eathshay'), ('covert',
'overtcay'), ('darker', 'arkerday'), ('harris', 'arrishay'), ('houses',
'ouseshay'), ('nature', 'aturenay'), ('thomas', 'omasthay'), ('submit',
'ubmitsay'), ('danger', 'angerday'), ('lawyer', 'awyerlay'), ('matter',
'attermay'), ('rugged', 'uggedray'), ('highly', 'ighlyhay'), ('whence',
'encewhay'), ('martha', 'arthamay'), ('hearty', 'eartyhay'), ('recall',
'ecallray'), ('raised', 'aisedray'), ('follow', 'ollowfay'), ('nearer',
'earernay'), ('gentle', 'entlegay'), ('pitied', 'itiedpay'), ('nurses',
'ursesnay'), ('punish', 'unishpay')], (7, 9): [('parcels', 'arcelspay'),
('welcome', 'elcomeway'), ('posture', 'osturepay'), ('mothers', 'othersmay'),
('tempers', 'emperstay'), ('carries', 'arriescay'), ('neglect', 'eglectnay'),
('prevail', 'evailpray'), ('declare', 'eclareday'), ('crowned', 'ownedcray'),
('mindful', 'indfulmay'), ('blessed', 'essedblay'), ('kitchen', 'itchenkay'),
('younger', 'youngeray'), ('hanover', 'anoverhay'), ('lowered', 'oweredlay'),
('summons', 'ummonssay'), ('suffice', 'ufficesay'), ('rousing', 'ousingray'),
('feeling', 'eelingfay'), ('parlors', 'arlorspay'), ('gravely', 'avelygray'),
('connect', 'onnectcay'), ('species', 'eciesspay'), ('revived', 'evivedray'),
```

```
('tribute', 'ibutetray'), ('pitying', 'ityingpay'), ('brother', 'otherbray'),
('retired', 'etiredray'), ('suffers', 'ufferssay'), ('spurned', 'urnedspay'),
('dropped', 'oppeddray'), ('galling', 'allinggay'), ('seducer', 'educersay'),
('foibles', 'oiblesfay'), ('tuesday', 'uesdaytay'), ('patient', 'atientpay'),
('worsted', 'orstedway'), ('trifled', 'ifledtray'), ('waiting', 'aitingway'),
('widower', 'idowerway'), ('hardily', 'ardilyhay'), ('glimpse', 'impseglay'),
('whether', 'etherwhay'), ('reverie', 'everieray'), ('resolve', 'esolveray'),
('propose', 'oposepray'), ('tallest', 'allesttay'), ('nominal', 'ominalnay'),
('languid', 'anguidlay'), ('wronged', 'ongedwray'), ('nothing', 'othingnay'),
('promote', 'omotepray'), ('novelty', 'oveltynay'), ('dooming', 'oomingday'),
('display', 'isplayday'), ('correct', 'orrectcay'), ('feigned', 'eignedfay'),
('schemes', 'emesschay'), ('deepest', 'eepestday'), ('flannel', 'annelflay'),
('hauteur', 'auteurhay'), ('sharing', 'aringshay'), ('getting', 'ettinggay'),
('screams', 'eamsscray'), ('variety', 'arietyvay'), ('yielded', 'yieldeday'),
('brought', 'oughtbray'), ('haughty', 'aughtyhay'), ('bristol', 'istolbray'),
('silence', 'ilencesay'), ('staying', 'ayingstay'), ('confess', 'onfesscay'),
('placing', 'acingplay'), ('noisier', 'oisiernay'), ('nursing', 'ursingnay'),
('tuition', 'uitiontay'), ('welfare', 'elfareway'), ('hunters', 'untershay'),
('poplars', 'oplarspay'), ('deprive', 'epriveday'), ('mansion', 'ansionmay'),
('reasons', 'easonsray'), ('conceit', 'onceitcay'), ('burgess', 'urgessbay'),
('receipt', 'eceiptray'), ('befalls', 'efallsbay'), ('fetches', 'etchesfay'),
('ranging', 'angingray'), ('passion', 'assionpay'), ('smirked', 'irkedsmay'),
('honours', 'onourshay'), ('covered', 'overedcay'), ('dancing', 'ancingday'),
('hundred', 'undredhay'), ('forgave', 'orgavefay'), ('fifteen', 'ifteenfay'),
('lasting', 'astinglay'), ('minuter', 'inutermay'), ('tremour', 'emourtray'),
('passing', 'assingpay'), ('streets', 'eetsstray'), ('resists', 'esistsray'),
('cleared', 'earedclay'), ('miracle', 'iraclemay'), ('several', 'everalsay'),
('sighing', 'ighingsay'), ('resumed', 'esumedray'), ('merited', 'eritedmay'),
('without', 'ithoutway'), ('removed', 'emovedray'), ('behaved', 'ehavedbay'),
('succour', 'uccoursay'), ('bribing', 'ibingbray'), ('charity', 'aritychay'),
('shortly', 'ortlyshay'), ('tempted', 'emptedtay'), ('briefly', 'ieflybray'),
('parties', 'artiespay'), ('touched', 'ouchedtay'), ('content', 'ontentcay'),
('proceed', 'oceedpray'), ('windows', 'indowsway'), ('ribbons', 'ibbonsray'),
('stepped', 'eppedstay'), ('pursuit', 'ursuitpay'), ('quarter', 'arterquay'),
('prudent', 'udentpray'), ('dignity', 'ignityday'), ('payment', 'aymentpay'),
('grounds', 'oundsgray'), ('bequest', 'equestbay'), ('donavan', 'onavanday'),
('renewed', 'enewedray'), ('smaller', 'allersmay'), ('fearful', 'earfulfay'),
('civilly', 'ivillycay'), ('romance', 'omanceray'), ('females', 'emalesfay'),
('grieved', 'ievedgray'), ('created', 'eatedcray'), ('command', 'ommandcay'),
('defined', 'efinedday'), ('bluntly', 'untlyblay'), ('recital', 'ecitalray'),
('sorrows', 'orrowssay'), ('whither', 'itherwhay'), ('spotted', 'ottedspay'),
('justify', 'ustifyjay'), ('conduit', 'onduitcay'), ('revenge', 'evengeray'),
('scenery', 'eneryscay'), ('started', 'artedstay'), ('degrees', 'egreesday'),
('whisper', 'isperwhay'), ('strange', 'angestray'), ('bidding', 'iddingbay'),
('musical', 'usicalmay'), ('happens', 'appenshay'), ('compare', 'omparecay'),
('presume', 'esumepray'), ('sitting', 'ittingsay'), ('regards', 'egardsray'),
('violins', 'iolinsvay'), ('rightly', 'ightlyray'), ('memento', 'ementomay'),
('courted', 'ourtedcay'), ('servant', 'ervantsay'), ('possess', 'ossesspay'),
```

```
('herself', 'erselfhay'), ('present', 'esentpray'), ('natural', 'aturalnay'),
('managed', 'anagedmay'), ('hostess', 'ostesshay'), ('leaving', 'eavinglay'),
('mention', 'entionmay'), ('circuit', 'ircuitcay'), ('counsel', 'ounselcay'),
('guineas', 'uineasgay'), ('solidly', 'olidlysay'), ('dashing', 'ashingday'),
('station', 'ationstay'), ('written', 'ittenwray'), ('writing', 'itingwray'),
('country', 'ountrycay'), ('delight', 'elightday'), ('westons', 'estonsway'),
('debated', 'ebatedday'), ('lengths', 'engthslay'), ('torment', 'ormenttay'),
('reverse', 'everseray'), ('bedroom', 'edroombay'), ('forbade', 'orbadefay'),
('comings', 'omingscay'), ('tonight', 'onighttay'), ('similar', 'imilarsay'),
('calling', 'allingcay'), ('cassino', 'assinocay'), ('strains', 'ainsstray'),
('scratch', 'atchscray'), ('gardens', 'ardensgay'), ('freedom', 'eedomfray'),
('compact', 'ompactcay'), ('message', 'essagemay'), ('heavily', 'eavilyhay'),
('manners', 'annersmay'), ('systems', 'ystemssay'), ('forbore', 'orborefay'),
('wrapped', 'appedwray'), ('defects', 'efectsday'), ('villain', 'illainvay'),
('studied', 'udiedstay'), ('playing', 'ayingplay'), ('diction', 'ictionday'),
('tending', 'endingtay'), ('seeking', 'eekingsay'), ('rapture', 'aptureray'),
('dawdled', 'awdledday'), ('wearing', 'earingway'), ('comfort', 'omfortcay'),
('finally', 'inallyfay'), ('circles', 'irclescay'), ('putting', 'uttingpay'),
('twisted', 'istedtway'), ('destroy', 'estroyday'), ('whereas', 'ereaswhay'),
('dismiss', 'ismissday'), ('visibly', 'isiblyvay'), ('talents', 'alentstay'),
('finding', 'indingfay'), ('removes', 'emovesray'), ('replied', 'epliedray'),
('bluster', 'usterblay'), ('raising', 'aisingray'), ('plainly', 'ainlyplay'),
('worried', 'orriedway'), ('meaning', 'eaningmay'), ('blushes', 'ushesblay'),
('relapse', 'elapseray'), ('careful', 'arefulcay'), ('beloved', 'elovedbay'),
('degrade', 'egradeday'), ('foresaw', 'oresawfay'), ('procure', 'ocurepray'),
('kindest', 'indestkay'), ('secures', 'ecuressay'), ('hanging', 'anginghay'),
('persons', 'ersonspay'), ('taverns', 'avernstay'), ('walking', 'alkingway'),
('quicker', 'ickerquay'), ('guessed', 'uessedgay'), ('confirm', 'onfirmcay'),
('venture', 'enturevay'), ('project', 'ojectpray'), ('prayers', 'ayerspray'),
('heights', 'eightshay'), ('subject', 'ubjectsay'), ('summits', 'ummitssay'),
('towards', 'owardstay'), ('closing', 'osingclay'), ('journey', 'ourneyjay'),
('pointer', 'ointerpay'), ('chained', 'ainedchay'), ('clothes', 'othesclay'),
('require', 'equireray'), ('chusing', 'usingchay'), ('reproof', 'eproofray'),
('reckons', 'eckonsray'), ('pressed', 'essedpray'), ('talking', 'alkingtay'),
('harbour', 'arbourhay'), ('garrets', 'arretsgay'), ('foresee', 'oreseefay'),
('lustily', 'ustilylay'), ('leagued', 'eaguedlay'), ('judging', 'udgingjay'),
('rejoice', 'ejoiceray'), ('honiton', 'onitonhay'), ('shorten', 'ortenshay'),
('stating', 'atingstay'), ('privacy', 'ivacypray'), ('steeles', 'eelesstay'),
('thickly', 'icklythay'), ('shocked', 'ockedshay'), ('limited', 'imitedlay'),
('blights', 'ightsblay'), ('seclude', 'ecludesay'), ('cordial', 'ordialcay'),
('request', 'equestray'), ('denoted', 'enotedday'), ('darling', 'arlingday'),
('coxcomb', 'oxcombcay'), ('however', 'oweverhay'), ('perform', 'erformpay'),
('society', 'ocietysay'), ('deserts', 'esertsday'), ('palmers', 'almerspay'),
('manager', 'anagermay'), ('rupture', 'uptureray'), ('concise', 'oncisecay'),
('scolded', 'oldedscay'), ('seduced', 'educedsay'), ('doubted', 'oubtedday'),
('forward', 'orwardfay'), ('leisure', 'eisurelay'), ('robbing', 'obbingray'),
('horizon', 'orizonhay'), ('blunder', 'underblay'), ('flushed', 'ushedflay'),
('divided', 'ividedday'), ('charged', 'argedchay'), ('disturb', 'isturbday'),
```

('smitten', 'ittensmay'), ('seconds', 'econdssay'), ('letters', 'etterslay'),
('provide', 'ovidepray'), ('conduct', 'onductcay'), ('plagued', 'aguedplay'),
('stretch', 'etchstray'), ('hoarded', 'oardedhay'), ('blushed', 'ushedblay'),
('nodding', 'oddingnay'), ('mankind', 'ankindmay'), ('caprice', 'apricecay'),
('quitted', 'ittedquay')], (5, 7): [('china', 'inachay'), ('deter', 'eterday'),
('woman', 'omanway'), ('clock', 'ockclay'), ('borne', 'ornebay'), ('noise',
'oisenay'), ('midst', 'idstmay'), ('dwell', 'elldway'), ('smith', 'ithsmay'),
('gowns', 'ownsgay'), ('named', 'amednay'), ('glass', 'assglay'), ('price',
'icepray'), ('close', 'oseclay'), ('guide', 'uidegay'), ('newly', 'ewlynay'),
('sweep', 'eepsway'), ('swept', 'eptsway'), ('those', 'osethay'), ('lower',
'owerlay'), ('plump', 'umpplay'), ('downs', 'ownsday'), ('purse', 'ursepay'),
('linen', 'inenlay'), ('later', 'aterlay'), ('broad', 'oadbray'), ('miles',
'ilesmay'), ('facts', 'actsfay'), ('sleep', 'eepslay'), ('steps', 'epsstay'),
('wants', 'antsway'), ('forth', 'orthfay'), ('drove', 'ovedray'), ('mouth',
'outhmay'), ('worry', 'orryway'), ('dried', 'ieddray'), ('stood', 'oodstay'),
('brush', 'ushbray'), ('sting', 'ingstay'), ('mirth', 'irthmay'), ('heads',
'eadshay'), ('sight', 'ightsay'), ('whose', 'osewhay'), ('drawn', 'awndray'),
('score', 'orescay'), ('party', 'artypay'), ('tears', 'earstay'), ('heart',
'earthay'), ('words', 'ordsway'), ('hunts', 'untshay'), ('trace', 'acetray'),
('trial', 'ialtray'), ('swoon', 'oonsway'), ('knees', 'eesknay'), ('bring',
'ingbray'), ('solid', 'olidsay'), ('sharp', 'arpshay'), ('basis', 'asisbay'),
('cavil', 'avilcay'), ('rated', 'atedray'), ('grave', 'avegray'), ('serve',
'ervesay'), ('betty', 'ettybay'), ('judge', 'udgejay'), ('boast', 'oastbay'),
('march', 'archmay'), ('built', 'uiltbay'), ('clear', 'earclay'), ('happy',
'appyhay'), ('fruit', 'uitfray'), ('witty', 'ittyway'), ('total', 'otaltay'),
('dared', 'aredday'), ('defer', 'eferday'), ('tried', 'iedtray'), ('vexed',
'exedvay'), ('walls', 'allsway'), ('liked', 'ikedlay'), ('prove', 'ovepray'),
('sneer', 'eersnay'), ('merit', 'eritmay'), ('scott', 'ottscay'), ('madam',
'adammay'), ('truth', 'uthtray'), ('minor', 'inormay'), ('stare', 'arestay'),
('calls', 'allscay'), ('lucky', 'uckylay'), ('hence', 'encehay'), ('press',
'esspray'), ('comes', 'omescay'), ('dined', 'inedday'), ('think', 'inkthay'),
('plate', 'ateplay'), ('grief', 'iefgray'), ('names', 'amesnay'), ('stock',
'ockstay'), ('short', 'ortshay'), ('moved', 'ovedmay'), ('teach', 'eachtay'),
('fears', 'earsfay'), ('tiled', 'iledtay'), ('cried', 'iedcray'), ('mound',
'oundmay'), ('small', 'allsmay'), ('twice', 'icetway'), ('lines', 'ineslay'),
('looks', 'ookslay'), ('rival', 'ivalray'), ('first', 'irstfay'), ('wound',
'oundway'), ('saves', 'avessay'), ('nurse', 'ursenay'), ('treat', 'eattray'),
('could', 'ouldcay'), ('rider', 'iderray'), ('lobby', 'obbylay'), ('table',
'abletay'), ('touch', 'ouchtay'), ('topic', 'opictay'), ('stand', 'andstay'),
('leant', 'eantlay'), ('grass', 'assgray'), ('check', 'eckchay'), ('quite',
'itequay'), ('wives', 'ivesway'), ('walks', 'alksway'), ('labor', 'aborlay'),
('house', 'ousehay'), ('drops', 'opsdray'), ('loose', 'ooselay'), ('tempt',
'empttay'), ('place', 'aceplay'), ('local', 'ocallay'), ('seats', 'eatssay'),
('gates', 'atesgay'), ('dropt', 'optdray'), ('cares', 'arescay'), ('falls',
'allsfay'), ('speak', 'eakspay'), ('shops', 'opsshay'), ('groom', 'oomgray'),
('sugar', 'ugarsay'), ('front', 'ontfray'), ('visit', 'isitvay'), ('raise',
'aiseray'), ('fetch', 'etchfay'), ('silks', 'ilkssay'), ('beset', 'esetbay'),
('rocks', 'ocksray'), ('start', 'artstay'), ('winks', 'inksway'), ('sends',

'endssay'), ('piano', 'ianopay'), ('fully', 'ullyfay'), ('widow', 'idowway'), ('wiser', 'iserway'), ('dirty', 'irtyday'), ('below', 'elowbay'), ('manor', 'anormay'), ('scale', 'alescay'), ('whole', 'olewhay'), ('minds', 'indsmay'), ('glebe', 'ebeglay'), ('fifty', 'iftyfay'), ('years', 'yearsay'), ('story', 'orystay'), ('fresh', 'eshfray'), ('begun', 'egunbay'), ('dance', 'anceday'), ('jokes', 'okesjay'), ('turns', 'urnstay'), ('honor', 'onorhay'), ('spite', 'itespay'), ('green', 'eengray'), ('saint', 'aintsay'), ('hated', 'atedhay'), ('doors', 'oorsday'), ('shown', 'ownshay'), ('power', 'owerpay'), ('today', 'odaytay'), ('floor', 'oorflay'), ('force', 'orcefay'), ('right', 'ightray'), ('trick', 'icktray'), ('dread', 'eaddray'), ('final', 'inalfay'), ('shake', 'akeshay'), ('lamps', 'ampslay'), ('dying', 'yingday'), ('black', 'ackblay'), ('grand', 'andgray'), ('field', 'ieldfay'), ('plead', 'eadplay'), ('loved', 'ovedlay'), ('terms', 'ermstay'), ('spend', 'endspay'), ('mohrs', 'ohrsmay'), ('young', 'youngay'), ('great', 'eatgray'), ('guess', 'uessgay'), ('ninny', 'innynay'), ('hasty', 'astyhay'), ('nerve', 'ervenay'), ('meals', 'ealsmay'), ('three', 'eethray'), ('shade', 'adeshay'), ('store', 'orestay'), ('daily', 'ailyday'), ('taste', 'astetay'), ('gross', 'ossgray'), ('dares', 'aresday'), ('grate', 'ategray'), ('stuff', 'uffstay'), ('sheet', 'eetshay')], (14, 18): [('breakfast-room', 'eakfastbray-oomray'), ('lavender-water', 'avenderlay-aterway'), ('broken-hearted', 'okenbray-eartedhay'), ('spunging-house', 'ungingspay-ousehay')], (9, 12): [('unwelcome', 'unwelcomeway'), ('exhausted', 'exhaustedway'), ('intrusion', 'intrusionway'), ('untouched', 'untouchedway'), ('interview', 'interviewway'), ('effusions', 'effusionsway'), ('impartial', 'impartialway'), ('inventive', 'inventiveway'), ('emergency', 'emergencyway'), ('officious', 'officiousway'), ('everybody', 'everybodyway'), ('observant', 'observantway'), ('emptiness', 'emptinessway'), ('occasions', 'occasionsway'), ('assembled', 'assembledway'), ('admirable', 'admirableway'), ('uselessly', 'uselesslyway'), ('excellent', 'excellentway'), ('associate', 'associateway'), ('affecting', 'affectingway'), ('absurdity', 'absurdityway'), ('allowable', 'allowableway'), ('appointed', 'appointedway'), ('overheard', 'overheardway'), ('improving', 'improvingway'), ('eagerness', 'eagernessway'), ('unluckily', 'unluckilyway'), ('allowance', 'allowanceway'), ('estimable', 'estimableway'), ('obviating', 'obviatingway'), ('entrusted', 'entrustedway'), ('arranging', 'arrangingway'), ('exclaimed', 'exclaimedway'), ('ingenuity', 'ingenuityway'), ('outwardly', 'outwardlyway'), ('expensive', 'expensiveway'), ('entreated', 'entreatedway'), ('unconcern', 'unconcernway'), ('obtaining', 'obtainingway'), ('interests', 'interestsway'), ('assiduous', 'assiduousway'), ('adjusting', 'adjustingway'), ('ungenteel', 'ungenteelway'), ('encourage', 'encourageway'), ('apartment', 'apartmentway'), ('inheritor', 'inheritorway'), ('unceasing', 'unceasingway'), ('insulting', 'insultingway'), ('unchanged', 'unchangedway'), ('animation', 'animationway'), ('enclosure', 'enclosureway'), ('apologize', 'apologizeway'), ('earnestly', 'earnestlyway'), ('exquisite', 'exquisiteway'), ('attending', 'attendingway'), ('invention', 'inventionway'), ('enquiries', 'enquiriesway'), ('appetites', 'appetitesway'), ('additions', 'additionsway'), ('impudence', 'impudenceway'), ('insincere', 'insincereway'), ('abundance', 'abundanceway'), ('exchanged', 'exchangedway'), ('extremely', 'extremelyway'), ('inflicted', 'inflictedway'), ('attracted', 'attractedway'), ('inelegant', 'inelegantway'), ('undoubted', 'undoubtedway'), ('etiquette', 'etiquetteway'),

('unfeigned', 'unfeignedway'), ('agreeable', 'agreeableway'), ('unsubdued', 'unsubduedway'), ('inspiring', 'inspiringway'), ('unguarded', 'unguardedway'), ('advancing', 'advancingway'), ('undressed', 'undressedway'), ('excursive', 'excursiveway'), ('extending', 'extendingway'), ('extolling', 'extollingway'), ('offending', 'offendingway'), ('eloquence', 'eloquenceway'), ('ornaments', 'ornamentsway'), ('intruding', 'intrudingway'), ('incessant', 'incessantway'), ('unwilling', 'unwillingway'), ('annamaria', 'annamariaway'), ('unbounded', 'unboundedway'), ('incurable', 'incurableway'), ('unnoticed', 'unnoticedway'), ('expressly', 'expresslyway'), ('impetuous', 'impetuousway'), ('appearing', 'appearingway'), ('ingenious', 'ingeniousway'), ('amazement', 'amazementway'), ('animating', 'animatingway'), ('interfere', 'interfereway'), ('ascending', 'ascendingway'), ('affection', 'affectionway'), ('undergone', 'undergoneway'), ('unstudied', 'unstudiedway'), ('attempted', 'attemptedway'), ('infirmity', 'infirmityway'), ('apologies', 'apologiesway'), ('elsewhere', 'elsewhereway'), ('expecting', 'expectingway'), ('informing', 'informingway'), ('admission', 'admissionway'), ('extremity', 'extremityway')], (6, 9): [('unread', 'unreadway'), ('owning', 'owningway'), ('odious', 'odiousway'), ('either', 'eitherway'), ('insult', 'insultway'), ('awhile', 'awhileway'), ('assist', 'assistway'), ('injure', 'injureway'), ('eluded', 'eludedway'), ('others', 'othersway'), ('untied', 'untiedway'), ('excess', 'excessway'), ('urgent', 'urgentway'), ('appear', 'appearway'), ('aspect', 'aspectway'), ('issued', 'issuedway'), ('evince', 'evinceway'), ('eating', 'eatingway'), ('easter', 'easterway'), ('arisen', 'arisenway'), ('earned', 'earnedway'), ('esteem', 'esteemway'), ('accuse', 'accuseway'), ('owners', 'ownersway'), ('events', 'eventsway'), ('energy', 'energyway'), ('affair', 'affairway'), ('effect', 'effectway'), ('unlike', 'unlikeway'), ('unison', 'unisonway'), ('enough', 'enoughway'), ('invite', 'inviteway'), ('income', 'incomeway'), ('extent', 'extentway'), ('office', 'officeway'), ('ankles', 'anklesway'), ('admire', 'admireway'), ('induce', 'induceway'), ('infant', 'infantway'), ('expand', 'expandway'), ('orders', 'ordersway'), ('occupy', 'occupyway'), ('attend', 'attendway'), ('unable', 'unableway'), ('olives', 'olivesway'), ('awaken', 'awakenway'), ('engage', 'engageway'), ('eldest', 'eldestway'), ('eyeing', 'eyeingway'), ('advice', 'adviceway'), ('always', 'alwaysway'), ('insist', 'insistway'), ('accept', 'acceptway'), ('indies', 'indiesway'), ('excite', 'exciteway'), ('openly', 'openlyway'), ('oblige', 'obligeway'), ('amidst', 'amidstway'), ('across', 'acrossway'), ('afraid', 'afraidway'), ('uproar', 'uproarway')], (8, 10): [('villages', 'illagesvay'), ('doubtful', 'oubtfulday'), ('deceived', 'eceivedday'), ('foreseen', 'oreseenfay'), ('dennison', 'ennisonday'), ('steadily', 'eadilystay'), ('startled', 'artledstay'), ('drawings', 'awingsdray'), ('rejected', 'ejectedray'), ('deserves', 'eservesday'), ('curricle', 'urriclecay'), ('domestic', 'omesticday'), ('believed', 'elievedbay'), ('packages', 'ackagespay'), ('kneeling', 'eelingknay'), ('degraded', 'egradedday'), ('medicine', 'edicinemay'), ('promoted', 'omotedpray'), ('wondered', 'onderedway'), ('reproach', 'eproachray'), ('somerset', 'omersetsay'), ('contents', 'ontentscay'), ('children', 'ildrenchay'), ('politics', 'oliticspay'), ('hastened', 'astenedhay'), ('relative', 'elativeray'), ('henshawe', 'enshawehay'), ('material', 'aterialmay'), ('welcomed', 'elcomedway'), ('weddings',

'eddingsway'), ('diverted', 'ivertedday'), ('patience', 'atiencepay'),
('delivery', 'eliveryday'), ('hazarded', 'azardedhay'), ('happiest',
'appiesthay'), ('trifling', 'iflingtray'), ('casement', 'asementcay'),
('reformed', 'eformedray'), ('demanded', 'emandedday'), ('ceremony',
'eremonycay'), ('vigorous', 'igorousvay'), ('mornings', 'orningsmay'),
('hilarity', 'ilarityhay'), ('daughter', 'aughterday'), ('vexation',
'exationvay'), ('shocking', 'ockingshay'), ('previous', 'eviouspray'),
('pleasant', 'easantplay'), ('candidly', 'andidlycay'), ('cheerful',
'eerfulchay'), ('festival', 'estivalfay'), ('changing', 'angingchay'),
('conveyed', 'onveyedcay'), ('lombardy', 'ombardylay'), ('militate',
'ilitatemay'), ('humanity', 'umanityhay'), ('giggling', 'igglinggay'),
('standard', 'andardstay'), ('returned', 'eturnedray'), ('proclaim',
'oclaimpray'), ('ridicule', 'idiculeray'), ('pointers', 'ointerspay'),
('grounded', 'oundedgray'), ('contrast', 'ontrastcay'), ('feelings',
'eelingsfay'), ('mistakes', 'istakesmay'), ('prepared', 'eparedpray'),
('stranger', 'angerstray'), ('habitual', 'abitualhay'), ('firmness',
'irmnessfay'), ('rational', 'ationalray'), ('regulate', 'egulateray'),
('claiming', 'aimingclay'), ('knocking', 'ockingknay'), ('servants',
'ervantssay'), ('goodness', 'oodnessgay'), ('mutually', 'utuallymay'),
('legacies', 'egacieslay'), ('sisterly', 'isterlysay'), ('punctual',
'unctualpay'), ('deciding', 'ecidingday'), ('suitable', 'uitablesay'),
('guarding', 'uardinggay'), ('converse', 'onversecay'), ('slighter',
'ighterslay'), ('dissuade', 'issuadeday'), ('despatch', 'espatchday'),
('blackest', 'ackestblay'), ('recalled', 'ecalledray'), ('loitered',
'oiteredlay'), ('dreading', 'eadingdray'), ('relieved', 'elievedray'),
('westward', 'estwardway'), ('thistles', 'istlesthay'), ('defended',
'efendedday'), ('contempt', 'ontemptcay'), ('sensibly', 'ensiblysay'),
('romantic', 'omanticray'), ('comforts', 'omfortscay'), ('jealousy',
'ealousyjay'), ('repeated', 'epeatedray'), ('banditti', 'andittibay'),
('greatest', 'eatestgray'), ('mistrust', 'istrustmay'), ('forwards',
'orwardsfay'), ('rewarded', 'ewardedray'), ('watching', 'atchingway'),
('negative', 'egativenay'), ('sympathy', 'ympathysay'), ('helpless',
'elplesshay'), ('quickest', 'ickestquay'), ('portrait', 'ortraitpay'),
('consoled', 'onsoledcay'), ('fourteen', 'ourteenfay'), ('retarded',
'etardedray'), ('comments', 'ommentscay'), ('jennings', 'enningsjay'),
('securing', 'ecuringsay'), ('withdrew', 'ithdrewway'), ('youthful',
'youthfulay'), ('bachelor', 'achelorbay'), ('convince', 'onvincecay'),
('measured', 'easuredmay'), ('goodwill', 'oodwillgay'), ('branched',
'anchedbray'), ('purchase', 'urchasepay'), ('covenant', 'ovenantcay'),
('reseated', 'eseatedray'), ('concerns', 'oncernscay'), ('hardship',
'ardshiphay'), ('lessened', 'essenedlay'), ('delaford', 'elafordday'),
('governed', 'overnedgay'), ('listened', 'istenedlay'), ('constant',
'onstantcay'), ('fatigued', 'atiguedfay'), ('persuade', 'ersuadepay'),
('critique', 'itiquecray'), ('remedies', 'emediesray'), ('stepping',
'eppingstay'), ('kindness', 'indnesskay'), ('sociable', 'ociablesay'),
('meetings', 'eetingsmay'), ('deserved', 'eservedday'), ('thoughts',
'oughtsthay'), ('becoming', 'ecomingbay'), ('heartily', 'eartilyhay'),
('promised', 'omisedpray'), ('shutting', 'uttingshay'), ('reproved',

'eprovedray'), ('ladyship', 'adyshiplay'), ('pursuing', 'ursuingpay'),
('residing', 'esidingray'), ('sentence', 'entencesay'), ('revealed',
'evealedray'), ('decrease', 'ecreaseday'), ('speedily', 'eedilyspay'),
('commands', 'ommandscay'), ('mistaken', 'istakenmay'), ('settling',
'ettlingsay'), ('polished', 'olishedpay'), ('stanhill', 'anhillstay'),
('retailed', 'etailedray'), ('pointing', 'ointingpay'), ('strongly',
'onglystray'), ('subjects', 'ubjectssay'), ('calmness', 'almnesscay'),
('doctrine', 'octrineday'), ('handsome', 'andsomehay'), ('profited',
'ofitedpray'), ('planning', 'anningplay'), ('denoting', 'enotingday'),
('commonly', 'ommonlycay'), ('bestowed', 'estowedbay'), ('marriage',
'arriagemay'), ('presents', 'esentspray'), ('complete', 'ompletecay'),
('striking', 'ikingstray'), ('needless', 'eedlessnay'), ('plighted',
'ightedplay'), ('tranquil', 'anquiltray'), ('spacious', 'aciousspay'),
('nonsense', 'onsensenay'), ('likeness', 'ikenesslay'), ('valuable',
'aluablevay'), ('laudable', 'audablelay'), ('delirium', 'eliriumday'),
('believes', 'elievesbay'), ('premises', 'emisespray'), ('presumed',
'esumedpray'), ('coldness', 'oldnesscay'), ('metaphor', 'etaphormay'),
('thankful', 'ankfulthay'), ('suffered', 'ufferedsay'), ('preserve',
'eservepray'), ('resorted', 'esortedray'), ('felicity', 'elicityfay'),
('betrayed', 'etrayedbay'), ('throwing', 'owingthray'), ('powerful',
'owerfulpay'), ('humoured', 'umouredhay'), ('practise', 'actisepray'),
('coloured', 'olouredcay'), ('regiment', 'egimentray'), ('stubborn',
'ubbornstay'), ('huswifes', 'uswifeshay'), ('painting', 'aintingpay'),
('friendly', 'iendlyfray'), ('happened', 'appenedhay'), ('moderate',
'oderatemay'), ('solitary', 'olitarysay'), ('wildness', 'ildnessway'),
('marianne', 'ariannemay'), ('delicacy', 'elicacyday'), ('progress',
'ogresspray'), ('coachman', 'oachmancay'), ('grieving', 'ievinggray'),
('troubles', 'oublestray'), ('coolness', 'oolnesscay'), ('possibly',
'ossiblypay'), ('relating', 'elatingray'), ('patterns', 'atternspay'),
('consists', 'onsistscay'), ('creating', 'eatingcray'), ('confused',
'onfusedcay'), ('critical', 'iticalcray'), ('laboured', 'abouredlay'),
('minutely', 'inutelymay'), ('tendency', 'endencytay'), ('distrust',
'istrustday'), ('distress', 'istressday'), ('conclude', 'oncludecay'),
('sixpence', 'ixpencesay'), ('somewhat', 'omewhatsay'), ('coincide',
'oincidecay'), ('harmless', 'armlesshay'), ('juvenile', 'uvenilejay'),
('pleasing', 'easingplay'), ('precious', 'eciouspray'), ('restrain',
'estrainray'), ('produced', 'oducedpray'), ('renewing', 'enewingray'),
('pinching', 'inchingpay'), ('severely', 'everelysay'), ('blossoms',
'ossomsblay'), ('heighten', 'eightenhay'), ('wandered', 'anderedway'),
('properly', 'operlypray'), ('veracity', 'eracityvay'), ('released',
'eleasedray'), ('transact', 'ansacttray'), ('helpmate', 'elpmatehay'),
('business', 'usinessbay'), ('concerto', 'oncertocay'), ('reserved',
'eservedray'), ('reliance', 'elianceray'), ('headache', 'eadachehay'),
('families', 'amiliesfay'), ('centered', 'enteredcay'), ('mulberry',
'ulberrymay'), ('sensible', 'ensiblesay'), ('steepest', 'eepeststay'),
('compared', 'omparedcay'), ('speaking', 'eakingspay'), ('censured',
'ensuredcay'), ('screened', 'eenedscray'), ('pictures', 'icturespay'),
('reaching', 'eachingray'), ('wonderer', 'ondererway'), ('whenever',

'eneverwhay'), ('detested', 'etestedday'), ('dispatch', 'ispatchday'),
('shutters', 'uttersshay'), ('disgrace', 'isgraceday'), ('dejected',
'ejectedday'), ('conquest', 'onquestcay'), ('foretell', 'oretellfay'),
('weakened', 'eakenedway'), ('brooding', 'oodingbray'), ('shedding',
'eddingshay'), ('poignant', 'oignantpay'), ('trembled', 'embledtray'),
('resolute', 'esoluteray'), ('reverted', 'evertedray'), ('silenced',
'ilencedsay'), ('jointure', 'ointurejay')], (10, 13): [('illaudable',
'illaudableway'), ('incivility', 'incivilityway'), ('oppression',
'oppressionway'), ('acquainted', 'acquaintedway'), ('economical',
'economicalway'), ('ebullition', 'ebullitionway'), ('affronting',
'affrontingway'), ('unforeseen', 'unforeseenway'), ('auspicious',
'auspiciousway'), ('indecorous', 'indecorousway'), ('attractive',
'attractiveway'), ('exercising', 'exercisingway'), ('unsatiable',
'unsatiableway'), ('individual', 'individualway'), ('enthusiasm',
'enthusiasmway'), ('announcing', 'announcingway'), ('indistinct',
'indistinctway'), ('affliction', 'afflictionway'), ('imaginable',
'imaginableway'), ('obstructed', 'obstructedway'), ('improperly',
'improperlyway'), ('illiterate', 'illiterateway'), ('interested',
'interestedway'), ('infamously', 'infamouslyway'), ('amusements',
'amusementsway'), ('endeavours', 'endeavoursway'), ('inhabiting',
'inhabitingway'), ('indisposed', 'indisposedway'), ('acquiesced',
'acquiescedway'), ('anticipate', 'anticipateway'), ('accurately',
'accuratelyway'), ('influenced', 'influencedway'), ('assemblies',
'assembliesway'), ('approached', 'approachedway'), ('accusation',
'accusationway'), ('apparently', 'apparentlyway'), ('intimately',
'intimatelyway'), ('uneasiness', 'uneasinessway'), ('alteration',
'alterationway'), ('artificial', 'artificialway'), ('admiration',
'admirationway'), ('accidental', 'accidentalway'), ('unexpected',
'unexpectedway'), ('insensible', 'insensibleway'), ('expressing',
'expressingway'), ('accelerate', 'accelerateway'), ('everything',
'everythingway'), ('increasing', 'increasingway'), ('undervalue',
'undervalueway'), ('invaluable', 'invaluableway'), ('affections',
'affectionsway'), ('infinitely', 'infinitelyway'), ('unsuitable',
'unsuitableway'), ('authorised', 'authorisedway'), ('impudently',
'impudentlyway'), ('unpardoned', 'unpardonedway'), ('inevitable',
'inevitableway'), ('ungenerous', 'ungenerousway'), ('impression',
'impressionway'), ('indelicacy', 'indelicacyway'), ('inducement',
'inducementway'), ('unaffected', 'unaffectedway'), ('instigated',
'instigatedway'), ('inquietude', 'inquietudeway'), ('attempting',
'attemptingway'), ('experience', 'experienceway'), ('impatience',
'impatienceway'), ('indulgence', 'indulgenceway'), ('affability',
'affabilityway'), ('unavailing', 'unavailingway'), ('incommoded',
'incommodedway'), ('altogether', 'altogetherway')], (4, 6): [('done', 'oneday'),
('wife', 'ifeway'), ('plea', 'eaplay'), ('mall', 'allmay'), ('sang', 'angsay'),
('clue', 'ueclay'), ('bell', 'ellbay'), ('bind', 'indbay'), ('join', 'oinjay'),
('went', 'entway'), ('vile', 'ilevay'), ('hens', 'enshay'), ('tore', 'oretay'),
('rage', 'ageray'), ('flow', 'owflay'), ('gate', 'ategay'), ('palm', 'almpay'),
('give', 'ivegay'), ('lips', 'ipslay'), ('sink', 'inksay'), ('rent', 'entray'),

```
('gave', 'avegay'), ('joke', 'okejay'), ('miss', 'issmay'), ('mere', 'eremay'),
('made', 'ademay'), ('dogs', 'ogsday'), ('bent', 'entbay'), ('duly', 'ulyday'),
('when', 'enwhay'), ('rate', 'ateray'), ('sure', 'uresay'), ('land', 'andlay'),
('pink', 'inkpay'), ('hint', 'inthay'), ('cast', 'astcay'), ('park', 'arkpay'),
('bold', 'oldbay'), ('door', 'oorday'), ('bulk', 'ulkbay'), ('doat', 'oatday'),
('coat', 'oatcay'), ('date', 'ateday'), ('foot', 'ootfay'), ('hand', 'andhay'),
('damp', 'ampday'), ('pall', 'allpay'), ('goes', 'oesgay'), ('rung', 'ungray'),
('none', 'onenay'), ('game', 'amegay'), ('nods', 'odsnay'), ('read', 'eadray'),
('gold', 'oldgay'), ('sing', 'ingsay'), ('puts', 'utspay'), ('cats', 'atscay'),
('many', 'anymay'), ('dare', 'areday'), ('noon', 'oonnay'), ('said', 'aidsay'),
('mark', 'arkmay'), ('case', 'asecay'), ('hate', 'atehay'), ('word', 'ordway'),
('beat', 'eatbay'), ('lose', 'oselay'), ('fish', 'ishfay'), ('laid', 'aidlay'),
('fine', 'inefay'), ('snug', 'ugsnay'), ('moss', 'ossmay'), ('heir', 'eirhay'),
('sigh', 'ighsay'), ('mend', 'endmay'), ('sake', 'akesay'), ('pass', 'asspay'),
('debt', 'ebtday'), ('lane', 'anelay'), ('days', 'aysday'), ('deep', 'eepday'),
('song', 'ongsay'), ('spot', 'otspay'), ('took', 'ooktay'), ('both', 'othbay'),
('year', 'yearay'), ('mama', 'amamay'), ('hill', 'illhay'), ('beds', 'edsbay'),
('less', 'esslay'), ('gown', 'owngay'), ('kill', 'illkay'), ('free', 'eefray'),
('hall', 'allhay'), ('room', 'oomray'), ('gray', 'aygray'), ('beef', 'eefbay'),
('kept', 'eptkay'), ('what', 'atwhay'), ('blue', 'ueblay'), ('fell', 'ellfay'),
('draw', 'awdray'), ('play', 'ayplay'), ('baby', 'abybay'), ('rude', 'uderay'),
('trim', 'imtray'), ('chat', 'atchay'), ('bely', 'elybay'), ('fled', 'edflay'),
('past', 'astpay'), ('wait', 'aitway'), ('meal', 'ealmay'), ('wont', 'ontway'),
('week', 'eekway'), ('best', 'estbay'), ('line', 'inelay'), ('lawn', 'awnlay'),
('tree', 'eetray'), ('tend', 'endtay'), ('find', 'indfay'), ('five', 'ivefay'),
('gone', 'onegay'), ('need', 'eednay'), ('drew', 'ewdray'), ('more', 'oremay'),
('cool', 'oolcay'), ('suit', 'uitsay'), ('from', 'omfray'), ('cure', 'urecay'),
('will', 'illway'), ('slow', 'owslay'), ('they', 'eythay'), ('name', 'amenay'),
('shot', 'otshay'), ('come', 'omecay'), ('tied', 'iedtay'), ('came', 'amecay'),
('boys', 'oysbay'), ('wish', 'ishway'), ('hour', 'ourhay'), ('kind', 'indkay'),
('mile', 'ilemay'), ('quit', 'itquay'), ('john', 'ohnjay'), ('hazy', 'azyhay'),
('fall', 'allfay'), ('sell', 'ellsay'), ('tale', 'aletay'), ('grew', 'ewgray'),
('seek', 'eeksay'), ('muff', 'uffmay'), ('soft', 'oftsay'), ('same', 'amesay'),
('body', 'odybay'), ('head', 'eadhay'), ('like', 'ikelay'), ('here', 'erehay'),
('help', 'elphay'), ('sail', 'ailsay'), ('pope', 'opepay'), ('deny', 'enyday'),
('bond', 'ondbay'), ('bath', 'athbay'), ('blow', 'owblay'), ('hold', 'oldhay'),
('safe', 'afesay'), ('path', 'athpay'), ('feed', 'eedfay'), ('thus', 'usthay'),
('four', 'ourfay'), ('lest', 'estlay'), ('thin', 'inthay')], (11, 14):
[('approaching', 'approachingway'), ('unequivocal', 'unequivocalway'),
('exceedingly', 'exceedinglyway'), ('application', 'applicationway'),
('observation', 'observationway'), ('unhappiness', 'unhappinessway'),
('interrupted', 'interruptedway'), ('enlightened', 'enlightenedway'),
('appearances', 'appearancesway'), ('unconnected', 'unconnectedway'),
('unconcerned', 'unconcernedway'), ('indignation', 'indignationway'),
('accommodate', 'accommodateway'), ('inattention', 'inattentionway'),
('undoubtedly', 'undoubtedlyway'), ('intercourse', 'intercourseway'),
('imperfectly', 'imperfectlyway'), ('imprudently', 'imprudentlyway'),
('affectation', 'affectationway'), ('instigation', 'instigationway'),
```

```
('inclination', 'inclinationway'), ('undervalued', 'undervaluedway'),
('affirmative', 'affirmativeway'), ('inconstancy', 'inconstancyway'),
('universally', 'universallyway'), ('engagements', 'engagementsway'),
('exclamation', 'exclamationway'), ('elucidation', 'elucidationway'),
('accustomary', 'accustomaryway'), ('approbation', 'approbationway'),
('unnecessary', 'unnecessaryway'), ('associating', 'associatingway'),
('attributing', 'attributingway'), ('unobtrusive', 'unobtrusiveway'),
('impropriety', 'improprietyway'), ('examination', 'examinationway'),
('alterations', 'alterationsway'), ('unexpressed', 'unexpressedway'),
('abridgement', 'abridgementway'), ('exhilarated', 'exhilaratedway'),
('unsuspected', 'unsuspectedway'), ('independent', 'independentway'),
('alternately', 'alternatelyway'), ('enforcement', 'enforcementway'),
('undertaking', 'undertakingway'), ('outstripped', 'outstrippedway'),
('appointment', 'appointmentway'), ('explanation', 'explanationway'),
('experienced', 'experiencedway'), ('unknowingly', 'unknowinglyway'),
('undesirable', 'undesirableway'), ('ascertained', 'ascertainedway'),
('enumeration', 'enumerationway'), ('alleviation', 'alleviationway'),
('instruction', 'instructionway'), ('undeserving', 'undeservingway'),
('earnestness', 'earnestnessway'), ('insincerity', 'insincerityway'),
('accompanied', 'accompaniedway'), ('employments', 'employmentsway'),
('inferiority', 'inferiorityway'), ('inexcusable', 'inexcusableway'),
('introducing', 'introducingway'), ('extravagant', 'extravagantway'),
('impatiently', 'impatientlyway'), ('attentively', 'attentivelyway'),
('eligibility', 'eligibilityway'), ('uncertainty', 'uncertaintyway')], (7, 10):
[('anxiety', 'anxietyway'), ('attacks', 'attacksway'), ('ailment',
'ailmentway'), ('awkward', 'awkwardway'), ('effects', 'effectsway'), ('induced',
'inducedway'), ('uncivil', 'uncivilway'), ('against', 'againstway'), ('apology',
'apologyway'), ('endless', 'endlessway'), ('alluded', 'alludedway'), ('ability',
'abilityway'), ('address', 'addressway'), ('engaged', 'engagedway'), ('eagerly',
'eagerlyway'), ('expects', 'expectsway'), ('elliott', 'elliottway'), ('oneself',
'oneselfway'), ('anybody', 'anybodyway'), ('augment', 'augmentway'), ('utility',
'utilityway'), ('unquiet', 'unquietway'), ('infancy', 'infancyway'), ('altered',
'alteredway'), ('unequal', 'unequalway'), ('adapted', 'adaptedway'), ('useless',
'uselessway'), ('opposed', 'opposedway'), ('artless', 'artlessway'), ('imagery',
'imageryway'), ('amiably', 'amiablyway'), ('uncouth', 'uncouthway'), ('affixed',
'affixedway'), ('impulse', 'impulseway'), ('opinion', 'opinionway'), ('imitate',
'imitateway'), ('admires', 'admiresway'), ('affront', 'affrontway'), ('amusing',
'amusingway'), ('accents', 'accentsway'), ('implied', 'impliedway'), ('ecstasy',
'ecstasyway'), ('illness', 'illnessway'), ('entreat', 'entreatway'), ('another',
'anotherway'), ('opening', 'openingway'), ('england', 'englandway'), ('ensured',
'ensuredway'), ('imputed', 'imputedway'), ('enabled', 'enabledway'), ('offence',
'offenceway'), ('insured', 'insuredway'), ('urgency', 'urgencyway'), ('amazing',
'amazingway'), ('uniform', 'uniformway'), ('avarice', 'avariceway'), ('ellison',
'ellisonway'), ('endowed', 'endowedway'), ('ashamed', 'ashamedway'), ('eloping',
'elopingway'), ('inhabit', 'inhabitway'), ('advised', 'advisedway'), ('attract',
'attractway'), ('already', 'alreadyway'), ('enquire', 'enquireway'), ('affects',
'affectsway'), ('invalid', 'invalidway')], (9, 11): [('genuinely',
'enuinelygay'), ('columella', 'olumellacay'), ('vouchsafe', 'ouchsafevay'),
```

('forwarded', 'orwardedfay'), ('scattered', 'atteredscay'), ('christmas',
'istmaschray'), ('crimsoned', 'imsonedcray'), ('contrives', 'ontrivescay'),
('miserable', 'iserablemay'), ('preceding', 'ecedingpray'), ('healthful',
'ealthfulhay'), ('confiding', 'onfidingcay'), ('perfectly', 'erfectlypay'),
('prospects', 'ospectspray'), ('withdrawn', 'ithdrawnway'), ('lingering',
'ingeringlay'), ('contained', 'ontainedcay'), ('depressed', 'epressedday'),
('disgusted', 'isgustedday'), ('promoting', 'omotingpray'), ('supported',
'upportedsay'), ('gradually', 'aduallygray'), ('disturbed', 'isturbedday'),
('perpetual', 'erpetualpay'), ('pronounce', 'onouncepray'), ('sincerely',
'incerelysay'), ('seventeen', 'eventeensay'), ('presented', 'esentedpray'),
('medicines', 'edicinesmay'), ('syllables', 'yllablessay'), ('behaviour',
'ehaviourbay'), ('niggardly', 'iggardlynay'), ('commonest', 'ommonestcay'),
('necessary', 'ecessarynay'), ('prettiest', 'ettiestpray'), ('condemned',
'ondemnedcay'), ('conquests', 'onquestscay'), ('fashioned', 'ashionedfay'),
('hysterics', 'ystericshay'), ('smallness', 'allnesssmay'), ('professed',
'ofessedpray'), ('conceited', 'onceitedcay'), ('compliant', 'ompliantcay'),
('frankness', 'anknessfray'), ('merriment', 'errimentmay'), ('resembled',
'esembledray'), ('leisurely', 'eisurelylay'), ('dispersed', 'ispersedday'),
('departure', 'epartureday'), ('pecuniary', 'ecuniarypay'), ('frivolous',
'ivolousfray'), ('situation', 'ituationsay'), ('discourse', 'iscourseday'),
('resettled', 'esettledray'), ('beholding', 'eholdingbay'), ('gentleman',
'entlemangay'), ('chagrined', 'agrinedchay'), ('struggled', 'uggledstray'),
('preparing', 'eparingpray'), ('respected', 'espectedray'), ('newspaper',
'ewspapernay'), ('justified', 'ustifiedjay'), ('sacrifice', 'acrificesay'),
('stammered', 'ammeredstay'), ('smartness', 'artnesssmay'), ('continued',
'ontinuedcay'), ('contrived', 'ontrivedcay'), ('reconcile', 'econcileray'),
('modestest', 'odestestmay'), ('cordially', 'ordiallycay'), ('sentiment',
'entimentsay'), ('shameless', 'amelessshay'), ('reflected', 'eflectedray'),
('feelingly', 'eelinglyfay'), ('confident', 'onfidentcay'), ('proposals',
'oposalspray'), ('warehouse', 'arehouseway'), ('deviation', 'eviationday'),
('statement', 'atementstay'), ('submitted', 'ubmittedsay'), ('believing',
'elievingbay'), ('gentility', 'entilitygay'), ('resources', 'esourcesray'),
('composure', 'omposurecay'), ('deserving', 'eservingday'), ('vehemence',
'ehemencevay'), ('disgraced', 'isgracedday'), ('perplexed', 'erplexedpay'),
('prejudice', 'ejudicepray'), ('quickened', 'ickenedquay'), ('blundered',
'underedblay'), ('strongest', 'ongeststray'), ('prevailed', 'evailedpray'),
('languages', 'anguageslay'), ('reanimate', 'eanimateray'), ('stationer',
'ationerstay'), ('reminding', 'emindingray'), ('blameless', 'amelessblay'),
('principle', 'inciplepray'), ('hardening', 'ardeninghay'), ('suggested',
'uggestedsay'), ('desperate', 'esperateday'), ('landscape', 'andscapelay'),
('paragraph', 'aragraphpay'), ('provision', 'ovisionpray'), ('relations',
'elationsray'), ('fruitless', 'uitlessfray'), ('christian', 'istianchray'),
('stipulate', 'ipulatestay'), ('constancy', 'onstancycay'), ('gentlemen',
'entlemengay'), ('venturing', 'enturingvay'), ('northward', 'orthwardnay'),
('neighbour', 'eighbournay'), ('weakening', 'eakeningway'), ('persisted',
'ersistedpay'), ('malignant', 'alignantmay'), ('wandering', 'anderingway'),
('witnesses', 'itnessesway'), ('dangerous', 'angerousday'), ('safeguard',
'afeguardsay'), ('treasured', 'easuredtray'), ('cleveland', 'evelandclay'),

('patronage', 'atronagepay'), ('despising', 'espisingday'), ('foolishly',
'oolishlyfay'), ('seriously', 'eriouslysay'), ('restraint', 'estraintray'),
('september', 'eptembersay'), ('distrusts', 'istrustsday'), ('temperate',
'emperatetay'), ('finishing', 'inishingfay'), ('persuaded', 'ersuadedpay'),
('stupified', 'upifiedstay'), ('declaring', 'eclaringday'), ('dejection',
'ejectionday'), ('soothings', 'oothingssay'), ('conveying', 'onveyingcay'),
('possessed', 'ossessedpay'), ('described', 'escribedday'), ('spreading',
'eadingspray'), ('foresight', 'oresightfay'), ('political', 'oliticalpay'),
('concluded', 'oncludedcay'), ('doubtless', 'oubtlessday'), ('confusion',
'onfusioncay'), ('disliking', 'islikingday'), ('carriages', 'arriagescay'),
('favourite', 'avouritefay'), ('brilliant', 'illiantbray'), ('hastening',
'asteninghay'), ('precisely', 'eciselypray'), ('wondering', 'onderingway'),
('shoulders', 'oulderssshay'), ('honorably', 'onorablyhay'), ('privilege',
'ivilegepray'), ('deference', 'eferenceday'), ('vulgarity', 'ulgarityvay'),
('reprimand', 'eprimandray'), ('surveying', 'urveyingsay'), ('handsomer',
'andsomerhay'), ('shivering', 'iveringshay'), ('tolerable', 'olerabletay'),
('requiring', 'equiringray'), ('precision', 'ecisionpray'), ('courtland',
'ourtlandcay'), ('sportsman', 'ortsmanspay'), ('happiness', 'appinesshay'),
('pointedly', 'ointedlypay'), ('benefited', 'enefitedbay'), ('substance',
'ubstancesay'), ('beginning', 'eginningbay'), ('gratitude', 'atitudegray'),
('negligent', 'egligentnay'), ('belonging', 'elongingbay'), ('mortified',
'ortifiedmay'), ('neglected', 'eglectednay'), ('shillings', 'illingsshay'),
('warranted', 'arrantedway'), ('performed', 'erformedpay'), ('necessity',
'ecessitynay'), ('slightest', 'ightestslay'), ('guardians', 'uardiansgay'),
('preferred', 'eferredpray'), ('forgotten', 'orgottenfay'), ('refreshed',
'efreshedray'), ('barouches', 'arouchesbay'), ('grievance', 'ievancegray'),
('seduction', 'eductionsay'), ('comprised', 'omprisedcay'), ('listening',
'isteninglay'), ('detection', 'etectionday'), ('narration', 'arrationnay'),
('deceiving', 'eceivingday'), ('provoking', 'ovokingpray'), ('cessation',
'essationcay'), ('questions', 'estionsquay'), ('continual', 'ontinualcay'),
('faintness', 'aintnessfay'), ('diffident', 'iffidentday'), ('referring',
'eferringray'), ('blessings', 'essingsblay'), ('therefore', 'ereforethay'),
('frugality', 'ugalityfray'), ('reprobate', 'eprobateray')], (11, 13):
[('tranquilize', 'anquilizetray'), ('fashionable', 'ashionablefay'),
('respectable', 'espectableray'), ('publication', 'ublicationpay'),
('superintend', 'uperintendsay'), ('recollected', 'ecollectedray'),
('cheerfuller', 'eerfullerchay'), ('concurrence', 'oncurrencecay'),
('restorative', 'estorativeray'), ('reproaching', 'eproachingray'),
('declaration', 'eclarationday'), ('discharging', 'ischargingday'),
('proclaiming', 'oclaimingpray'), ('discovering', 'iscoveringday'),
('possibility', 'ossibilitypay'), ('contributed', 'ontributedcay'),
('captivating', 'aptivatingcay'), ('misinformed', 'isinformedmay'),
('carefulness', 'arefulnesscay'), ('philosophic', 'ilosophicphay'),
('calculation', 'alculationcay'), ('consumption', 'onsumptioncay'),
('confinement', 'onfinementcay'), ('certainties', 'ertaintiescay'),
('destination', 'estinationday'), ('separations', 'eparationssay'),
('voluntarily', 'oluntarilyvay'), ('connoisseur', 'onnoisseurcay'),
('refinements', 'efinementsray'), ('subservient', 'ubservientsay'),

```
('marlborough', 'arlboroughmay'), ('sagaciously', 'agaciouslysay'),
('fascinating', 'ascinatingfay'), ('comfortable', 'omfortablecay'),
('reflections', 'eflectionsray'), ('sensibility', 'ensibilitysay'),
('forbearance', 'orbearancefay'), ('mischievous', 'ischievousmay'),
('disgraceful', 'isgracefulday'), ('temporizing', 'emporizingtay'),
('sacrificing', 'acrificingsay'), ('spontaneous', 'ontaneousspay'),
('flourishing', 'ourishingflay'), ('lamentation', 'amentationlay'),
('meditations', 'editationsmay'), ('quarrelling', 'arrellingquay'),
('discussions', 'iscussionsday'), ('supposition', 'uppositionsay'),
('comfortably', 'omfortablycay'), ('disappeared', 'isappearedday'),
('preparation', 'eparationpray'), ('consistency', 'onsistencycay'),
('compliments', 'omplimentscay'), ('promptitude', 'omptitudepray'),
('determining', 'eterminingday'), ('countenance', 'ountenancecay'),
('maintenance', 'aintenancemay'), ('contraction', 'ontractioncay'),
('dissipation', 'issipationday'), ('comparative', 'omparativecay'),
('desperation', 'esperationday'), ('displeasure', 'ispleasureday'),
('despatching', 'espatchingday'), ('playfellows', 'ayfellowsplay'),
('plantations', 'antationsplay'), ('vindication', 'indicationvay'),
('significant', 'ignificantsay'), ('discontents', 'iscontentsday'),
('housekeeper', 'ousekeeperhay'), ('playfulness', 'ayfulnessplay'),
('threatening', 'eateningthray'), ('meritorious', 'eritoriousmay'),
('tediousness', 'ediousnesstay'), ('convenience', 'onveniencecay'),
('possessions', 'ossessionspay'), ('consolation', 'onsolationcay'),
('composition', 'ompositioncay'), ('proposition', 'opositionpray'),
('correctness', 'orrectnesscay'), ('consequence', 'onsequencecay'),
('barbarously', 'arbarouslybay'), ('booksellers', 'ooksellersbay'),
('conjectures', 'onjecturescay'), ('violoncello', 'ioloncellovay'),
('dissembling', 'issemblingday'), ('rapturously', 'apturouslyray'),
('speculative', 'eculativespay'), ('remembering', 'ememberingray'),
('probability', 'obabilitypray'), ('punctuality', 'unctualitypay'),
('forgiveness', 'orgivenessfay'), ('disapproves', 'isapprovesday'),
('troublesome', 'oublesometray')], (3, 5): [('say', 'aysay'), ('for', 'orfay'),
('get', 'etgay'), ('pin', 'inpay'), ('not', 'otnay'), ('fit', 'itfay'), ('rap',
'apray'), ('lip', 'iplay'), ('pry', 'ypray'), ('rob', 'obray'), ('fir',
'irfay'), ('can', 'ancay'), ('two', 'otway'), ('him', 'imhay'), ('son',
'onsay'), ('let', 'etlay'), ('try', 'ytray'), ('hot', 'othay'), ('sum',
'umsay'), ('got', 'otgay'), ('law', 'awlay'), ('how', 'owhay'), ('far',
'arfay'), ('the', 'ethay'), ('sky', 'yskay'), ('sly', 'yslay'), ('cut',
'utcay'), ('hop', 'ophay'), ('had', 'adhay'), ('was', 'asway'), ('bow',
'owbay'), ('hue', 'uehay'), ('now', 'ownay'), ('vex', 'exvay'), ('yet',
'yetay'), ('sit', 'itsay'), ('due', 'ueday'), ('cap', 'apcay'), ('sat',
'atsay'), ('hit', 'ithay'), ('see', 'eesay'), ('pay', 'aypay'), ('nor',
'ornay'), ('fly', 'yflay'), ('dog', 'ogday'), ('too', 'ootay'), ('six',
'ixsay'), ('lap', 'aplay'), ('new', 'ewnay'), ('led', 'edlay'), ('sin',
'insay'), ('ten', 'entay'), ('his', 'ishay'), ('day', 'ayday')], (14, 17):
[('insurmountable', 'insurmountableway'), ('inconveniences',
'inconveniencesway'), ('aggrandizement', 'aggrandizementway'),
('irreconcilable', 'irreconcilableway'), ('unacknowledged',
```

'unacknowledgedway'), ('unpleasantness', 'unpleasantnessway')], (13, 16):
[('understanding', 'understandingway'), ('intentionally', 'intentionallyway'),
('embarrassment', 'embarrassmentway'), ('unembarrassed', 'unembarrassedway'),
('administering', 'administeringway'), ('instantaneous', 'instantaneousway'),
('intrinsically', 'intrinsicallyway'), ('indispensable', 'indispensableway'),
('impoverishing', 'impoverishingway'), ('opportunities', 'opportunitiesway'),
('impertinently', 'impertinentlyway'), ('independently', 'independentlyway'),
('encroachments', 'encroachmentsway'), ('uninteresting', 'uninterestingway'),
('investigation', 'investigationway'), ('unaccountable', 'unaccountableway'),
('irrepressible', 'irrepressibleway'), ('impossibility', 'impossibilityway')],
(10, 14): [('thirty-six', 'irtythay-ixsay')], (8, 11): [('increase',
'increaseway'), ('agitated', 'agitatedway'), ('unkindly', 'unkindlyway'),
('occurred', 'occurredway'), ('arranged', 'arrangedway'), ('appetite',
'appetiteway'), ('inclined', 'inclinedway'), ('indolent', 'indolentway'),
('inquired', 'inquiredway'), ('evenings', 'eveningsway'), ('obliging',
'obligingway'), ('equalled', 'equalledway'), ('accepted', 'acceptedway'),
('extended', 'extendedway'), ('entreaty', 'entreatyway'), ('exigence',
'exigenceway'), ('election', 'electionway'), ('examined', 'examinedway'),
('inviting', 'invitingway'), ('embitter', 'embitterway'), ('idleness',
'idlenessway'), ('admirers', 'admirersway'), ('editions', 'editionsway'),
('extorted', 'extortedway'), ('outweigh', 'outweighway'), ('attended',
'attendedway'), ('addition', 'additionway'), ('entitled', 'entitledway'),
('awakened', 'awakenedway'), ('attacked', 'attackedway'), ('arriving',
'arrivingway'), ('overlook', 'overlookway'), ('eminence', 'eminenceway'),
('adhering', 'adheringway'), ('obtained', 'obtainedway'), ('openness',
'opennessway'), ('allusion', 'allusionway'), ('anything', 'anythingway'),
('overcome', 'overcomeway'), ('announce', 'announceway'), ('enjoying',
'enjoyingway'), ('outlived', 'outlivedway'), ('emphasis', 'emphasisway'),
('insulted', 'insultedway'), ('esteemed', 'esteemedway'), ('alphabet',
'alphabetway'), ('enquired', 'enquiredway'), ('accident', 'accidentway'),
('occupied', 'occupiedway'), ('ellisons', 'ellisonsway'), ('avoiding',
'avoidingway'), ('attained', 'attainedway'), ('unwarily', 'unwarilyway'),
('employed', 'employedway'), ('altering', 'alteringway'), ('intimate',
'intimateway'), ('effected', 'effectedway'), ('offences', 'offencesway'),
('entirely', 'entirelyway'), ('abounded', 'aboundedway'), ('actually',
'actuallyway'), ('identify', 'identifyway'), ('applying', 'applyingway'),
('exterior', 'exteriorway'), ('auditors', 'auditorsway'), ('unfairly',
'unfairlyway'), ('eighteen', 'eighteenway'), ('embraced', 'embracedway'),
('unfolded', 'unfoldedway'), ('injuries', 'injuriesway'), ('elevated',
'elevatedway'), ('alarming', 'alarmingway'), ('adequate', 'adequateway'),
('unabated', 'unabatedway'), ('incurred', 'incurredway'), ('uncommon',
'uncommonway'), ('unawares', 'unawaresway'), ('unsuited', 'unsuitedway'),
('instance', 'instanceway'), ('interval', 'intervalway'), ('audacity',
'audacityway')], (5, 8): [('other', 'otherway'), ('ideas', 'ideasway'),
('agony', 'agonyway'), ('avoid', 'avoidway'), ('awful', 'awfulway'), ('april',
'aprilway'), ('enter', 'enterway'), ('infer', 'inferway'), ('abuse',
'abuseway'), ('ivory', 'ivoryway'), ('adopt', 'adoptway'), ('image',
'imageway'), ('agree', 'agreeway'), ('alike', 'alikeway'), ('earth',

'earthway'), ('ended', 'endedway'), ('owner', 'ownerway'), ('using',
'usingway'), ('ought', 'oughtway'), ('abhor', 'abhorway'), ('altar',
'altarway'), ('often', 'oftenway'), ('atone', 'atoneway'), ('angel',
'angelway'), ('arise', 'ariseway'), ('under', 'underway'), ('abode',
'abodeway'), ('exert', 'exertway'), ('usual', 'usualway'), ('aches',
'achesway'), ('enjoy', 'enjoyway'), ('urged', 'urgedway'), ('admit',
'admitway'), ('aside', 'asideway'), ('equal', 'equalway'), ('alive',
'aliveway'), ('arose', 'aroseway'), ('eight', 'eightway'), ('opera',
'operaway'), ('ennui', 'ennuiway'), ('utter', 'utterway'), ('ankle',
'ankleway')], (2, 5): [('is', 'isway'), ('ay', 'ayway'), ('ah', 'ahway'), ('ii',
'iiway'), ('or', 'orway'), ('am', 'amway'), ('if', 'ifway'), ('an', 'anway')],
(12, 14): [('tranquillity', 'anquillitytray'), ('humiliations',
'umiliationshay'), ('contributing', 'ontributingcay'), ('significancy',
'ignificancysay'), ('wretchedness', 'etchednesswray'), ('prodigiously',
'odigiouslypray'), ('particularly', 'articularlypay'), ('carelessness',
'arelessnesscay'), ('refreshments', 'efreshmentsray'), ('satisfaction',
'atisfactionsay'), ('construction', 'onstructioncay'), ('contribution',
'ontributioncay'), ('gracefulness', 'acefulnessgray'), ('consequences',
'onsequencescay'), ('grandmothers', 'andmothersgray'), ('proportioned',
'oportionedpray'), ('perverseness', 'erversenesspay'), ('conciliation',
'onciliationcay'), ('representing', 'epresentingray'), ('watchfulness',
'atchfulnessway'), ('recollecting', 'ecollectingray'), ('hesitatingly',
'esitatinglyhay'), ('pleasantness', 'easantnessplay'), ('benefactress',
'enefactressbay'), ('distractedly', 'istractedlyday'), ('perseverance',
'erseverancepay'), ('performances', 'erformancespay'), ('disapproving',
'isapprovingday'), ('recollection', 'ecollectionray'), ('comprehended',
'omprehendedcay'), ('relinquished', 'elinquishedray'), ('conjecturing',
'onjecturingcay'), ('reservedness', 'eservednessray'), ('promontories',
'omontoriespray'), ('discontented', 'iscontentedday'), ('persecutions',
'ersecutionspay'), ('relationship', 'elationshipray'), ('triumphantly',
'iumphantlytray'), ('deliberation', 'eliberationday'), ('supplication',
'upplicationsay'), ('dishonorable', 'ishonorableday'), ('propensities',
'opensitiespray'), ('contemptuous', 'ontemptuouscay'), ('transporting',
'ansportingtray')], (13, 15): [('distinguished', 'istinguishedday'),
('resuscitation', 'esuscitationray'), ('particularity', 'articularitypay'),
('superannuated', 'uperannuatedsay'), ('disagreements', 'isagreementsday'),
('gentlemanlike', 'entlemanlikegay'), ('prepossessing', 'epossessingpray'),
('conscientious', 'onscientiouscay'), ('considerately', 'onsideratelycay'),
('felicitations', 'elicitationsfay'), ('prepossession', 'epossessionpray'),
('consternation', 'onsternationcay'), ('disproportion', 'isproportionday'),
('commiseration', 'ommiserationcay'), ('mortification', 'ortificationmay'),
('solicitations', 'olicitationssay'), ('qualification', 'alificationquay'),
('philanthropic', 'ilanthropicphay'), ('disappointing', 'isappointingday'),
('prescriptions', 'escriptionspray'), ('strengthening', 'engtheningstray')],
(13, 17): [('mulberry-tree', 'ulberrymay-eetray'), ('good-humoured', 'oodgay-
umouredhay'), ('mean-spirited', 'eanmay-iritedspay'), ('self-reproach', 'elfsay-
eproachray'), ('drawing-table', 'awingdray-abletay'), ('well-disposed', 'ellway-
isposedday'), ('proud-looking', 'oudpray-ookinglay'), ('flower-garden',

'owerflay-ardengay'), ('mansion-house', 'ansionmay-ousehay'), ('landing-place', 'andinglay-aceplay'), ('music-sellers', 'usicmay-ellerssay')], (4, 7): [('acts', 'actsway'), ('anne', 'anneway'), ('upon', 'uponway'), ('able', 'ableway'), ('inch', 'inchway'), ('ages', 'agesway'), ('envy', 'envyway'), ('idea', 'ideaway'), ('ends', 'endsway'), ('-and', '-andway'), ('eyed', 'eyedway'), ('arms', 'armsway'), ('over', 'overway'), ('easy', 'easyway'), ('ever', 'everway'), ('ours', 'oursway'), ('ears', 'earsway'), ('away', 'awayway')], (16, 20): [('pleasure-grounds', 'easureplay-oundsgray'), ('self-provocation', 'elfsay-ovocationpray')], (11, 15): [('bed-chamber', 'edbay-amberchay'), ('high-minded', 'ighhay-indedmay'), ('before-hand', 'eforebay-andhay'), ('watch-tower', 'atchway-owertay'), ('print-shops', 'intpray-opsshay'), ('card-tables', 'ardcay-ablestay'), ('post-horses', 'ostpay-orseshay'), ('good-nature', 'oodgay-aturenay'), ('dining-room', 'iningday-oomray'), ('high-church', 'ighhay-urchchay'), ('well-wisher', 'ellway-isherway'), ('bond-street', 'ondbay-eetstray'), ('carpet-work', 'arpetcay-orkway')], (2, 4): [('to', 'otay'), ('he', 'ehay'), ('no', 'onay'), ('me', 'emay')], (12, 16): [('good-hearted', 'oodgay-eartedhay'), ('well-judging', 'ellway-udgingjay'), ('drawing-room', 'awingdray-oomray'), ('common-place', 'ommoncay-aceplay'), ('self-control', 'elfsay-ontrolcay'), ('hand-writing', 'andhay-itingwray'), ('well-behaved', 'ellway-ehavedbay'), ('mantel-piece', 'antelmay-iecepay')], (10, 15): [('ill-humour', 'illway-umourhay'), ('ill-judged', 'illway-udgedjay'), ('ill-suited', 'illway-uitedsay')], (1, 3): [('c', 'cay')], (8, 13): [('ill-wind', 'illway-indway'), ('ill-will', 'illway-illway'), ('over-set', 'overway-etsay')], (17, 19): [('disqualifications', 'isqualificationsday')], (9, 16): [('by-and-by', 'ybay-andway-ybay')], (3, 6): [('aye', 'ayeway'), ('use', 'useway'), ('owe', 'oweway'), ('our', 'ourway'), ('old', 'oldway'), ('ear', 'earway'), ('its', 'itsway'), ('ill', 'illway')], (12, 17): [('ill-breeding', 'illway-eedingbray'), ('re-assurance', 'eray-assuranceway'), ('over-powered', 'overway-oweredpay'), ('ill-disposed', 'illway-isposedday')], (11, 16): [('ill-natured', 'illway-aturednay')], (14, 16): [('reconciliation', 'econciliationray'), ('recommendation', 'ecommendationray'), ('reasonableness', 'easonablenessray'), ('disinclination', 'isinclinationday'), ('respectability', 'espectabilityray'), ('circumspection', 'ircumspectioncay'), ('considerations', 'onsiderationscay'), ('qualifications', 'alificationsquay'), ('congratulating', 'ongratulatingcay')], (15, 17): [('dissatisfaction', 'issatisfactionday'), ('disrespectfully', 'isrespectfullyday'), ('conscientiously', 'onscientiouslycay'), ('misconstruction', 'isconstructionmay')], (15, 18): [('instantaneously', 'instantaneouslyway')], (15, 19): [('shooting-jacket', 'ootingshay-acketjay')], (14, 21): [('sisters-in-law', 'isterssay-inway-awlay')], (17, 21): [('self-condemnation', 'elfsay-ondemnationcay')], (13, 20): [('sister-in-law', 'istersay-inway-awlay')], (9, 14): [('ill-timed', 'illway-imedtay')], (13, 18): [('ill-treatment', 'illway-eatmenttray'), ('old-fashioned', 'oldway-ashionedfay'), ('self-interest', 'elfsay-interestway')], (16, 22): [('good-for-nothing', 'oodgay-orfay-othingnay')], (9, 13): [('dove-cote', 'oveday-otecay'), ('tea-table', 'eatay-abletay'), ('work-bags', 'orkway-agsbay')], (1, 4): [('i', 'iway')], (16, 19): [('incomprehensible', 'incomprehensibleway')], (8, 12): [('to-night', 'otay-ightnay'), ('land-tax', 'andlay-axtay')], (11, 18): [('tete-a-tete', 'etetay-away-etetay')], (5, 9): [('to-do', 'otay-oday')]})

```
Moved models to GPU!
Epoch:   0 | Flops (T): 0.002374394016, Tokens (M) 0.052678, Train Loss 3.102,
Val Loss 2.931 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   1 | Flops (T): 0.004748788032, Tokens (M) 0.105356, Train Loss 2.912,
Val Loss 2.903 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   2 | Flops (T): 0.007123182048, Tokens (M) 0.158034, Train Loss 2.899,
Val Loss 2.897 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   3 | Flops (T): 0.009497576064, Tokens (M) 0.210712, Train Loss 2.895,
Val Loss 2.894 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   4 | Flops (T): 0.01187197008, Tokens (M) 0.26339, Train Loss 2.892, Val
Loss 2.891 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   5 | Flops (T): 0.014246364096, Tokens (M) 0.316068, Train Loss 2.887,
Val Loss 2.882 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   6 | Flops (T): 0.016620758112, Tokens (M) 0.368746, Train Loss 2.878,
Val Loss 2.874 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   7 | Flops (T): 0.018995152128, Tokens (M) 0.421424, Train Loss 2.871,
Val Loss 2.864 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   8 | Flops (T): 0.021369546144, Tokens (M) 0.474102, Train Loss 2.863,
Val Loss 2.857 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   9 | Flops (T): 0.02374394016, Tokens (M) 0.52678, Train Loss 2.856, Val
Loss 2.851 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  10 | Flops (T): 0.026118334176, Tokens (M) 0.579458, Train Loss 2.849,
Val Loss 2.844 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  11 | Flops (T): 0.028492728192, Tokens (M) 0.632136, Train Loss 2.841,
Val Loss 2.838 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  12 | Flops (T): 0.030867122208, Tokens (M) 0.684814, Train Loss 2.832,
Val Loss 2.829 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  13 | Flops (T): 0.033241516224, Tokens (M) 0.737492, Train Loss 2.823,
Val Loss 2.819 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  14 | Flops (T): 0.03561591024, Tokens (M) 0.79017, Train Loss 2.815, Val
Loss 2.809 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  15 | Flops (T): 0.037990304256, Tokens (M) 0.842848, Train Loss 2.808,
Val Loss 2.801 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  16 | Flops (T): 0.040364698272, Tokens (M) 0.895526, Train Loss 2.801,
Val Loss 2.793 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  17 | Flops (T): 0.042739092288, Tokens (M) 0.948204, Train Loss 2.794,
Val Loss 2.786 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  18 | Flops (T): 0.045113486304, Tokens (M) 1.000882, Train Loss 2.787,
Val Loss 2.780 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  19 | Flops (T): 0.04748788032, Tokens (M) 1.05356, Train Loss 2.780, Val
Loss 2.774 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  20 | Flops (T): 0.049862274336, Tokens (M) 1.106238, Train Loss 2.773,
Val Loss 2.768 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  21 | Flops (T): 0.052236668352, Tokens (M) 1.158916, Train Loss 2.766,
Val Loss 2.763 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  22 | Flops (T): 0.054611062368, Tokens (M) 1.211594, Train Loss 2.760,
Val Loss 2.759 | Gen: aaaaaaaaaayyyyyyyyy
Epoch:  23 | Flops (T): 0.056985456384, Tokens (M) 1.264272, Train Loss 2.753,
```

```
Val Loss 2.754 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  24 | Flops (T): 0.0593598504, Tokens (M) 1.31695, Train Loss 2.746, Val
Loss 2.750 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  25 | Flops (T): 0.061734244416, Tokens (M) 1.369628, Train Loss 2.739,
Val Loss 2.743 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  26 | Flops (T): 0.064108638432, Tokens (M) 1.422306, Train Loss 2.731,
Val Loss 2.735 | Gen: oooooooooooooooooooo
Epoch:  27 | Flops (T): 0.066483032448, Tokens (M) 1.474984, Train Loss 2.720,
Val Loss 2.723 | Gen: oooooooooooooooooooo
Epoch:  28 | Flops (T): 0.068857426464, Tokens (M) 1.527662, Train Loss 2.704,
Val Loss 2.701 | Gen: aaoooooooooooaaaaaaa
Epoch:  29 | Flops (T): 0.07123182048, Tokens (M) 1.58034, Train Loss 2.668, Val
Loss 2.649 | Gen: oooooooooaaaaaaaaaaa
Epoch:  30 | Flops (T): 0.073606214496, Tokens (M) 1.633018, Train Loss 2.588,
Val Loss 2.568 | Gen: oooooooooooooaaaaaa
Epoch:  31 | Flops (T): 0.075980608512, Tokens (M) 1.685696, Train Loss 2.526,
Val Loss 2.520 | Gen: oooooooooooooaaaaaa
Epoch:  32 | Flops (T): 0.078355002528, Tokens (M) 1.738374, Train Loss 2.459,
Val Loss 2.480 | Gen: oooooooooooooa
Epoch:  33 | Flops (T): 0.080729396544, Tokens (M) 1.791052, Train Loss 2.415,
Val Loss 2.453 | Gen: oooooooooooooy
Epoch:  34 | Flops (T): 0.08310379056, Tokens (M) 1.84373, Train Loss 2.292, Val
Loss 2.271 | Gen: oooooooooooooooooooo
Epoch:  35 | Flops (T): 0.085478184576, Tokens (M) 1.896408, Train Loss 2.233,
Val Loss 2.178 | Gen: oooooooooooooaaaaaaa
Epoch:  36 | Flops (T): 0.087852578592, Tokens (M) 1.949086, Train Loss 2.167,
Val Loss 2.153 | Gen: oooooooooooooaaaaaa
Epoch:  37 | Flops (T): 0.090226972608, Tokens (M) 2.001764, Train Loss 2.147,
Val Loss 2.128 | Gen: oooooooooooooooooooa
Epoch:  38 | Flops (T): 0.092601366624, Tokens (M) 2.054442, Train Loss 2.134,
Val Loss 2.120 | Gen: oooooooooooooooooooa
Epoch:  39 | Flops (T): 0.09497576064, Tokens (M) 2.10712, Train Loss 2.124, Val
Loss 2.146 | Gen: oooooooooooooooooooo
Epoch:  40 | Flops (T): 0.097350154656, Tokens (M) 2.159798, Train Loss 2.126,
Val Loss 2.127 | Gen: oooooooooooooooooooo
Epoch:  41 | Flops (T): 0.099724548672, Tokens (M) 2.212476, Train Loss 2.106,
Val Loss 2.097 | Gen: oooooooooooooooooooo
Epoch:  42 | Flops (T): 0.102098942688, Tokens (M) 2.265154, Train Loss 2.094,
Val Loss 2.092 | Gen: oooooooooooooooooooo
Epoch:  43 | Flops (T): 0.104473336704, Tokens (M) 2.317832, Train Loss 2.083,
Val Loss 2.084 | Gen: oooooooooooooooooooo
Epoch:  44 | Flops (T): 0.10684773072, Tokens (M) 2.37051, Train Loss 2.075, Val
Loss 2.081 | Gen: oooooooooooooooooooo
Epoch:  45 | Flops (T): 0.109222124736, Tokens (M) 2.423188, Train Loss 2.068,
Val Loss 2.074 | Gen: oooooooooooooooooooo
Epoch:  46 | Flops (T): 0.111596518752, Tokens (M) 2.475866, Train Loss 2.063,
Val Loss 2.068 | Gen: oooooooooooooooooooo
Epoch:  47 | Flops (T): 0.113970912768, Tokens (M) 2.528544, Train Loss 2.061,
```

```
Val Loss 2.062 | Gen: oooooooooooooooooooa
Epoch:  48 | Flops (T): 0.116345306784, Tokens (M) 2.581222, Train Loss 2.054,
Val Loss 2.056 | Gen: oooooooooooooooooooa
Epoch:  49 | Flops (T): 0.1187197008, Tokens (M) 2.6339, Train Loss 2.045, Val
Loss 2.050 | Gen: oooooooooooooooooooa
Epoch:  50 | Flops (T): 0.121094094816, Tokens (M) 2.686578, Train Loss 2.040,
Val Loss 2.047 | Gen: oooooooooooooooooooa
Epoch:  51 | Flops (T): 0.123468488832, Tokens (M) 2.739256, Train Loss 2.036,
Val Loss 2.044 | Gen: oooooooooooooooooooa
Epoch:  52 | Flops (T): 0.125842882848, Tokens (M) 2.791934, Train Loss 2.034,
Val Loss 2.065 | Gen: oooooooooooooooooooo
Epoch:  53 | Flops (T): 0.128217276864, Tokens (M) 2.844612, Train Loss 2.027,
Val Loss 2.029 | Gen: oooooooooooooooooooo
Epoch:  54 | Flops (T): 0.13059167088, Tokens (M) 2.89729, Train Loss 2.022, Val
Loss 2.049 | Gen: oooooooooooooooooooo
Epoch:  55 | Flops (T): 0.132966064896, Tokens (M) 2.949968, Train Loss 2.018,
Val Loss 2.052 | Gen: oooooooooooooooooooa
Epoch:  56 | Flops (T): 0.135340458912, Tokens (M) 3.002646, Train Loss 2.013,
Val Loss 2.050 | Gen: oooooooooooooooooooa
Epoch:  57 | Flops (T): 0.137714852928, Tokens (M) 3.055324, Train Loss 2.011,
Val Loss 2.050 | Gen: oooooooooooooooooooa
Epoch:  58 | Flops (T): 0.140089246944, Tokens (M) 3.108002, Train Loss 2.011,
Val Loss 2.019 | Gen: oooooooooooooooooooa
Epoch:  59 | Flops (T): 0.14246364096, Tokens (M) 3.16068, Train Loss 2.008, Val
Loss 2.048 | Gen: oooooooooooooooooooa
Epoch:  60 | Flops (T): 0.144838034976, Tokens (M) 3.213358, Train Loss 2.004,
Val Loss 2.058 | Gen: ooooooooooooowayyyy
Epoch:  61 | Flops (T): 0.147212428992, Tokens (M) 3.266036, Train Loss 2.006,
Val Loss 2.051 | Gen: oooooooooooooooooooa
Epoch:  62 | Flops (T): 0.149586823008, Tokens (M) 3.318714, Train Loss 2.000,
Val Loss 2.048 | Gen: oooooooooooooooooooa
Epoch:  63 | Flops (T): 0.151961217024, Tokens (M) 3.371392, Train Loss 1.998,
Val Loss 2.048 | Gen: oooooooooooooooooooo
Epoch:  64 | Flops (T): 0.15433561104, Tokens (M) 3.42407, Train Loss 1.994, Val
Loss 2.045 | Gen: oooooooooooooooooooo
Epoch:  65 | Flops (T): 0.156710005056, Tokens (M) 3.476748, Train Loss 1.991,
Val Loss 2.036 | Gen: oooooooooooooooooooo
Epoch:  66 | Flops (T): 0.159084399072, Tokens (M) 3.529426, Train Loss 1.987,
Val Loss 2.037 | Gen: oooooooooooooooooooo
Epoch:  67 | Flops (T): 0.161458793088, Tokens (M) 3.582104, Train Loss 1.985,
Val Loss 2.035 | Gen: oooooooooooooooooooo
Epoch:  68 | Flops (T): 0.163833187104, Tokens (M) 3.634782, Train Loss 1.982,
Val Loss 2.034 | Gen: oooooooooooooooooooo
Epoch:  69 | Flops (T): 0.16620758112, Tokens (M) 3.68746, Train Loss 1.980, Val
Loss 2.033 | Gen: oooooooooooooooooooo
Epoch:  70 | Flops (T): 0.168581975136, Tokens (M) 3.740138, Train Loss 1.978,
Val Loss 2.031 | Gen: oooooooooooooooooooo
Epoch:  71 | Flops (T): 0.170956369152, Tokens (M) 3.792816, Train Loss 1.977,
```

```
Val Loss 2.039 | Gen: oooooooooooooooooooo
Epoch:  72 | Flops (T): 0.173330763168, Tokens (M) 3.845494, Train Loss 1.975,
Val Loss 2.032 | Gen: oooooooooooooooooooo
Epoch:  73 | Flops (T): 0.175705157184, Tokens (M) 3.898172, Train Loss 1.970,
Val Loss 2.029 | Gen: oooooooooooooooooooo
Epoch:  74 | Flops (T): 0.1780795512, Tokens (M) 3.95085, Train Loss 1.968, Val
Loss 2.028 | Gen: oooooooooooooooooooo
Epoch:  75 | Flops (T): 0.180453945216, Tokens (M) 4.003528, Train Loss 1.966,
Val Loss 2.027 | Gen: oooooooooooooooooooo
Epoch:  76 | Flops (T): 0.182828339232, Tokens (M) 4.056206, Train Loss 1.965,
Val Loss 2.026 | Gen: oooooooooooooooooooo
Epoch:  77 | Flops (T): 0.185202733248, Tokens (M) 4.108884, Train Loss 1.963,
Val Loss 2.025 | Gen: oooooooooooooooooooo
Epoch:  78 | Flops (T): 0.187577127264, Tokens (M) 4.161562, Train Loss 1.961,
Val Loss 2.025 | Gen: ooooooooooooooooooow
Epoch:  79 | Flops (T): 0.18995152128, Tokens (M) 4.21424, Train Loss 1.961, Val
Loss 2.025 | Gen: ooooooooooooooooooow
Epoch:  80 | Flops (T): 0.192325915296, Tokens (M) 4.266918, Train Loss 1.958,
Val Loss 2.024 | Gen: ooooooooooooooooooow
Epoch:  81 | Flops (T): 0.194700309312, Tokens (M) 4.319596, Train Loss 1.957,
Val Loss 2.023 | Gen: ooooooooooooooooooow
Epoch:  82 | Flops (T): 0.197074703328, Tokens (M) 4.372274, Train Loss 1.955,
Val Loss 2.022 | Gen: ooooooooooooooooooow
Epoch:  83 | Flops (T): 0.199449097344, Tokens (M) 4.424952, Train Loss 1.954,
Val Loss 2.020 | Gen: oooooooooooooooooooo
Epoch:  84 | Flops (T): 0.20182349136, Tokens (M) 4.47763, Train Loss 1.955, Val
Loss 2.099 | Gen: oooooooooooooowa
Epoch:  85 | Flops (T): 0.204197885376, Tokens (M) 4.530308, Train Loss 1.984,
Val Loss 2.051 | Gen: oooooooooooooooa
Epoch:  86 | Flops (T): 0.206572279392, Tokens (M) 4.582986, Train Loss 1.959,
Val Loss 2.026 | Gen: ooooooooooooooooooow
Epoch:  87 | Flops (T): 0.208946673408, Tokens (M) 4.635664, Train Loss 1.954,
Val Loss 2.034 | Gen: ooooooooooooooooooow
Epoch:  88 | Flops (T): 0.211321067424, Tokens (M) 4.688342, Train Loss 1.954,
Val Loss 2.026 | Gen: oooooooooooooooooooo
Epoch:  89 | Flops (T): 0.21369546144, Tokens (M) 4.74102, Train Loss 1.951, Val
Loss 2.022 | Gen: oooooooooooooooooooo
Epoch:  90 | Flops (T): 0.216069855456, Tokens (M) 4.793698, Train Loss 1.949,
Val Loss 2.019 | Gen: oooooooooooooooooooo
Epoch:  91 | Flops (T): 0.218444249472, Tokens (M) 4.846376, Train Loss 1.947,
Val Loss 2.018 | Gen: oooooooooooooooooooo
Epoch:  92 | Flops (T): 0.220818643488, Tokens (M) 4.899054, Train Loss 1.946,
Val Loss 2.017 | Gen: oooooooooooooooooooo
Epoch:  93 | Flops (T): 0.223193037504, Tokens (M) 4.951732, Train Loss 1.945,
Val Loss 2.018 | Gen: oooooooooooooooooooo
Epoch:  94 | Flops (T): 0.22556743152, Tokens (M) 5.00441, Train Loss 1.944, Val
Loss 2.016 | Gen: oooooooooooooooooooo
Epoch:  95 | Flops (T): 0.227941825536, Tokens (M) 5.057088, Train Loss 1.943,
```

```
Val Loss 2.016 | Gen: ooooooooooooooooooo
Epoch:  96 | Flops (T): 0.230316219552, Tokens (M) 5.109766, Train Loss 1.942,
Val Loss 2.017 | Gen: ooooooooooooooooooo
Epoch:  97 | Flops (T): 0.232690613568, Tokens (M) 5.162444, Train Loss 1.941,
Val Loss 2.017 | Gen: ooooooooooooooooooo
Epoch:  98 | Flops (T): 0.235065007584, Tokens (M) 5.215122, Train Loss 1.940,
Val Loss 2.017 | Gen: ooooooooooooooooooo
Epoch:  99 | Flops (T): 0.2374394016, Tokens (M) 5.2678, Train Loss 1.939, Val
Loss 2.017 | Gen: ooooooooooooooooooo
Obtained lowest validation loss of: 2.0158522628074467
Avg tokens per example (Seq Len): 20.59343236903831, Avg flops per example:
928222.8365910868
```

[ ]:
```python
TEST_SENTENCE = "floor"
trans32_args_s = AttrDict()
args_dict_32 = {
    "data_file_name": "pig_latin_small", #pig_latin_small debug
    "cuda": use_cuda,
    "nepochs": 100,
    "checkpoint_dir": "checkpoints",
    "learning_rate": 5e-4,
    "early_stopping_patience": 100,
    "lr_decay": 0.99,
    "batch_size": 64,
    "hidden_size": 32,
    "num_transformer_layers": 4,
    "loss_seq_span": "target_only",
    "num_layers": 4,
    "vocab_size": 30
}
trans32_args_s.update(args_dict_32)
print_opts(trans32_args_s)
trans32_decoder_s, trans32_losses_s, idx_dict = train(trans32_args_s)
```

```
================================================================================
                                       Opts
--------------------------------------------------------------------------------
                         data_file_name: pig_latin_small
                                   cuda: 1
                                 nepochs: 100
                         checkpoint_dir: checkpoints
                          learning_rate: 0.0005
                early_stopping_patience: 100
                                lr_decay: 0.99
                              batch_size: 64
                             hidden_size: 32
                 num_transformer_layers: 4
                          loss_seq_span: target_only
```

```
                              num_layers: 4
                              vocab_size: 30
================================================================================
================================================================================
                                  Data Stats
--------------------------------------------------------------------------------
('irresistible', 'irresistibleway')
('repentance', 'epentanceray')
('pretty', 'ettypray')
('parcels', 'arcelspay')
('survey', 'urveysay')
Num unique word pairs: 3198
Vocabulary: dict_keys(['-', 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j',
'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z',
'SOS', 'EOS', 'EOP'])
Vocab size: 30
================================================================================
defaultdict(<class 'list'>, {(12, 15): [('irresistible', 'irresistibleway'),
('advantageous', 'advantageousway'), ('impartiality', 'impartialityway'),
('interruption', 'interruptionway'), ('anticipation', 'anticipationway'),
('impenetrable', 'impenetrableway'), ('excellencies', 'excellenciesway'),
('intermission', 'intermissionway'), ('interspersed', 'interspersedway'),
('unaffectedly', 'unaffectedlyway'), ('insinuations', 'insinuationsway'),
('unpardonable', 'unpardonableway'), ('arrangements', 'arrangementsway'),
('unexpectedly', 'unexpectedlyway'), ('introduction', 'introductionway'),
('occasionally', 'occasionallyway'), ('establishing', 'establishingway'),
('unanswerable', 'unanswerableway'), ('independence', 'independenceway'),
('inclinations', 'inclinationsway'), ('instructions', 'instructionsway'),
('undoubtingly', 'undoubtinglyway'), ('unfathomable', 'unfathomableway'),
('impoverished', 'impoverishedway'), ('outstretched', 'outstretchedway'),
('unrestrained', 'unrestrainedway'), ('ungraciously', 'ungraciouslyway'),
('imperfection', 'imperfectionway'), ('unreasonably', 'unreasonablyway'),
('interference', 'interferenceway'), ('inconvenient', 'inconvenientway'),
('enthusiastic', 'enthusiasticway'), ('extinguished', 'extinguishedway'),
('entanglement', 'entanglementway'), ('affectionate', 'affectionateway')], (10,
12): [('repentance', 'epentanceray'), ('capability', 'apabilitycay'),
('beneficial', 'eneficialbay'), ('retrenched', 'etrenchedray'), ('devonshire',
'evonshireday'), ('palanquins', 'alanquinspay'), ('contribute', 'ontributecay'),
('minuteness', 'inutenessmay'), ('dissimilar', 'issimilarday'), ('flattering',
'atteringflay'), ('confounded', 'onfoundedcay'), ('compromise', 'ompromisecay'),
('contrasted', 'ontrastedcay'), ('precaution', 'ecautionpray'), ('fastidious',
'astidiousfay'), ('blackguard', 'ackguardblay'), ('heroically', 'eroicallyhay'),
('lengthened', 'engthenedlay'), ('pocketbook', 'ocketbookpay'), ('companions',
'ompanionscay'), ('displeased', 'ispleasedday'), ('pianoforte', 'ianofortepay'),
('succession', 'uccessionsay'), ('composedly', 'omposedlycay'), ('pronounced',
'onouncedpray'), ('honourable', 'onourablehay'), ('solicitude', 'olicitudesay'),
('justifying', 'ustifyingjay'), ('collecting', 'ollectingcay'), ('headstrong',
'eadstronghay'), ('forbidding', 'orbiddingfay'), ('successful', 'uccessfulsay'),
```

('confidence', 'onfidencecay'), ('prescribed', 'escribedpray'), ('scrutinies', 'utiniesscray'), ('melancholy', 'elancholymay'), ('handsomest', 'andsomesthay'), ('prescience', 'esciencepray'), ('conclusion', 'onclusioncay'), ('remarkably', 'emarkablyray'), ('gratefully', 'atefullygray'), ('michaelmas', 'ichaelmasmay'), ('containing', 'ontainingcay'), ('discussion', 'iscussionday'), ('resentment', 'esentmentray'), ('restraints', 'estraintsray'), ('tablecloth', 'ableclothtay'), ('projecting', 'ojectingpray'), ('frightened', 'ightenedfray'), ('pardonable', 'ardonablepay'), ('variations', 'ariationsvay'), ('sensations', 'ensationssay'), ('rheumatism', 'eumatismrhay'), ('dispirited', 'ispiritedday'), ('politeness', 'olitenesspay'), ('suggesting', 'uggestingsay'), ('willoughby', 'illoughbyway'), ('reasonable', 'easonableray'), ('discerning', 'iscerningday'), ('previously', 'eviouslypray'), ('passionate', 'assionatepay'), ('prophecies', 'opheciespray'), ('possessing', 'ossessingpay'), ('perfection', 'erfectionpay'), ('heightened', 'eightenedhay'), ('generously', 'enerouslygay'), ('faithfully', 'aithfullyfay'), ('reconciled', 'econciledray'), ('distressed', 'istressedday'), ('respective', 'espectiveray'), ('liberality', 'iberalitylay'), ('disengaged', 'isengagedday'), ('honourably', 'onourablyhay'), ('persuading', 'ersuadingpay'), ('cultivated', 'ultivatedcay'), ('projection', 'ojectionpray'), ('disapprove', 'isapproveday'), ('conciliate', 'onciliatecay'), ('satisfying', 'atisfyingsay'), ('dispatches', 'ispatchesday'), ('repeatedly', 'epeatedlyray'), ('suspicions', 'uspicionssay'), ('perplexity', 'erplexitypay'), ('strengthen', 'engthenstray'), ('dependence', 'ependenceday'), ('dissipated', 'issipatedday'), ('conformity', 'onformitycay'), ('hysterical', 'ystericalhay'), ('throughout', 'oughoutthray'), ('sacrificed', 'acrificedsay'), ('dimensions', 'imensionsday'), ('flatteries', 'atteriesflay'), ('completely', 'ompletelycay'), ('reproaches', 'eproachesray'), ('contriving', 'ontrivingcay'), ('consequent', 'onsequentcay'), ('liveliness', 'ivelinesslay'), ('resistance', 'esistanceray'), ('caricature', 'aricaturecay'), ('stimulated', 'imulatedstay'), ('brightened', 'ightenedbray'), ('prevailing', 'evailingpray'), ('disclosure', 'isclosureday'), ('bequeathed', 'equeathedbay'), ('disappoint', 'isappointday'), ('complexion', 'omplexioncay'), ('dismounted', 'ismountedday'), ('remembered', 'ememberedray'), ('delineated', 'elineatedday'), ('subsequent', 'ubsequentsay'), ('persevered', 'erseveredpay'), ('doubtingly', 'oubtinglyday'), ('characters', 'aracterschay'), ('neighbours', 'eighboursnay'), ('witticisms', 'itticismsway'), ('materially', 'ateriallymay'), ('playthings', 'aythingsplay'), ('suspecting', 'uspectingsay'), ('conditions', 'onditionscay'), ('discovered', 'iscoveredday'), ('deficiency', 'eficiencyday'), ('regularity', 'egularityray'), ('subjection', 'ubjectionsay'), ('brightness', 'ightnessbray'), ('pleasantly', 'easantlyplay'), ('delicacies', 'elicaciesday'), ('preference', 'eferencepray'), ('difference', 'ifferenceday'), ('compassion', 'ompassioncay'), ('competence', 'ompetencecay'), ('generosity', 'enerositygay'), ('retirement', 'etirementray'), ('strikingly', 'ikinglystray'), ('subsisting', 'ubsistingsay'), ('conscience', 'onsciencecay'), ('diabolical', 'iabolicalday'), ('temptation', 'emptationtay'), ('formidable', 'ormidablefay'), ('reconsider', 'econsiderray'), ('prejudiced', 'ejudicedpray'), ('comprehend', 'omprehendcay'), ('compliment', 'omplimentcay'), ('sufferings', 'ufferingssay'), ('concluding', 'oncludingcay'), ('compliance', 'ompliancecay'), ('mosquitoes', 'osquitoesmay'), ('thoughtful', 'oughtfulthay'), ('bitterness', 'itternessbay'), ('mentioning', 'entioningmay'), ('toleration', 'olerationtay'), ('profession', 'ofessionpray'), ('discompose', 'iscomposeday'),

```
('waistcoats', 'aistcoatsway'), ('reflecting', 'eflectingray'), ('positively',
'ositivelypay'), ('diminished', 'iminishedday')], (6, 8): [('pretty',
'ettypray'), ('survey', 'urveysay'), ('demure', 'emureday'), ('secure',
'ecuresay'), ('sobbed', 'obbedsay'), ('moment', 'omentmay'), ('guilty',
'uiltygay'), ('combat', 'ombatcay'), ('modern', 'odernmay'), ('father',
'atherfay'), ('horses', 'orseshay'), ('closer', 'oserclay'), ('nicety',
'icetynay'), ('shaken', 'akenshay'), ('centre', 'entrecay'), ('powers',
'owerspay'), ('simple', 'implesay'), ('beings', 'eingsbay'), ('recent',
'ecentray'), ('sketch', 'etchskay'), ('loving', 'ovinglay'), ('dagger',
'aggerday'), ('hearts', 'eartshay'), ('copied', 'opiedcay'), ('gazing',
'azinggay'), ('sooner', 'oonersay'), ('disarm', 'isarmday'), ('suited',
'uitedsay'), ('doubts', 'oubtsday'), ('lifted', 'iftedlay'), ('vacant',
'acantvay'), ('palmer', 'almerpay'), ('grudge', 'udgegray'), ('folded',
'oldedfay'), ('remove', 'emoveray'), ('borrow', 'orrowbay'), ('scheme',
'emeschay'), ('solemn', 'olemnsay'), ('denote', 'enoteday'), ('bustle',
'ustlebay'), ('turned', 'urnedtay'), ('mutton', 'uttonmay'), ('tender',
'endertay'), ('hearer', 'earerhay'), ('begins', 'eginsbay'), ('taught',
'aughttay'), ('honour', 'onourhay'), ('robert', 'obertray'), ('gibson',
'ibsongay'), ('virtue', 'irtuevay'), ('vanish', 'anishvay'), ('writes',
'iteswray'), ('silent', 'ilentsay'), ('singly', 'inglysay'), ('jargon',
'argonjay'), ('grieve', 'ievegray'), ('design', 'esignday'), ('linger',
'ingerlay'), ('bathed', 'athedbay'), ('pardon', 'ardonpay'), ('rumour',
'umourray'), ('dances', 'ancesday'), ('remark', 'emarkray'), ('worded',
'ordedway'), ('honest', 'onesthay'), ('chosen', 'osenchay'), ('boyish',
'oyishbay'), ('stress', 'essstray'), ('caught', 'aughtcay'), ('potent',
'otentpay'), ('during', 'uringday'), ('stocks', 'ocksstay'), ('relics',
'elicsray'), ('tongue', 'onguetay'), ('street', 'eetstray'), ('whiled',
'iledwhay'), ('manner', 'annermay'), ('truths', 'uthstray'), ('travel',
'aveltray'), ('couple', 'ouplecay'), ('gaiety', 'aietygay'), ('giving',
'ivinggay'), ('county', 'ountycay'), ('cramps', 'ampscray'), ('wealth',
'ealthway'), ('divine', 'ivineday'), ('behold', 'eholdbay'), ('feared',
'earedfay'), ('sparks', 'arksspay'), ('finger', 'ingerfay'), ('talent',
'alenttay'), ('demand', 'emandday'), ('coffee', 'offeecay'), ('merest',
'erestmay'), ('detail', 'etailday'), ('casino', 'asinocay'), ('choose',
'oosechay'), ('warmed', 'armedway'), ('relate', 'elateray'), ('sashes',
'ashessay'), ('chance', 'ancechay'), ('rushed', 'ushedray'), ('health',
'ealthhay'), ('preyed', 'eyedpray'), ('mutual', 'utualmay'), ('months',
'onthsmay'), ('raises', 'aisesray'), ('things', 'ingsthay'), ('master',
'astermay'), ('thanks', 'anksthay'), ('cheeks', 'eekschay'), ('hazard',
'azardhay'), ('slight', 'ightslay'), ('guests', 'uestsgay'), ('halves',
'alveshay'), ('lessen', 'essenlay'), ('notice', 'oticenay'), ('higher',
'igherhay'), ('truest', 'uesttray'), ('porter', 'orterpay'), ('fickle',
'icklefay'), ('behave', 'ehavebay'), ('credit', 'editcray'), ('number',
'umbernay'), ('mother', 'othermay'), ('tricks', 'ickstray'), ('traced',
'acedtray'), ('subdue', 'ubduesay'), ('ruined', 'uinedray'), ('famous',
'amousfay'), ('danced', 'ancedday'), ('clever', 'everclay'), ('drives',
'ivesdray'), ('resist', 'esistray'), ('forget', 'orgetfay'), ('stable',
'ablestay'), ('bigger', 'iggerbay'), ('betray', 'etraybay'), ('senses',
```

'ensessay'), ('smiled', 'iledsmay'), ('hating', 'atinghay'), ('bolder',
'olderbay'), ('banish', 'anishbay'), ('remain', 'emainray'), ('period',
'eriodpay'), ('gossip', 'ossipgay'), ('basket', 'asketbay'), ('genius',
'eniusgay'), ('seized', 'eizedsay'), ('widely', 'idelyway'), ('pursue',
'ursuepay'), ('favour', 'avourfay'), ('sighed', 'ighedsay'), ('plants',
'antsplay'), ('proved', 'ovedpray'), ('season', 'easonsay'), ('circle',
'irclecay'), ('taking', 'akingtay'), ('screen', 'eenscray'), ('talker',
'alkertay'), ('teazed', 'eazedtay'), ('suffer', 'uffersay'), ('curate',
'uratecay'), ('traits', 'aitstray'), ('defect', 'efectday'), ('called',
'alledcay'), ('dining', 'iningday'), ('direct', 'irectday'), ('dinner',
'innerday'), ('joking', 'okingjay'), ('whiter', 'iterwhay'), ('limits',
'imitslay'), ('papers', 'aperspay'), ('public', 'ublicpay'), ('losing',
'osinglay'), ('handed', 'andedhay'), ('wisest', 'isestway'), ('delays',
'elaysday'), ('leaves', 'eaveslay'), ('points', 'ointspay'), ('hither',
'itherhay'), ('horrid', 'orridhay'), ('middle', 'iddlemay'), ('vastly',
'astlyvay'), ('luxury', 'uxurylay'), ('failed', 'ailedfay'), ('lodges',
'odgeslay'), ('female', 'emalefay'), ('walked', 'alkedway'), ('nipped',
'ippednay'), ('nephew', 'ephewnay'), ('settle', 'ettlesay'), ('kinder',
'inderkay'), ('bowing', 'owingbay'), ('smooth', 'oothsmay'), ('roared',
'oaredray'), ('wiping', 'ipingway'), ('jewels', 'ewelsjay'), ('breath',
'eathbray'), ('steady', 'eadystay'), ('breach', 'eachbray'), ('polite',
'olitepay'), ('colder', 'oldercay'), ('though', 'oughthay'), ('poured',
'ouredpay'), ('paused', 'ausedpay'), ('nought', 'oughtnay'), ('shrewd',
'ewdshray'), ('fourth', 'ourthfay'), ('parish', 'arishpay'), ('traded',
'adedtray'), ('glance', 'anceglay'), ('lesser', 'esserlay'), ('curacy',
'uracycay'), ('labour', 'abourlay'), ('became', 'ecamebay'), ('decide',
'ecideday'), ('kisses', 'isseskay'), ('styled', 'yledstay'), ('sprung',
'ungspray'), ('priory', 'iorypray'), ('derive', 'eriveday'), ('school',
'oolschay'), ('liking', 'ikinglay'), ('dawned', 'awnedday'), ('weaken',
'eakenway'), ('winter', 'interway'), ('rivals', 'ivalsray'), ('sallow',
'allowsay'), ('defend', 'efendday'), ('seated', 'eatedsay'), ('penury',
'enurypay'), ('repair', 'epairray'), ('chaise', 'aisechay'), ('church',
'urchchay'), ('beyond', 'eyondbay'), ('wisdom', 'isdomway'), ('retain',
'etainray'), ('toilet', 'oilettay'), ('talked', 'alkedtay'), ('dearly',
'earlyday'), ('reward', 'ewardray'), ('safely', 'afelysay'), ('making',
'akingmay'), ('debate', 'ebateday'), ('future', 'uturefay'), ('broken',
'okenbray'), ('begged', 'eggedbay'), ('thirty', 'irtythay'), ('muslin',
'uslinmay'), ('marked', 'arkedmay'), ('hinder', 'inderhay'), ('beheld',
'eheldbay'), ('maxims', 'aximsmay'), ('theirs', 'eirsthay'), ('corner',
'ornercay'), ('piqued', 'iquedpay'), ('rested', 'estedray'), ('needed',
'eedednay'), ('fitted', 'ittedfay'), ('candid', 'andidcay'), ('laughs',
'aughslay'), ('window', 'indowway'), ('tastes', 'astestay'), ('spread',
'eadspray'), ('hollow', 'ollowhay'), ('sorrow', 'orrowsay'), ('partly',
'artlypay'), ('served', 'ervedsay'), ('flowed', 'owedflay'), ('remedy',
'emedyray'), ('trying', 'yingtray'), ('latest', 'atestlay'), ('manage',
'anagemay'), ('gained', 'ainedgay'), ('plenty', 'entyplay'), ('garden',
'ardengay'), ('frosts', 'ostsfray'), ('thirds', 'irdsthay'), ('handle',
'andlehay'), ('verbal', 'erbalvay'), ('sheath', 'eathshay'), ('covert',

```
'overtcay'), ('darker', 'arkerday'), ('harris', 'arrishay'), ('houses',
'ouseshay'), ('nature', 'aturenay'), ('thomas', 'omasthay'), ('submit',
'ubmitsay'), ('danger', 'angerday'), ('lawyer', 'awyerlay'), ('matter',
'attermay'), ('rugged', 'uggedray'), ('highly', 'ighlyhay'), ('whence',
'encewhay'), ('martha', 'arthamay'), ('hearty', 'eartyhay'), ('recall',
'ecallray'), ('raised', 'aisedray'), ('follow', 'ollowfay'), ('nearer',
'earernay'), ('gentle', 'entlegay'), ('pitied', 'itiedpay'), ('nurses',
'ursesnay'), ('punish', 'unishpay')], (7, 9): [('parcels', 'arcelspay'),
('welcome', 'elcomeway'), ('posture', 'osturepay'), ('mothers', 'othersmay'),
('tempers', 'emperstay'), ('carries', 'arriescay'), ('neglect', 'eglectnay'),
('prevail', 'evailpray'), ('declare', 'eclareday'), ('crowned', 'ownedcray'),
('mindful', 'indfulmay'), ('blessed', 'essedblay'), ('kitchen', 'itchenkay'),
('younger', 'youngeray'), ('hanover', 'anoverhay'), ('lowered', 'oweredlay'),
('summons', 'ummonssay'), ('suffice', 'ufficesay'), ('rousing', 'ousingray'),
('feeling', 'eelingfay'), ('parlors', 'arlorspay'), ('gravely', 'avelygray'),
('connect', 'onnectcay'), ('species', 'eciesspay'), ('revived', 'evivedray'),
('tribute', 'ibutetray'), ('pitying', 'ityingpay'), ('brother', 'otherbray'),
('retired', 'etiredray'), ('suffers', 'ufferssay'), ('spurned', 'urnedspay'),
('dropped', 'oppeddray'), ('galling', 'allinggay'), ('seducer', 'educersay'),
('foibles', 'oiblesfay'), ('tuesday', 'uesdaytay'), ('patient', 'atientpay'),
('worsted', 'orstedway'), ('trifled', 'ifledtray'), ('waiting', 'aitingway'),
('widower', 'idowerway'), ('hardily', 'ardilyhay'), ('glimpse', 'impseglay'),
('whether', 'etherwhay'), ('reverie', 'everieray'), ('resolve', 'esolveray'),
('propose', 'oposepray'), ('tallest', 'allesttay'), ('nominal', 'ominalnay'),
('languid', 'anguidlay'), ('wronged', 'ongedwray'), ('nothing', 'othingnay'),
('promote', 'omotepray'), ('novelty', 'oveltynay'), ('dooming', 'oomingday'),
('display', 'isplayday'), ('correct', 'orrectcay'), ('feigned', 'eignedfay'),
('schemes', 'emesschay'), ('deepest', 'eepestday'), ('flannel', 'annelflay'),
('hauteur', 'auteurhay'), ('sharing', 'aringshay'), ('getting', 'ettinggay'),
('screams', 'eamsscray'), ('variety', 'arietyvay'), ('yielded', 'yieldeday'),
('brought', 'oughtbray'), ('haughty', 'aughtyhay'), ('bristol', 'istolbray'),
('silence', 'ilencesay'), ('staying', 'ayingstay'), ('confess', 'onfesscay'),
('placing', 'acingplay'), ('noisier', 'oisiernay'), ('nursing', 'ursingnay'),
('tuition', 'uitiontay'), ('welfare', 'elfareway'), ('hunters', 'untershay'),
('poplars', 'oplarspay'), ('deprive', 'epriveday'), ('mansion', 'ansionmay'),
('reasons', 'easonsray'), ('conceit', 'onceitcay'), ('burgess', 'urgessbay'),
('receipt', 'eceiptray'), ('befalls', 'efallsbay'), ('fetches', 'etchesfay'),
('ranging', 'angingray'), ('passion', 'assionpay'), ('smirked', 'irkedsmay'),
('honours', 'onourshay'), ('covered', 'overedcay'), ('dancing', 'ancingday'),
('hundred', 'undredhay'), ('forgave', 'orgavefay'), ('fifteen', 'ifteenfay'),
('lasting', 'astinglay'), ('minuter', 'inutermay'), ('tremour', 'emourtray'),
('passing', 'assingpay'), ('streets', 'eetsstray'), ('resists', 'esistsray'),
('cleared', 'earedclay'), ('miracle', 'iraclemay'), ('several', 'everalsay'),
('sighing', 'ighingsay'), ('resumed', 'esumedray'), ('merited', 'eritedmay'),
('without', 'ithoutway'), ('removed', 'emovedray'), ('behaved', 'ehavedbay'),
('succour', 'uccoursay'), ('bribing', 'ibingbray'), ('charity', 'aritychay'),
('shortly', 'ortlyshay'), ('tempted', 'emptedtay'), ('briefly', 'ieflybray'),
('parties', 'artiespay'), ('touched', 'ouchedtay'), ('content', 'ontentcay'),
```

```
('proceed', 'oceedpray'), ('windows', 'indowsway'), ('ribbons', 'ibbonsray'),
('stepped', 'eppedstay'), ('pursuit', 'ursuitpay'), ('quarter', 'arterquay'),
('prudent', 'udentpray'), ('dignity', 'ignityday'), ('payment', 'aymentpay'),
('grounds', 'oundsgray'), ('bequest', 'equestbay'), ('donavan', 'onavanday'),
('renewed', 'enewedray'), ('smaller', 'allersmay'), ('fearful', 'earfulfay'),
('civilly', 'ivillycay'), ('romance', 'omanceray'), ('females', 'emalesfay'),
('grieved', 'ievedgray'), ('created', 'eatedcray'), ('command', 'ommandcay'),
('defined', 'efinedday'), ('bluntly', 'untlyblay'), ('recital', 'ecitalray'),
('sorrows', 'orrowssay'), ('whither', 'itherwhay'), ('spotted', 'ottedspay'),
('justify', 'ustifyjay'), ('conduit', 'onduitcay'), ('revenge', 'evengeray'),
('scenery', 'eneryscay'), ('started', 'artedstay'), ('degrees', 'egreesday'),
('whisper', 'isperwhay'), ('strange', 'angestray'), ('bidding', 'iddingbay'),
('musical', 'usicalmay'), ('happens', 'appenshay'), ('compare', 'omparecay'),
('presume', 'esumepray'), ('sitting', 'ittingsay'), ('regards', 'egardsray'),
('violins', 'iolinsvay'), ('rightly', 'ightlyray'), ('memento', 'ementomay'),
('courted', 'ourtedcay'), ('servant', 'ervantsay'), ('possess', 'ossesspay'),
('herself', 'erselfhay'), ('present', 'esentpray'), ('natural', 'aturalnay'),
('managed', 'anagedmay'), ('hostess', 'ostesshay'), ('leaving', 'eavinglay'),
('mention', 'entionmay'), ('circuit', 'ircuitcay'), ('counsel', 'ounselcay'),
('guineas', 'uineasgay'), ('solidly', 'olidlysay'), ('dashing', 'ashingday'),
('station', 'ationstay'), ('written', 'ittenwray'), ('writing', 'itingwray'),
('country', 'ountrycay'), ('delight', 'elightday'), ('westons', 'estonsway'),
('debated', 'ebatedday'), ('lengths', 'engthslay'), ('torment', 'ormenttay'),
('reverse', 'everseray'), ('bedroom', 'edroombay'), ('forbade', 'orbadefay'),
('comings', 'omingscay'), ('tonight', 'onighttay'), ('similar', 'imilarsay'),
('calling', 'allingcay'), ('cassino', 'assinocay'), ('strains', 'ainsstray'),
('scratch', 'atchscray'), ('gardens', 'ardensgay'), ('freedom', 'eedomfray'),
('compact', 'ompactcay'), ('message', 'essagemay'), ('heavily', 'eavilyhay'),
('manners', 'annersmay'), ('systems', 'ystemssay'), ('forbore', 'orborefay'),
('wrapped', 'appedwray'), ('defects', 'efectsday'), ('villain', 'illainvay'),
('studied', 'udiedstay'), ('playing', 'ayingplay'), ('diction', 'ictionday'),
('tending', 'endingtay'), ('seeking', 'eekingsay'), ('rapture', 'aptureray'),
('dawdled', 'awdledday'), ('wearing', 'earingway'), ('comfort', 'omfortcay'),
('finally', 'inallyfay'), ('circles', 'irclescay'), ('putting', 'uttingpay'),
('twisted', 'istedtway'), ('destroy', 'estroyday'), ('whereas', 'ereaswhay'),
('dismiss', 'ismissday'), ('visibly', 'isiblyvay'), ('talents', 'alentstay'),
('finding', 'indingfay'), ('removes', 'emovesray'), ('replied', 'epliedray'),
('bluster', 'usterblay'), ('raising', 'aisingray'), ('plainly', 'ainlyplay'),
('worried', 'orriedway'), ('meaning', 'eaningmay'), ('blushes', 'ushesblay'),
('relapse', 'elapseray'), ('careful', 'arefulcay'), ('beloved', 'elovedbay'),
('degrade', 'egradeday'), ('foresaw', 'oresawfay'), ('procure', 'ocurepray'),
('kindest', 'indestkay'), ('secures', 'ecuressay'), ('hanging', 'anginghay'),
('persons', 'ersonspay'), ('taverns', 'avernstay'), ('walking', 'alkingway'),
('quicker', 'ickerquay'), ('guessed', 'uessedgay'), ('confirm', 'onfirmcay'),
('venture', 'enturevay'), ('project', 'ojectpray'), ('prayers', 'ayerspray'),
('heights', 'eightshay'), ('subject', 'ubjectsay'), ('summits', 'ummitssay'),
('towards', 'owardstay'), ('closing', 'osingclay'), ('journey', 'ourneyjay'),
('pointer', 'ointerpay'), ('chained', 'ainedchay'), ('clothes', 'othesclay'),
```

('require', 'equireray'), ('chusing', 'usingchay'), ('reproof', 'eproofray'),
('reckons', 'eckonsray'), ('pressed', 'essedpray'), ('talking', 'alkingtay'),
('harbour', 'arbourhay'), ('garrets', 'arretsgay'), ('foresee', 'oreseefay'),
('lustily', 'ustilylay'), ('leagued', 'eaguedlay'), ('judging', 'udgingjay'),
('rejoice', 'ejoiceray'), ('honiton', 'onitonhay'), ('shorten', 'ortenshay'),
('stating', 'atingstay'), ('privacy', 'ivacypray'), ('steeles', 'eelesstay'),
('thickly', 'icklythay'), ('shocked', 'ockedshay'), ('limited', 'imitedlay'),
('blights', 'ightsblay'), ('seclude', 'ecludesay'), ('cordial', 'ordialcay'),
('request', 'equestray'), ('denoted', 'enotedday'), ('darling', 'arlingday'),
('coxcomb', 'oxcombcay'), ('however', 'oweverhay'), ('perform', 'erformpay'),
('society', 'ocietysay'), ('deserts', 'esertsday'), ('palmers', 'almerspay'),
('manager', 'anagermay'), ('rupture', 'uptureray'), ('concise', 'oncisecay'),
('scolded', 'oldedscay'), ('seduced', 'educedsay'), ('doubted', 'oubtedday'),
('forward', 'orwardfay'), ('leisure', 'eisurelay'), ('robbing', 'obbingray'),
('horizon', 'orizonhay'), ('blunder', 'underblay'), ('flushed', 'ushedflay'),
('divided', 'ividedday'), ('charged', 'argedchay'), ('disturb', 'isturbday'),
('smitten', 'ittensmay'), ('seconds', 'econdssay'), ('letters', 'etterslay'),
('provide', 'ovidepray'), ('conduct', 'onductcay'), ('plagued', 'aguedplay'),
('stretch', 'etchstray'), ('hoarded', 'oardedhay'), ('blushed', 'ushedblay'),
('nodding', 'oddingnay'), ('mankind', 'ankindmay'), ('caprice', 'apricecay'),
('quitted', 'ittedquay')], (5, 7): [('china', 'inachay'), ('deter', 'eterday'),
('woman', 'omanway'), ('clock', 'ockclay'), ('borne', 'ornebay'), ('noise',
'oisenay'), ('midst', 'idstmay'), ('dwell', 'elldway'), ('smith', 'ithsmay'),
('gowns', 'ownsgay'), ('named', 'amednay'), ('glass', 'assglay'), ('price',
'icepray'), ('close', 'oseclay'), ('guide', 'uidegay'), ('newly', 'ewlynay'),
('sweep', 'eepsway'), ('swept', 'eptsway'), ('those', 'osethay'), ('lower',
'owerlay'), ('plump', 'umpplay'), ('downs', 'ownsday'), ('purse', 'ursepay'),
('linen', 'inenlay'), ('later', 'aterlay'), ('broad', 'oadbray'), ('miles',
'ilesmay'), ('facts', 'actsfay'), ('sleep', 'eepslay'), ('steps', 'epsstay'),
('wants', 'antsway'), ('forth', 'orthfay'), ('drove', 'ovedray'), ('mouth',
'outhmay'), ('worry', 'orryway'), ('dried', 'ieddray'), ('stood', 'oodstay'),
('brush', 'ushbray'), ('sting', 'ingstay'), ('mirth', 'irthmay'), ('heads',
'eadshay'), ('sight', 'ightsay'), ('whose', 'osewhay'), ('drawn', 'awndray'),
('score', 'orescay'), ('party', 'artypay'), ('tears', 'earstay'), ('heart',
'earthay'), ('words', 'ordsway'), ('hunts', 'untshay'), ('trace', 'acetray'),
('trial', 'ialtray'), ('swoon', 'oonsway'), ('knees', 'eesknay'), ('bring',
'ingbray'), ('solid', 'olidsay'), ('sharp', 'arpshay'), ('basis', 'asisbay'),
('cavil', 'avilcay'), ('rated', 'atedray'), ('grave', 'avegray'), ('serve',
'ervesay'), ('betty', 'ettybay'), ('judge', 'udgejay'), ('boast', 'oastbay'),
('march', 'archmay'), ('built', 'uiltbay'), ('clear', 'earclay'), ('happy',
'appyhay'), ('fruit', 'uitfray'), ('witty', 'ittyway'), ('total', 'otaltay'),
('dared', 'aredday'), ('defer', 'eferday'), ('tried', 'iedtray'), ('vexed',
'exedvay'), ('walls', 'allsway'), ('liked', 'ikedlay'), ('prove', 'ovepray'),
('sneer', 'eersnay'), ('merit', 'eritmay'), ('scott', 'ottscay'), ('madam',
'adammay'), ('truth', 'uthtray'), ('minor', 'inormay'), ('stare', 'arestay'),
('calls', 'allscay'), ('lucky', 'uckylay'), ('hence', 'encehay'), ('press',
'esspray'), ('comes', 'omescay'), ('dined', 'inedday'), ('think', 'inkthay'),
('plate', 'ateplay'), ('grief', 'iefgray'), ('names', 'amesnay'), ('stock',

'ockstay'), ('short', 'ortshay'), ('moved', 'ovedmay'), ('teach', 'eachtay'), ('fears', 'earsfay'), ('tiled', 'iledtay'), ('cried', 'iedcray'), ('mound', 'oundmay'), ('small', 'allsmay'), ('twice', 'icetway'), ('lines', 'ineslay'), ('looks', 'ookslay'), ('rival', 'ivalray'), ('first', 'irstfay'), ('wound', 'oundway'), ('saves', 'avessay'), ('nurse', 'ursenay'), ('treat', 'eattray'), ('could', 'ouldcay'), ('rider', 'iderray'), ('lobby', 'obbylay'), ('table', 'abletay'), ('touch', 'ouchtay'), ('topic', 'opictay'), ('stand', 'andstay'), ('leant', 'eantlay'), ('grass', 'assgray'), ('check', 'eckchay'), ('quite', 'itequay'), ('wives', 'ivesway'), ('walks', 'alksway'), ('labor', 'aborlay'), ('house', 'ousehay'), ('drops', 'opsdray'), ('loose', 'ooselay'), ('tempt', 'empttay'), ('place', 'aceplay'), ('local', 'ocallay'), ('seats', 'eatssay'), ('gates', 'atesgay'), ('dropt', 'optdray'), ('cares', 'arescay'), ('falls', 'allsfay'), ('speak', 'eakspay'), ('shops', 'opsshay'), ('groom', 'oomgray'), ('sugar', 'ugarsay'), ('front', 'ontfray'), ('visit', 'isitvay'), ('raise', 'aiseray'), ('fetch', 'etchfay'), ('silks', 'ilkssay'), ('beset', 'esetbay'), ('rocks', 'ocksray'), ('start', 'artstay'), ('winks', 'inksway'), ('sends', 'endssay'), ('piano', 'ianopay'), ('fully', 'ullyfay'), ('widow', 'idowway'), ('wiser', 'iserway'), ('dirty', 'irtyday'), ('below', 'elowbay'), ('manor', 'anormay'), ('scale', 'alescay'), ('whole', 'olewhay'), ('minds', 'indsmay'), ('glebe', 'ebeglay'), ('fifty', 'iftyfay'), ('years', 'yearsay'), ('story', 'orystay'), ('fresh', 'eshfray'), ('begun', 'egunbay'), ('dance', 'anceday'), ('jokes', 'okesjay'), ('turns', 'urnstay'), ('honor', 'onorhay'), ('spite', 'itespay'), ('green', 'eengray'), ('saint', 'aintsay'), ('hated', 'atedhay'), ('doors', 'oorsday'), ('shown', 'ownshay'), ('power', 'owerpay'), ('today', 'odaytay'), ('floor', 'oorflay'), ('force', 'orcefay'), ('right', 'ightray'), ('trick', 'icktray'), ('dread', 'eaddray'), ('final', 'inalfay'), ('shake', 'akeshay'), ('lamps', 'ampslay'), ('dying', 'yingday'), ('black', 'ackblay'), ('grand', 'andgray'), ('field', 'ieldfay'), ('plead', 'eadplay'), ('loved', 'ovedlay'), ('terms', 'ermstay'), ('spend', 'endspay'), ('mohrs', 'ohrsmay'), ('young', 'youngay'), ('great', 'eatgray'), ('guess', 'uessgay'), ('ninny', 'innynay'), ('hasty', 'astyhay'), ('nerve', 'ervenay'), ('meals', 'ealsmay'), ('three', 'eethray'), ('shade', 'adeshay'), ('store', 'orestay'), ('daily', 'ailyday'), ('taste', 'astetay'), ('gross', 'ossgray'), ('dares', 'aresday'), ('grate', 'ategray'), ('stuff', 'uffstay'), ('sheet', 'eetshay')], (14, 18): [('breakfast-room', 'eakfastbray-oomray'), ('lavender-water', 'avenderlay-aterway'), ('broken-hearted', 'okenbray-eartedhay'), ('spunging-house', 'ungingspay-ousehay')], (9, 12): [('unwelcome', 'unwelcomeway'), ('exhausted', 'exhaustedway'), ('intrusion', 'intrusionway'), ('untouched', 'untouchedway'), ('interview', 'interviewway'), ('effusions', 'effusionsway'), ('impartial', 'impartialway'), ('inventive', 'inventiveway'), ('emergency', 'emergencyway'), ('officious', 'officiousway'), ('everybody', 'everybodyway'), ('observant', 'observantway'), ('emptiness', 'emptinessway'), ('occasions', 'occasionsway'), ('assembled', 'assembledway'), ('admirable', 'admirableway'), ('uselessly', 'uselesslyway'), ('excellent', 'excellentway'), ('associate', 'associateway'), ('affecting', 'affectingway'), ('absurdity', 'absurdityway'), ('allowable', 'allowableway'), ('appointed', 'appointedway'), ('overheard', 'overheardway'), ('improving', 'improvingway'), ('eagerness', 'eagernessway'), ('unluckily', 'unluckilyway'), ('allowance', 'allowanceway'), ('estimable', 'estimableway'),

('obviating', 'obviatingway'), ('entrusted', 'entrustedway'), ('arranging',
'arrangingway'), ('exclaimed', 'exclaimedway'), ('ingenuity', 'ingenuityway'),
('outwardly', 'outwardlyway'), ('expensive', 'expensiveway'), ('entreated',
'entreatedway'), ('unconcern', 'unconcernway'), ('obtaining', 'obtainingway'),
('interests', 'interestsway'), ('assiduous', 'assiduousway'), ('adjusting',
'adjustingway'), ('ungenteel', 'ungenteelway'), ('encourage', 'encourageway'),
('apartment', 'apartmentway'), ('inheritor', 'inheritorway'), ('unceasing',
'unceasingway'), ('insulting', 'insultingway'), ('unchanged', 'unchangedway'),
('animation', 'animationway'), ('enclosure', 'enclosureway'), ('apologize',
'apologizeway'), ('earnestly', 'earnestlyway'), ('exquisite', 'exquisiteway'),
('attending', 'attendingway'), ('invention', 'inventionway'), ('enquiries',
'enquiriesway'), ('appetites', 'appetitesway'), ('additions', 'additionsway'),
('impudence', 'impudenceway'), ('insincere', 'insincereway'), ('abundance',
'abundanceway'), ('exchanged', 'exchangedway'), ('extremely', 'extremelyway'),
('inflicted', 'inflictedway'), ('attracted', 'attractedway'), ('inelegant',
'inelegantway'), ('undoubted', 'undoubtedway'), ('etiquette', 'etiquetteway'),
('unfeigned', 'unfeignedway'), ('agreeable', 'agreeableway'), ('unsubdued',
'unsubduedway'), ('inspiring', 'inspiringway'), ('unguarded', 'unguardedway'),
('advancing', 'advancingway'), ('undressed', 'undressedway'), ('excursive',
'excursiveway'), ('extending', 'extendingway'), ('extolling', 'extollingway'),
('offending', 'offendingway'), ('eloquence', 'eloquenceway'), ('ornaments',
'ornamentsway'), ('intruding', 'intrudingway'), ('incessant', 'incessantway'),
('unwilling', 'unwillingway'), ('annamaria', 'annamariaway'), ('unbounded',
'unboundedway'), ('incurable', 'incurableway'), ('unnoticed', 'unnoticedway'),
('expressly', 'expresslyway'), ('impetuous', 'impetuousway'), ('appearing',
'appearingway'), ('ingenious', 'ingeniousway'), ('amazement', 'amazementway'),
('animating', 'animatingway'), ('interfere', 'interfereway'), ('ascending',
'ascendingway'), ('affection', 'affectionway'), ('undergone', 'undergoneway'),
('unstudied', 'unstudiedway'), ('attempted', 'attemptedway'), ('infirmity',
'infirmityway'), ('apologies', 'apologiesway'), ('elsewhere', 'elsewhereway'),
('expecting', 'expectingway'), ('informing', 'informingway'), ('admission',
'admissionway'), ('extremity', 'extremityway')], (6, 9): [('unread',
'unreadway'), ('owning', 'owningway'), ('odious', 'odiousway'), ('either',
'eitherway'), ('insult', 'insultway'), ('awhile', 'awhileway'), ('assist',
'assistway'), ('injure', 'injureway'), ('eluded', 'eludedway'), ('others',
'othersway'), ('untied', 'untiedway'), ('excess', 'excessway'), ('urgent',
'urgentway'), ('appear', 'appearway'), ('aspect', 'aspectway'), ('issued',
'issuedway'), ('evince', 'evinceway'), ('eating', 'eatingway'), ('easter',
'easterway'), ('arisen', 'arisenway'), ('earned', 'earnedway'), ('esteem',
'esteemway'), ('accuse', 'accuseway'), ('owners', 'ownersway'), ('events',
'eventsway'), ('energy', 'energyway'), ('affair', 'affairway'), ('effect',
'effectway'), ('unlike', 'unlikeway'), ('unison', 'unisonway'), ('enough',
'enoughway'), ('invite', 'inviteway'), ('income', 'incomeway'), ('extent',
'extentway'), ('office', 'officeway'), ('ankles', 'anklesway'), ('admire',
'admireway'), ('induce', 'induceway'), ('infant', 'infantway'), ('expand',
'expandway'), ('orders', 'ordersway'), ('occupy', 'occupyway'), ('attend',
'attendway'), ('unable', 'unableway'), ('olives', 'olivesway'), ('awaken',
'awakenway'), ('engage', 'engageway'), ('eldest', 'eldestway'), ('eyeing',

'eyeingway'), ('advice', 'adviceway'), ('always', 'alwaysway'), ('insist',
'insistway'), ('accept', 'acceptway'), ('indies', 'indiesway'), ('excite',
'exciteway'), ('openly', 'openlyway'), ('oblige', 'obligeway'), ('amidst',
'amidstway'), ('across', 'acrossway'), ('afraid', 'afraidway'), ('uproar',
'uproarway')], (8, 10): [('villages', 'illagesvay'), ('doubtful', 'oubtfulday'),
('deceived', 'eceivedday'), ('foreseen', 'oreseenfay'), ('dennison',
'ennisonday'), ('steadily', 'eadilystay'), ('startled', 'artledstay'),
('drawings', 'awingsdray'), ('rejected', 'ejectedray'), ('deserves',
'eservesday'), ('curricle', 'urriclecay'), ('domestic', 'omesticday'),
('believed', 'elievedbay'), ('packages', 'ackagespay'), ('kneeling',
'eelingknay'), ('degraded', 'egradedday'), ('medicine', 'edicinemay'),
('promoted', 'omotedpray'), ('wondered', 'onderedway'), ('reproach',
'eproachray'), ('somerset', 'omersetsay'), ('contents', 'ontentscay'),
('children', 'ildrenchay'), ('politics', 'oliticspay'), ('hastened',
'astenedhay'), ('relative', 'elativeray'), ('henshawe', 'enshawehay'),
('material', 'aterialmay'), ('welcomed', 'elcomedway'), ('weddings',
'eddingsway'), ('diverted', 'ivertedday'), ('patience', 'atiencepay'),
('delivery', 'eliveryday'), ('hazarded', 'azardedhay'), ('happiest',
'appiesthay'), ('trifling', 'iflingtray'), ('casement', 'asementcay'),
('reformed', 'eformedray'), ('demanded', 'emandedday'), ('ceremony',
'eremonycay'), ('vigorous', 'igorousvay'), ('mornings', 'orningsmay'),
('hilarity', 'ilarityhay'), ('daughter', 'aughterday'), ('vexation',
'exationvay'), ('shocking', 'ockingshay'), ('previous', 'eviouspray'),
('pleasant', 'easantplay'), ('candidly', 'andidlycay'), ('cheerful',
'eerfulchay'), ('festival', 'estivalfay'), ('changing', 'angingchay'),
('conveyed', 'onveyedcay'), ('lombardy', 'ombardylay'), ('militate',
'ilitatemay'), ('humanity', 'umanityhay'), ('giggling', 'igglinggay'),
('standard', 'andardstay'), ('returned', 'eturnedray'), ('proclaim',
'oclaimpray'), ('ridicule', 'idiculeray'), ('pointers', 'ointerspay'),
('grounded', 'oundedgray'), ('contrast', 'ontrastcay'), ('feelings',
'eelingsfay'), ('mistakes', 'istakesmay'), ('prepared', 'eparedpray'),
('stranger', 'angerstray'), ('habitual', 'abitualhay'), ('firmness',
'irmnessfay'), ('rational', 'ationalray'), ('regulate', 'egulateray'),
('claiming', 'aimingclay'), ('knocking', 'ockingknay'), ('servants',
'ervantssay'), ('goodness', 'oodnessgay'), ('mutually', 'utuallymay'),
('legacies', 'egacieslay'), ('sisterly', 'isterlysay'), ('punctual',
'unctualpay'), ('deciding', 'ecidingday'), ('suitable', 'uitablesay'),
('guarding', 'uardinggay'), ('converse', 'onversecay'), ('slighter',
'ighterslay'), ('dissuade', 'issuadeday'), ('despatch', 'espatchday'),
('blackest', 'ackestblay'), ('recalled', 'ecalledray'), ('loitered',
'oiteredlay'), ('dreading', 'eadingdray'), ('relieved', 'elievedray'),
('westward', 'estwardway'), ('thistles', 'istlesthay'), ('defended',
'efendedday'), ('contempt', 'ontemptcay'), ('sensibly', 'ensiblysay'),
('romantic', 'omanticray'), ('comforts', 'omfortscay'), ('jealousy',
'ealousyjay'), ('repeated', 'epeatedray'), ('banditti', 'andittibay'),
('greatest', 'eatestgray'), ('mistrust', 'istrustmay'), ('forwards',
'orwardsfay'), ('rewarded', 'ewardedray'), ('watching', 'atchingway'),
('negative', 'egativenay'), ('sympathy', 'ympathysay'), ('helpless',

'elplesshay'), ('quickest', 'ickestquay'), ('portrait', 'ortraitpay'),
('consoled', 'onsoledcay'), ('fourteen', 'ourteenfay'), ('retarded',
'etardedray'), ('comments', 'ommentscay'), ('jennings', 'enningsjay'),
('securing', 'ecuringsay'), ('withdrew', 'ithdrewway'), ('youthful',
'youthfulay'), ('bachelor', 'achelorbay'), ('convince', 'onvincecay'),
('measured', 'easuredmay'), ('goodwill', 'oodwillgay'), ('branched',
'anchedbray'), ('purchase', 'urchasepay'), ('covenant', 'ovenantcay'),
('reseated', 'eseatedray'), ('concerns', 'oncernscay'), ('hardship',
'ardshiphay'), ('lessened', 'essenedlay'), ('delaford', 'elafordday'),
('governed', 'overnedgay'), ('listened', 'istenedlay'), ('constant',
'onstantcay'), ('fatigued', 'atiguedfay'), ('persuade', 'ersuadepay'),
('critique', 'itiquecray'), ('remedies', 'emediesray'), ('stepping',
'eppingstay'), ('kindness', 'indnesskay'), ('sociable', 'ociablesay'),
('meetings', 'eetingsmay'), ('deserved', 'eservedday'), ('thoughts',
'oughtsthay'), ('becoming', 'ecomingbay'), ('heartily', 'eartilyhay'),
('promised', 'omisedpray'), ('shutting', 'uttingshay'), ('reproved',
'eprovedray'), ('ladyship', 'adyshiplay'), ('pursuing', 'ursuingpay'),
('residing', 'esidingray'), ('sentence', 'entencesay'), ('revealed',
'evealedray'), ('decrease', 'ecreaseday'), ('speedily', 'eedilyspay'),
('commands', 'ommandscay'), ('mistaken', 'istakenmay'), ('settling',
'ettlingsay'), ('polished', 'olishedpay'), ('stanhill', 'anhillstay'),
('retailed', 'etailedray'), ('pointing', 'ointingpay'), ('strongly',
'onglystray'), ('subjects', 'ubjectssay'), ('calmness', 'almnesscay'),
('doctrine', 'octrineday'), ('handsome', 'andsomehay'), ('profited',
'ofitedpray'), ('planning', 'anningplay'), ('denoting', 'enotingday'),
('commonly', 'ommonlycay'), ('bestowed', 'estowedbay'), ('marriage',
'arriagemay'), ('presents', 'esentspray'), ('complete', 'ompletecay'),
('striking', 'ikingstray'), ('needless', 'eedlessnay'), ('plighted',
'ightedplay'), ('tranquil', 'anquiltray'), ('spacious', 'aciousspay'),
('nonsense', 'onsensenay'), ('likeness', 'ikenesslay'), ('valuable',
'aluablevay'), ('laudable', 'audablelay'), ('delirium', 'eliriumday'),
('believes', 'elievesbay'), ('premises', 'emisespray'), ('presumed',
'esumedpray'), ('coldness', 'oldnesscay'), ('metaphor', 'etaphormay'),
('thankful', 'ankfulthay'), ('suffered', 'ufferedsay'), ('preserve',
'eservepray'), ('resorted', 'esortedray'), ('felicity', 'elicityfay'),
('betrayed', 'etrayedbay'), ('throwing', 'owingthray'), ('powerful',
'owerfulpay'), ('humoured', 'umouredhay'), ('practise', 'actisepray'),
('coloured', 'olouredcay'), ('regiment', 'egimentray'), ('stubborn',
'ubbornstay'), ('huswifes', 'uswifeshay'), ('painting', 'aintingpay'),
('friendly', 'iendlyfray'), ('happened', 'appenedhay'), ('moderate',
'oderatemay'), ('solitary', 'olitarysay'), ('wildness', 'ildnessway'),
('marianne', 'ariannemay'), ('delicacy', 'elicacyday'), ('progress',
'ogresspray'), ('coachman', 'oachmancay'), ('grieving', 'ievinggray'),
('troubles', 'oublestray'), ('coolness', 'oolnesscay'), ('possibly',
'ossiblypay'), ('relating', 'elatingray'), ('patterns', 'atternspay'),
('consists', 'onsistscay'), ('creating', 'eatingcray'), ('confused',
'onfusedcay'), ('critical', 'iticalcray'), ('laboured', 'abouredlay'),
('minutely', 'inutelymay'), ('tendency', 'endencytay'), ('distrust',

'istrustday'), ('distress', 'istressday'), ('conclude', 'oncludecay'), ('sixpence', 'ixpencesay'), ('somewhat', 'omewhatsay'), ('coincide', 'oincidecay'), ('harmless', 'armlesshay'), ('juvenile', 'uvenilejay'), ('pleasing', 'easingplay'), ('precious', 'eciouspray'), ('restrain', 'estrainray'), ('produced', 'oducedpray'), ('renewing', 'enewingray'), ('pinching', 'inchingpay'), ('severely', 'everelysay'), ('blossoms', 'ossomsblay'), ('heighten', 'eightenhay'), ('wandered', 'anderedway'), ('properly', 'operlypray'), ('veracity', 'eracityvay'), ('released', 'eleasedray'), ('transact', 'ansacttray'), ('helpmate', 'elpmatehay'), ('business', 'usinessbay'), ('concerto', 'oncertocay'), ('reserved', 'eservedray'), ('reliance', 'elianceray'), ('headache', 'eadachehay'), ('families', 'amiliesfay'), ('centered', 'enteredcay'), ('mulberry', 'ulberrymay'), ('sensible', 'ensiblesay'), ('steepest', 'eepeststay'), ('compared', 'omparedcay'), ('speaking', 'eakingspay'), ('censured', 'ensuredcay'), ('screened', 'eenedscray'), ('pictures', 'icturespay'), ('reaching', 'eachingray'), ('wonderer', 'ondererway'), ('whenever', 'eneverwhay'), ('detested', 'etestedday'), ('dispatch', 'ispatchday'), ('shutters', 'uttersshay'), ('disgrace', 'isgraceday'), ('dejected', 'ejectedday'), ('conquest', 'onquestcay'), ('foretell', 'oretellfay'), ('weakened', 'eakenedway'), ('brooding', 'oodingbray'), ('shedding', 'eddingshay'), ('poignant', 'oignantpay'), ('trembled', 'embledtray'), ('resolute', 'esoluteray'), ('reverted', 'evertedray'), ('silenced', 'ilencedsay'), ('jointure', 'ointurejay')], (10, 13): [('illaudable', 'illaudableway'), ('incivility', 'incivilityway'), ('oppression', 'oppressionway'), ('acquainted', 'acquaintedway'), ('economical', 'economicalway'), ('ebullition', 'ebullitionway'), ('affronting', 'affrontingway'), ('unforeseen', 'unforeseenway'), ('auspicious', 'auspiciousway'), ('indecorous', 'indecorousway'), ('attractive', 'attractiveway'), ('exercising', 'exercisingway'), ('unsatiable', 'unsatiableway'), ('individual', 'individualway'), ('enthusiasm', 'enthusiasmway'), ('announcing', 'announcingway'), ('indistinct', 'indistinctway'), ('affliction', 'afflictionway'), ('imaginable', 'imaginableway'), ('obstructed', 'obstructedway'), ('improperly', 'improperlyway'), ('illiterate', 'illiterateway'), ('interested', 'interestedway'), ('infamously', 'infamouslyway'), ('amusements', 'amusementsway'), ('endeavours', 'endeavoursway'), ('inhabiting', 'inhabitingway'), ('indisposed', 'indisposedway'), ('acquiesced', 'acquiescedway'), ('anticipate', 'anticipateway'), ('accurately', 'accuratelyway'), ('influenced', 'influencedway'), ('assemblies', 'assembliesway'), ('approached', 'approachedway'), ('accusation', 'accusationway'), ('apparently', 'apparentlyway'), ('intimately', 'intimatelyway'), ('uneasiness', 'uneasinessway'), ('alteration', 'alterationway'), ('artificial', 'artificialway'), ('admiration', 'admirationway'), ('accidental', 'accidentalway'), ('unexpected', 'unexpectedway'), ('insensible', 'insensibleway'), ('expressing', 'expressingway'), ('accelerate', 'accelerateway'), ('everything', 'everythingway'), ('increasing', 'increasingway'), ('undervalue', 'undervalueway'), ('invaluable', 'invaluableway'), ('affections',

```
'affectionsway'), ('infinitely', 'infinitelyway'), ('unsuitable',
'unsuitableway'), ('authorised', 'authorisedway'), ('impudently',
'impudentlyway'), ('unpardoned', 'unpardonedway'), ('inevitable',
'inevitableway'), ('ungenerous', 'ungenerousway'), ('impression',
'impressionway'), ('indelicacy', 'indelicacyway'), ('inducement',
'inducementway'), ('unaffected', 'unaffectedway'), ('instigated',
'instigatedway'), ('inquietude', 'inquietudeway'), ('attempting',
'attemptingway'), ('experience', 'experienceway'), ('impatience',
'impatienceway'), ('indulgence', 'indulgenceway'), ('affability',
'affabilityway'), ('unavailing', 'unavailingway'), ('incommoded',
'incommodedway'), ('altogether', 'altogetherway')], (4, 6): [('done', 'oneday'),
('wife', 'ifeway'), ('plea', 'eaplay'), ('mall', 'allmay'), ('sang', 'angsay'),
('clue', 'ueclay'), ('bell', 'ellbay'), ('bind', 'indbay'), ('join', 'oinjay'),
('went', 'entway'), ('vile', 'ilevay'), ('hens', 'enshay'), ('tore', 'oretay'),
('rage', 'ageray'), ('flow', 'owflay'), ('gate', 'ategay'), ('palm', 'almpay'),
('give', 'ivegay'), ('lips', 'ipslay'), ('sink', 'inksay'), ('rent', 'entray'),
('gave', 'avegay'), ('joke', 'okejay'), ('miss', 'issmay'), ('mere', 'eremay'),
('made', 'ademay'), ('dogs', 'ogsday'), ('bent', 'entbay'), ('duly', 'ulyday'),
('when', 'enwhay'), ('rate', 'ateray'), ('sure', 'uresay'), ('land', 'andlay'),
('pink', 'inkpay'), ('hint', 'inthay'), ('cast', 'astcay'), ('park', 'arkpay'),
('bold', 'oldbay'), ('door', 'oorday'), ('bulk', 'ulkbay'), ('doat', 'oatday'),
('coat', 'oatcay'), ('date', 'ateday'), ('foot', 'ootfay'), ('hand', 'andhay'),
('damp', 'ampday'), ('pall', 'allpay'), ('goes', 'oesgay'), ('rung', 'ungray'),
('none', 'onenay'), ('game', 'amegay'), ('nods', 'odsnay'), ('read', 'eadray'),
('gold', 'oldgay'), ('sing', 'ingsay'), ('puts', 'utspay'), ('cats', 'atscay'),
('many', 'anymay'), ('dare', 'areday'), ('noon', 'oonnay'), ('said', 'aidsay'),
('mark', 'arkmay'), ('case', 'asecay'), ('hate', 'atehay'), ('word', 'ordway'),
('beat', 'eatbay'), ('lose', 'oselay'), ('fish', 'ishfay'), ('laid', 'aidlay'),
('fine', 'inefay'), ('snug', 'ugsnay'), ('moss', 'ossmay'), ('heir', 'eirhay'),
('sigh', 'ighsay'), ('mend', 'endmay'), ('sake', 'akesay'), ('pass', 'asspay'),
('debt', 'ebtday'), ('lane', 'anelay'), ('days', 'aysday'), ('deep', 'eepday'),
('song', 'ongsay'), ('spot', 'otspay'), ('took', 'ooktay'), ('both', 'othbay'),
('year', 'yearay'), ('mama', 'amamay'), ('hill', 'illhay'), ('beds', 'edsbay'),
('less', 'esslay'), ('gown', 'owngay'), ('kill', 'illkay'), ('free', 'eefray'),
('hall', 'allhay'), ('room', 'oomray'), ('gray', 'aygray'), ('beef', 'eefbay'),
('kept', 'eptkay'), ('what', 'atwhay'), ('blue', 'ueblay'), ('fell', 'ellfay'),
('draw', 'awdray'), ('play', 'ayplay'), ('baby', 'abybay'), ('rude', 'uderay'),
('trim', 'imtray'), ('chat', 'atchay'), ('bely', 'elybay'), ('fled', 'edflay'),
('past', 'astpay'), ('wait', 'aitway'), ('meal', 'ealmay'), ('wont', 'ontway'),
('week', 'eekway'), ('best', 'estbay'), ('line', 'inelay'), ('lawn', 'awnlay'),
('tree', 'eetray'), ('tend', 'endtay'), ('find', 'indfay'), ('five', 'ivefay'),
('gone', 'onegay'), ('need', 'eednay'), ('drew', 'ewdray'), ('more', 'oremay'),
('cool', 'oolcay'), ('suit', 'uitsay'), ('from', 'omfray'), ('cure', 'urecay'),
('will', 'illway'), ('slow', 'owslay'), ('they', 'eythay'), ('name', 'amenay'),
('shot', 'otshay'), ('come', 'omecay'), ('tied', 'iedtay'), ('came', 'amecay'),
('boys', 'oysbay'), ('wish', 'ishway'), ('hour', 'ourhay'), ('kind', 'indkay'),
('mile', 'ilemay'), ('quit', 'itquay'), ('john', 'ohnjay'), ('hazy', 'azyhay'),
('fall', 'allfay'), ('sell', 'ellsay'), ('tale', 'aletay'), ('grew', 'ewgray'),
```

```
('seek', 'eeksay'), ('muff', 'uffmay'), ('soft', 'oftsay'), ('same', 'amesay'),
('body', 'odybay'), ('head', 'eadhay'), ('like', 'ikelay'), ('here', 'erehay'),
('help', 'elphay'), ('sail', 'ailsay'), ('pope', 'opepay'), ('deny', 'enyday'),
('bond', 'ondbay'), ('bath', 'athbay'), ('blow', 'owblay'), ('hold', 'oldhay'),
('safe', 'afesay'), ('path', 'athpay'), ('feed', 'eedfay'), ('thus', 'usthay'),
('four', 'ourfay'), ('lest', 'estlay'), ('thin', 'inthay')], (11, 14):
[('approaching', 'approachingway'), ('unequivocal', 'unequivocalway'),
('exceedingly', 'exceedinglyway'), ('application', 'applicationway'),
('observation', 'observationway'), ('unhappiness', 'unhappinessway'),
('interrupted', 'interruptedway'), ('enlightened', 'enlightenedway'),
('appearances', 'appearancesway'), ('unconnected', 'unconnectedway'),
('unconcerned', 'unconcernedway'), ('indignation', 'indignationway'),
('accommodate', 'accommodateway'), ('inattention', 'inattentionway'),
('undoubtedly', 'undoubtedlyway'), ('intercourse', 'intercourseway'),
('imperfectly', 'imperfectlyway'), ('imprudently', 'imprudentlyway'),
('affectation', 'affectationway'), ('instigation', 'instigationway'),
('inclination', 'inclinationway'), ('undervalued', 'undervaluedway'),
('affirmative', 'affirmativeway'), ('inconstancy', 'inconstancyway'),
('universally', 'universallyway'), ('engagements', 'engagementsway'),
('exclamation', 'exclamationway'), ('elucidation', 'elucidationway'),
('accustomary', 'accustomaryway'), ('approbation', 'approbationway'),
('unnecessary', 'unnecessaryway'), ('associating', 'associatingway'),
('attributing', 'attributingway'), ('unobtrusive', 'unobtrusiveway'),
('impropriety', 'improprietyway'), ('examination', 'examinationway'),
('alterations', 'alterationsway'), ('unexpressed', 'unexpressedway'),
('abridgement', 'abridgementway'), ('exhilarated', 'exhilaratedway'),
('unsuspected', 'unsuspectedway'), ('independent', 'independentway'),
('alternately', 'alternatelyway'), ('enforcement', 'enforcementway'),
('undertaking', 'undertakingway'), ('outstripped', 'outstrippedway'),
('appointment', 'appointmentway'), ('explanation', 'explanationway'),
('experienced', 'experiencedway'), ('unknowingly', 'unknowinglyway'),
('undesirable', 'undesirableway'), ('ascertained', 'ascertainedway'),
('enumeration', 'enumerationway'), ('alleviation', 'alleviationway'),
('instruction', 'instructionway'), ('undeserving', 'undeservingway'),
('earnestness', 'earnestnessway'), ('insincerity', 'insincerityway'),
('accompanied', 'accompaniedway'), ('employments', 'employmentsway'),
('inferiority', 'inferiorityway'), ('inexcusable', 'inexcusableway'),
('introducing', 'introducingway'), ('extravagant', 'extravagantway'),
('impatiently', 'impatientlyway'), ('attentively', 'attentivelyway'),
('eligibility', 'eligibilityway'), ('uncertainty', 'uncertaintyway')], (7, 10):
[('anxiety', 'anxietyway'), ('attacks', 'attacksway'), ('ailment',
'ailmentway'), ('awkward', 'awkwardway'), ('effects', 'effectsway'), ('induced',
'inducedway'), ('uncivil', 'uncivilway'), ('against', 'againstway'), ('apology',
'apologyway'), ('endless', 'endlessway'), ('alluded', 'alludedway'), ('ability',
'abilityway'), ('address', 'addressway'), ('engaged', 'engagedway'), ('eagerly',
'eagerlyway'), ('expects', 'expectsway'), ('elliott', 'elliottway'), ('oneself',
'oneselfway'), ('anybody', 'anybodyway'), ('augment', 'augmentway'), ('utility',
'utilityway'), ('unquiet', 'unquietway'), ('infancy', 'infancyway'), ('altered',
```

'alteredway'), ('unequal', 'unequalway'), ('adapted', 'adaptedway'), ('useless',
'uselessway'), ('opposed', 'opposedway'), ('artless', 'artlessway'), ('imagery',
'imageryway'), ('amiably', 'amiablyway'), ('uncouth', 'uncouthway'), ('affixed',
'affixedway'), ('impulse', 'impulseway'), ('opinion', 'opinionway'), ('imitate',
'imitateway'), ('admires', 'admiresway'), ('affront', 'affrontway'), ('amusing',
'amusingway'), ('accents', 'accentsway'), ('implied', 'impliedway'), ('ecstasy',
'ecstasyway'), ('illness', 'illnessway'), ('entreat', 'entreatway'), ('another',
'anotherway'), ('opening', 'openingway'), ('england', 'englandway'), ('ensured',
'ensuredway'), ('imputed', 'imputedway'), ('enabled', 'enabledway'), ('offence',
'offenceway'), ('insured', 'insuredway'), ('urgency', 'urgencyway'), ('amazing',
'amazingway'), ('uniform', 'uniformway'), ('avarice', 'avariceway'), ('ellison',
'ellisonway'), ('endowed', 'endowedway'), ('ashamed', 'ashamedway'), ('eloping',
'elopingway'), ('inhabit', 'inhabitway'), ('advised', 'advisedway'), ('attract',
'attractway'), ('already', 'alreadyway'), ('enquire', 'enquireway'), ('affects',
'affectsway'), ('invalid', 'invalidway')], (9, 11): [('genuinely',
'enuinelygay'), ('columella', 'olumellacay'), ('vouchsafe', 'ouchsafevay'),
('forwarded', 'orwardedfay'), ('scattered', 'atteredscay'), ('christmas',
'istmaschray'), ('crimsoned', 'imsonedcray'), ('contrives', 'ontrivescay'),
('miserable', 'iserablemay'), ('preceding', 'ecedingpray'), ('healthful',
'ealthfulhay'), ('confiding', 'onfidingcay'), ('perfectly', 'erfectlypay'),
('prospects', 'ospectspray'), ('withdrawn', 'ithdrawnway'), ('lingering',
'ingeringlay'), ('contained', 'ontainedcay'), ('depressed', 'epressedday'),
('disgusted', 'isgustedday'), ('promoting', 'omotingpray'), ('supported',
'upportedsay'), ('gradually', 'aduallygray'), ('disturbed', 'isturbedday'),
('perpetual', 'erpetualpay'), ('pronounce', 'onouncepray'), ('sincerely',
'incerelysay'), ('seventeen', 'eventeensay'), ('presented', 'esentedpray'),
('medicines', 'edicinesmay'), ('syllables', 'yllablessay'), ('behaviour',
'ehaviourbay'), ('niggardly', 'iggardlynay'), ('commonest', 'ommonestcay'),
('necessary', 'ecessarynay'), ('prettiest', 'ettiestpray'), ('condemned',
'ondemnedcay'), ('conquests', 'onquestscay'), ('fashioned', 'ashionedfay'),
('hysterics', 'ystericshay'), ('smallness', 'allnesssmay'), ('professed',
'ofessedpray'), ('conceited', 'onceitedcay'), ('compliant', 'ompliantcay'),
('frankness', 'anknessfray'), ('merriment', 'errimentmay'), ('resembled',
'esembledray'), ('leisurely', 'eisurelylay'), ('dispersed', 'ispersedday'),
('departure', 'epartureday'), ('pecuniary', 'ecuniarypay'), ('frivolous',
'ivolousfray'), ('situation', 'ituationsay'), ('discourse', 'iscourseday'),
('resettled', 'esettledray'), ('beholding', 'eholdingbay'), ('gentleman',
'entlemangay'), ('chagrined', 'agrinedchay'), ('struggled', 'uggledstray'),
('preparing', 'eparingpray'), ('respected', 'espectedray'), ('newspaper',
'ewspapernay'), ('justified', 'ustifiedjay'), ('sacrifice', 'acrificesay'),
('stammered', 'ammeredstay'), ('smartness', 'artnesssmay'), ('continued',
'ontinuedcay'), ('contrived', 'ontrivedcay'), ('reconcile', 'econcileray'),
('modestest', 'odestestmay'), ('cordially', 'ordiallycay'), ('sentiment',
'entimentsay'), ('shameless', 'amelessshay'), ('reflected', 'eflectedray'),
('feelingly', 'eelinglyfay'), ('confident', 'onfidentcay'), ('proposals',
'oposalspray'), ('warehouse', 'arehouseway'), ('deviation', 'eviationday'),
('statement', 'atementstay'), ('submitted', 'ubmittedsay'), ('believing',
'elievingbay'), ('gentility', 'entilitygay'), ('resources', 'esourcesray'),

('composure', 'omposurecay'), ('deserving', 'eservingday'), ('vehemence', 'ehemencevay'), ('disgraced', 'isgracedday'), ('perplexed', 'erplexedpay'), ('prejudice', 'ejudicepray'), ('quickened', 'ickenedquay'), ('blundered', 'underedblay'), ('strongest', 'ongeststray'), ('prevailed', 'evailedpray'), ('languages', 'anguageslay'), ('reanimate', 'eanimateray'), ('stationer', 'ationerstay'), ('reminding', 'emindingray'), ('blameless', 'amelessblay'), ('principle', 'inciplepray'), ('hardening', 'ardeninghay'), ('suggested', 'uggestedsay'), ('desperate', 'esperateday'), ('landscape', 'andscapelay'), ('paragraph', 'aragraphpay'), ('provision', 'ovisionpray'), ('relations', 'elationsray'), ('fruitless', 'uitlessfray'), ('christian', 'istianchray'), ('stipulate', 'ipulatestay'), ('constancy', 'onstancycay'), ('gentlemen', 'entlemengay'), ('venturing', 'enturingvay'), ('northward', 'orthwardnay'), ('neighbour', 'eighbournay'), ('weakening', 'eakeningway'), ('persisted', 'ersistedpay'), ('malignant', 'alignantmay'), ('wandering', 'anderingway'), ('witnesses', 'itnessesway'), ('dangerous', 'angerousday'), ('safeguard', 'afeguardsay'), ('treasured', 'easuredtray'), ('cleveland', 'evelandclay'), ('patronage', 'atronagepay'), ('despising', 'espisingday'), ('foolishly', 'oolishlyfay'), ('seriously', 'eriouslysay'), ('restraint', 'estraintray'), ('september', 'eptembersay'), ('distrusts', 'istrustsday'), ('temperate', 'emperatetay'), ('finishing', 'inishingfay'), ('persuaded', 'ersuadedpay'), ('stupified', 'upifiedstay'), ('declaring', 'eclaringday'), ('dejection', 'ejectionday'), ('soothings', 'oothingssay'), ('conveying', 'onveyingcay'), ('possessed', 'ossessedpay'), ('described', 'escribedday'), ('spreading', 'eadingspray'), ('foresight', 'oresightfay'), ('political', 'oliticalpay'), ('concluded', 'oncludedcay'), ('doubtless', 'oubtlessday'), ('confusion', 'onfusioncay'), ('disliking', 'islikingday'), ('carriages', 'arriagescay'), ('favourite', 'avouritefay'), ('brilliant', 'illiantbray'), ('hastening', 'asteninghay'), ('precisely', 'eciselypray'), ('wondering', 'onderingway'), ('shoulders', 'oulderssshay'), ('honorably', 'onorablyhay'), ('privilege', 'ivilegepray'), ('deference', 'eferenceday'), ('vulgarity', 'ulgarityvay'), ('reprimand', 'eprimandray'), ('surveying', 'urveyingsay'), ('handsomer', 'andsomerhay'), ('shivering', 'iveringshay'), ('tolerable', 'olerabletay'), ('requiring', 'equiringray'), ('precision', 'ecisionpray'), ('courtland', 'ourtlandcay'), ('sportsman', 'ortsmanspay'), ('happiness', 'appinesshay'), ('pointedly', 'ointedlypay'), ('benefited', 'enefitedbay'), ('substance', 'ubstancesay'), ('beginning', 'eginningbay'), ('gratitude', 'atitudegray'), ('negligent', 'egligentnay'), ('belonging', 'elongingbay'), ('mortified', 'ortifiedmay'), ('neglected', 'eglectednay'), ('shillings', 'illingsshay'), ('warranted', 'arrantedway'), ('performed', 'erformedpay'), ('necessity', 'ecessitynay'), ('slightest', 'ightestslay'), ('guardians', 'uardiansgay'), ('preferred', 'eferredpray'), ('forgotten', 'orgottenfay'), ('refreshed', 'efreshedray'), ('barouches', 'arouchesbay'), ('grievance', 'ievancegray'), ('seduction', 'eductionsay'), ('comprised', 'omprisedcay'), ('listening', 'isteninglay'), ('detection', 'etectionday'), ('narration', 'arrationnay'), ('deceiving', 'eceivingday'), ('provoking', 'ovokingpray'), ('cessation', 'essationcay'), ('questions', 'estionsquay'), ('continual', 'ontinualcay'), ('faintness', 'aintnessfay'), ('diffident', 'iffidentday'), ('referring', 'eferringray'), ('blessings', 'essingsblay'), ('therefore', 'ereforethay'),

```
('frugality', 'ugalityfray'), ('reprobate', 'eprobateray')], (11, 13):
[('tranquilize', 'anquilizetray'), ('fashionable', 'ashionablefay'),
('respectable', 'espectableray'), ('publication', 'ublicationpay'),
('superintend', 'uperintendsay'), ('recollected', 'ecollectedray'),
('cheerfuller', 'eerfullerchay'), ('concurrence', 'oncurrencecay'),
('restorative', 'estorativeray'), ('reproaching', 'eproachingray'),
('declaration', 'eclarationday'), ('discharging', 'ischargingday'),
('proclaiming', 'oclaimingpray'), ('discovering', 'iscoveringday'),
('possibility', 'ossibilitypay'), ('contributed', 'ontributedcay'),
('captivating', 'aptivatingcay'), ('misinformed', 'isinformedmay'),
('carefulness', 'arefulnesscay'), ('philosophic', 'ilosophicphay'),
('calculation', 'alculationcay'), ('consumption', 'onsumptioncay'),
('confinement', 'onfinementcay'), ('certainties', 'ertaintiescay'),
('destination', 'estinationday'), ('separations', 'eparationssay'),
('voluntarily', 'oluntarilyvay'), ('connoisseur', 'onnoisseurcay'),
('refinements', 'efinementsray'), ('subservient', 'ubservientsay'),
('marlborough', 'arlboroughmay'), ('sagaciously', 'agaciouslysay'),
('fascinating', 'ascinatingfay'), ('comfortable', 'omfortablecay'),
('reflections', 'eflectionsray'), ('sensibility', 'ensibilitysay'),
('forbearance', 'orbearancefay'), ('mischievous', 'ischievousmay'),
('disgraceful', 'isgracefulday'), ('temporizing', 'emporizingtay'),
('sacrificing', 'acrificingsay'), ('spontaneous', 'ontaneousspay'),
('flourishing', 'ourishingflay'), ('lamentation', 'amentationlay'),
('meditations', 'editationsmay'), ('quarrelling', 'arrellingquay'),
('discussions', 'iscussionsday'), ('supposition', 'uppositionsay'),
('comfortably', 'omfortablycay'), ('disappeared', 'isappearedday'),
('preparation', 'eparationpray'), ('consistency', 'onsistencycay'),
('compliments', 'omplimentscay'), ('promptitude', 'omptitudepray'),
('determining', 'eterminingday'), ('countenance', 'ountenancecay'),
('maintenance', 'aintenancemay'), ('contraction', 'ontractioncay'),
('dissipation', 'issipationday'), ('comparative', 'omparativecay'),
('desperation', 'esperationday'), ('displeasure', 'ispleasureday'),
('despatching', 'espatchingday'), ('playfellows', 'ayfellowsplay'),
('plantations', 'antationsplay'), ('vindication', 'indicationvay'),
('significant', 'ignificantsay'), ('discontents', 'iscontentsday'),
('housekeeper', 'ousekeeperhay'), ('playfulness', 'ayfulnessplay'),
('threatening', 'eateningthray'), ('meritorious', 'eritoriousmay'),
('tediousness', 'ediousnesstay'), ('convenience', 'onveniencecay'),
('possessions', 'ossessionspay'), ('consolation', 'onsolationcay'),
('composition', 'ompositioncay'), ('proposition', 'opositionpray'),
('correctness', 'orrectnesscay'), ('consequence', 'onsequencecay'),
('barbarously', 'arbarouslybay'), ('booksellers', 'ooksellersbay'),
('conjectures', 'onjecturescay'), ('violoncello', 'ioloncellovay'),
('dissembling', 'issemblingday'), ('rapturously', 'apturouslyray'),
('speculative', 'eculativespay'), ('remembering', 'ememberingray'),
('probability', 'obabilitypray'), ('punctuality', 'unctualitypay'),
('forgiveness', 'orgivenessfay'), ('disapproves', 'isapprovesday'),
('troublesome', 'oublesometray')], (3, 5): [('say', 'aysay'), ('for', 'orfay'),
```

```
('get', 'etgay'), ('pin', 'inpay'), ('not', 'otnay'), ('fit', 'itfay'), ('rap',
'apray'), ('lip', 'iplay'), ('pry', 'ypray'), ('rob', 'obray'), ('fir',
'irfay'), ('can', 'ancay'), ('two', 'otway'), ('him', 'imhay'), ('son',
'onsay'), ('let', 'etlay'), ('try', 'ytray'), ('hot', 'othay'), ('sum',
'umsay'), ('got', 'otgay'), ('law', 'awlay'), ('how', 'owhay'), ('far',
'arfay'), ('the', 'ethay'), ('sky', 'yskay'), ('sly', 'yslay'), ('cut',
'utcay'), ('hop', 'ophay'), ('had', 'adhay'), ('was', 'asway'), ('bow',
'owbay'), ('hue', 'uehay'), ('now', 'ownay'), ('vex', 'exvay'), ('yet',
'yetay'), ('sit', 'itsay'), ('due', 'ueday'), ('cap', 'apcay'), ('sat',
'atsay'), ('hit', 'ithay'), ('see', 'eesay'), ('pay', 'aypay'), ('nor',
'ornay'), ('fly', 'yflay'), ('dog', 'ogday'), ('too', 'ootay'), ('six',
'ixsay'), ('lap', 'aplay'), ('new', 'ewnay'), ('led', 'edlay'), ('sin',
'insay'), ('ten', 'entay'), ('his', 'ishay'), ('day', 'ayday')], (14, 17):
[('insurmountable', 'insurmountableway'), ('inconveniences',
'inconveniencesway'), ('aggrandizement', 'aggrandizementway'),
('irreconcilable', 'irreconcilableway'), ('unacknowledged',
'unacknowledgedway'), ('unpleasantness', 'unpleasantnessway')], (13, 16):
[('understanding', 'understandingway'), ('intentionally', 'intentionallyway'),
('embarrassment', 'embarrassmentway'), ('unembarrassed', 'unembarrassedway'),
('administering', 'administeringway'), ('instantaneous', 'instantaneousway'),
('intrinsically', 'intrinsicallyway'), ('indispensable', 'indispensableway'),
('impoverishing', 'impoverishingway'), ('opportunities', 'opportunitiesway'),
('impertinently', 'impertinentlyway'), ('independently', 'independentlyway'),
('encroachments', 'encroachmentsway'), ('uninteresting', 'uninterestingway'),
('investigation', 'investigationway'), ('unaccountable', 'unaccountableway'),
('irrepressible', 'irrepressibleway'), ('impossibility', 'impossibilityway')],
(10, 14): [('thirty-six', 'irtythay-ixsay')], (8, 11): [('increase',
'increaseway'), ('agitated', 'agitatedway'), ('unkindly', 'unkindlyway'),
('occurred', 'occurredway'), ('arranged', 'arrangedway'), ('appetite',
'appetiteway'), ('inclined', 'inclinedway'), ('indolent', 'indolentway'),
('inquired', 'inquiredway'), ('evenings', 'eveningsway'), ('obliging',
'obligingway'), ('equalled', 'equalledway'), ('accepted', 'acceptedway'),
('extended', 'extendedway'), ('entreaty', 'entreatyway'), ('exigence',
'exigenceway'), ('election', 'electionway'), ('examined', 'examinedway'),
('inviting', 'invitingway'), ('embitter', 'embitterway'), ('idleness',
'idlenessway'), ('admirers', 'admirersway'), ('editions', 'editionsway'),
('extorted', 'extortedway'), ('outweigh', 'outweighway'), ('attended',
'attendedway'), ('addition', 'additionway'), ('entitled', 'entitledway'),
('awakened', 'awakenedway'), ('attacked', 'attackedway'), ('arriving',
'arrivingway'), ('overlook', 'overlookway'), ('eminence', 'eminenceway'),
('adhering', 'adheringway'), ('obtained', 'obtainedway'), ('openness',
'opennessway'), ('allusion', 'allusionway'), ('anything', 'anythingway'),
('overcome', 'overcomeway'), ('announce', 'announceway'), ('enjoying',
'enjoyingway'), ('outlived', 'outlivedway'), ('emphasis', 'emphasisway'),
('insulted', 'insultedway'), ('esteemed', 'esteemedway'), ('alphabet',
'alphabetway'), ('enquired', 'enquiredway'), ('accident', 'accidentway'),
('occupied', 'occupiedway'), ('ellisons', 'ellisonsway'), ('avoiding',
'avoidingway'), ('attained', 'attainedway'), ('unwarily', 'unwarilyway'),
```

('employed', 'employedway'), ('altering', 'alteringway'), ('intimate', 'intimateway'), ('effected', 'effectedway'), ('offences', 'offencesway'), ('entirely', 'entirelyway'), ('abounded', 'aboundedway'), ('actually', 'actuallyway'), ('identify', 'identifyway'), ('applying', 'applyingway'), ('exterior', 'exteriorway'), ('auditors', 'auditorsway'), ('unfairly', 'unfairlyway'), ('eighteen', 'eighteenway'), ('embraced', 'embracedway'), ('unfolded', 'unfoldedway'), ('injuries', 'injuriesway'), ('elevated', 'elevatedway'), ('alarming', 'alarmingway'), ('adequate', 'adequateway'), ('unabated', 'unabatedway'), ('incurred', 'incurredway'), ('uncommon', 'uncommonway'), ('unawares', 'unawaresway'), ('unsuited', 'unsuitedway'), ('instance', 'instanceway'), ('interval', 'intervalway'), ('audacity', 'audacityway')], (5, 8): [('other', 'otherway'), ('ideas', 'ideasway'), ('agony', 'agonyway'), ('avoid', 'avoidway'), ('awful', 'awfulway'), ('april', 'aprilway'), ('enter', 'enterway'), ('infer', 'inferway'), ('abuse', 'abuseway'), ('ivory', 'ivoryway'), ('adopt', 'adoptway'), ('image', 'imageway'), ('agree', 'agreeway'), ('alike', 'alikeway'), ('earth', 'earthway'), ('ended', 'endedway'), ('owner', 'ownerway'), ('using', 'usingway'), ('ought', 'oughtway'), ('abhor', 'abhorway'), ('altar', 'altarway'), ('often', 'oftenway'), ('atone', 'atoneway'), ('angel', 'angelway'), ('arise', 'ariseway'), ('under', 'underway'), ('abode', 'abodeway'), ('exert', 'exertway'), ('usual', 'usualway'), ('aches', 'achesway'), ('enjoy', 'enjoyway'), ('urged', 'urgedway'), ('admit', 'admitway'), ('aside', 'asideway'), ('equal', 'equalway'), ('alive', 'aliveway'), ('arose', 'aroseway'), ('eight', 'eightway'), ('opera', 'operaway'), ('ennui', 'ennuiway'), ('utter', 'utterway'), ('ankle', 'ankleway')], (2, 5): [('is', 'isway'), ('ay', 'ayway'), ('ah', 'ahway'), ('ii', 'iiway'), ('or', 'orway'), ('am', 'amway'), ('if', 'ifway'), ('an', 'anway')], (12, 14): [('tranquillity', 'anquillitytray'), ('humiliations', 'umiliationshay'), ('contributing', 'ontributingcay'), ('significancy', 'ignificancysay'), ('wretchedness', 'etchednesswray'), ('prodigiously', 'odigiouslypray'), ('particularly', 'articularlypay'), ('carelessness', 'arelessnesscay'), ('refreshments', 'efreshmentsray'), ('satisfaction', 'atisfactionsay'), ('construction', 'onstructioncay'), ('contribution', 'ontributioncay'), ('gracefulness', 'acefulnessgray'), ('consequences', 'onsequencescay'), ('grandmothers', 'andmothersgray'), ('proportioned', 'oportionedpray'), ('perverseness', 'erversenesspay'), ('conciliation', 'onciliationcay'), ('representing', 'epresentingray'), ('watchfulness', 'atchfulnessway'), ('recollecting', 'ecollectingray'), ('hesitatingly', 'esitatinglyhay'), ('pleasantness', 'easantnessplay'), ('benefactress', 'enefactressbay'), ('distractedly', 'istractedlyday'), ('perseverance', 'erseverancepay'), ('performances', 'erformancespay'), ('disapproving', 'isapprovingday'), ('recollection', 'ecollectionray'), ('comprehended', 'omprehendedcay'), ('relinquished', 'elinquishedray'), ('conjecturing', 'onjecturingcay'), ('reservedness', 'eservednessray'), ('promontories', 'omontoriespray'), ('discontented', 'iscontentedday'), ('persecutions', 'ersecutionspay'), ('relationship', 'elationshipray'), ('triumphantly', 'iumphantlytray'), ('deliberation', 'eliberationday'), ('supplication', 'upplicationsay'), ('dishonorable', 'ishonorableday'), ('propensities',

'opensitiespray'), ('contemptuous', 'ontemptuouscay'), ('transporting', 'ansportingtray')], (13, 15): [('distinguished', 'istinguishedday'), ('resuscitation', 'esuscitationray'), ('particularity', 'articularitypay'), ('superannuated', 'uperannuatedsay'), ('disagreements', 'isagreementsday'), ('gentlemanlike', 'entlemanlikegay'), ('prepossessing', 'epossessingpray'), ('conscientious', 'onscientiouscay'), ('considerately', 'onsideratelycay'), ('felicitations', 'elicitationsfay'), ('prepossession', 'epossessionpray'), ('consternation', 'onsternationcay'), ('disproportion', 'isproportionday'), ('commiseration', 'ommiserationcay'), ('mortification', 'ortificationmay'), ('solicitations', 'olicitationssay'), ('qualification', 'alificationquay'), ('philanthropic', 'ilanthropicphay'), ('disappointing', 'isappointingday'), ('prescriptions', 'escriptionspray'), ('strengthening', 'engtheningstray')], (13, 17): [('mulberry-tree', 'ulberrymay-eetray'), ('good-humoured', 'oodgay-umouredhay'), ('mean-spirited', 'eanmay-iritedspay'), ('self-reproach', 'elfsay-eproachray'), ('drawing-table', 'awingdray-abletay'), ('well-disposed', 'ellway-isposedday'), ('proud-looking', 'oudpray-ookinglay'), ('flower-garden', 'owerflay-ardengay'), ('mansion-house', 'ansionmay-ousehay'), ('landing-place', 'andinglay-aceplay'), ('music-sellers', 'usicmay-ellerssay')], (4, 7): [('acts', 'actsway'), ('anne', 'anneway'), ('upon', 'uponway'), ('able', 'ableway'), ('inch', 'inchway'), ('ages', 'agesway'), ('envy', 'envyway'), ('idea', 'ideaway'), ('ends', 'endsway'), ('-and', '-andway'), ('eyed', 'eyedway'), ('arms', 'armsway'), ('over', 'overway'), ('easy', 'easyway'), ('ever', 'everway'), ('ours', 'oursway'), ('ears', 'earsway'), ('away', 'awayway')], (16, 20): [('pleasure-grounds', 'easureplay-oundsgray'), ('self-provocation', 'elfsay-ovocationpray')], (11, 15): [('bed-chamber', 'edbay-amberchay'), ('high-minded', 'ighhay-indedmay'), ('before-hand', 'eforebay-andhay'), ('watch-tower', 'atchway-owertay'), ('print-shops', 'intpray-opsshay'), ('card-tables', 'ardcay-ablestay'), ('post-horses', 'ostpay-orseshay'), ('good-nature', 'oodgay-aturenay'), ('dining-room', 'iningday-oomray'), ('high-church', 'ighhay-urchchay'), ('well-wisher', 'ellway-isherway'), ('bond-street', 'ondbay-eetstray'), ('carpet-work', 'arpetcay-orkway')], (2, 4): [('to', 'otay'), ('he', 'ehay'), ('no', 'onay'), ('me', 'emay')], (12, 16): [('good-hearted', 'oodgay-eartedhay'), ('well-judging', 'ellway-udgingjay'), ('drawing-room', 'awingdray-oomray'), ('common-place', 'ommoncay-aceplay'), ('self-control', 'elfsay-ontrolcay'), ('hand-writing', 'andhay-itingwray'), ('well-behaved', 'ellway-ehavedbay'), ('mantel-piece', 'antelmay-iecepay')], (10, 15): [('ill-humour', 'illway-umourhay'), ('ill-judged', 'illway-udgedjay'), ('ill-suited', 'illway-uitedsay')], (1, 3): [('c', 'cay')], (8, 13): [('ill-wind', 'illway-indway'), ('ill-will', 'illway-illway'), ('over-set', 'overway-etsay')], (17, 19): [('disqualifications', 'isqualificationsday')], (9, 16): [('by-and-by', 'ybay-andway-ybay')], (3, 6): [('aye', 'ayeway'), ('use', 'useway'), ('owe', 'oweway'), ('our', 'ourway'), ('old', 'oldway'), ('ear', 'earway'), ('its', 'itsway'), ('ill', 'illway')], (12, 17): [('ill-breeding', 'illway-eedingbray'), ('re-assurance', 'eray-assuranceway'), ('over-powered', 'overway-oweredpay'), ('ill-disposed', 'illway-isposedday')], (11, 16): [('ill-natured', 'illway-aturednay')], (14, 16): [('reconciliation', 'econciliationray'), ('recommendation', 'ecommendationray'), ('reasonableness', 'easonablenessray'), ('disinclination', 'isinclinationday'), ('respectability', 'espectabilityray'),

```
('circumspection', 'ircumspectioncay'), ('considerations', 'onsiderationscay'),
('qualifications', 'alificationsquay'), ('congratulating', 'ongratulatingcay')],
(15, 17): [('dissatisfaction', 'issatisfactionday'), ('disrespectfully',
'isrespectfullyday'), ('conscientiously', 'onscientiouslycay'),
('misconstruction', 'isconstructionmay')], (15, 18): [('instantaneously',
'instantaneouslyway')], (15, 19): [('shooting-jacket', 'ootingshay-acketjay')],
(14, 21): [('sisters-in-law', 'isterssay-inway-awlay')], (17, 21): [('self-
condemnation', 'elfsay-ondemnationcay')], (13, 20): [('sister-in-law',
'istersay-inway-awlay')], (9, 14): [('ill-timed', 'illway-imedtay')], (13, 18):
[('ill-treatment', 'illway-eatmenttray'), ('old-fashioned', 'oldway-
ashionedfay'), ('self-interest', 'elfsay-interestway')], (16, 22): [('good-for-
nothing', 'oodgay-orfay-othingnay')], (9, 13): [('dove-cote', 'oveday-otecay'),
('tea-table', 'eatay-abletay'), ('work-bags', 'orkway-agsbay')], (1, 4): [('i',
'iway')], (16, 19): [('incomprehensible', 'incomprehensibleway')], (8, 12):
[('to-night', 'otay-ightnay'), ('land-tax', 'andlay-axtay')], (11, 18): [('tete-
a-tete', 'etetay-away-etetay')], (5, 9): [('to-do', 'otay-oday')]})
Moved models to GPU!
Epoch:   0 | Flops (T): 0.007296461856, Tokens (M) 0.052678, Train Loss 3.077,
Val Loss 2.933 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   1 | Flops (T): 0.014592923712, Tokens (M) 0.105356, Train Loss 2.910,
Val Loss 2.903 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   2 | Flops (T): 0.021889385568, Tokens (M) 0.158034, Train Loss 2.893,
Val Loss 2.883 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   3 | Flops (T): 0.029185847424, Tokens (M) 0.210712, Train Loss 2.866,
Val Loss 2.867 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   4 | Flops (T): 0.03648230928, Tokens (M) 0.26339, Train Loss 2.846, Val
Loss 2.854 | Gen: oooooooooooooooooooo
Epoch:   5 | Flops (T): 0.043778771136, Tokens (M) 0.316068, Train Loss 2.830,
Val Loss 2.840 | Gen: oooooooooooooooooooo
Epoch:   6 | Flops (T): 0.051075232992, Tokens (M) 0.368746, Train Loss 2.814,
Val Loss 2.822 | Gen: oooooooooooooooooooo
Epoch:   7 | Flops (T): 0.058371694848, Tokens (M) 0.421424, Train Loss 2.793,
Val Loss 2.801 | Gen: oooooooooooooooooooo
Epoch:   8 | Flops (T): 0.065668156704, Tokens (M) 0.474102, Train Loss 2.729,
Val Loss 2.699 | Gen: oooooooooooooooooooo
Epoch:   9 | Flops (T): 0.07296461856, Tokens (M) 0.52678, Train Loss 2.580, Val
Loss 2.559 | Gen: oooooooooooooooooooe
Epoch:  10 | Flops (T): 0.080261080416, Tokens (M) 0.579458, Train Loss 2.426,
Val Loss 2.406 | Gen: oooooooooooooooooeee
Epoch:  11 | Flops (T): 0.087557542272, Tokens (M) 0.632136, Train Loss 2.334,
Val Loss 2.325 | Gen: oooooooooooooooeeeee
Epoch:  12 | Flops (T): 0.094854004128, Tokens (M) 0.684814, Train Loss 2.287,
Val Loss 2.371 | Gen: ooooooooeeeeeeeeeeee
Epoch:  13 | Flops (T): 0.102150465984, Tokens (M) 0.737492, Train Loss 2.260,
Val Loss 2.329 | Gen: oooooooooooooooooeee
Epoch:  14 | Flops (T): 0.10944692784, Tokens (M) 0.79017, Train Loss 2.190, Val
Loss 2.330 | Gen: oooooooooooooeeeeeee
Epoch:  15 | Flops (T): 0.116743389696, Tokens (M) 0.842848, Train Loss 2.151,
```

```
Val Loss 2.221 | Gen: ooooooooeeeeeeeeeee
Epoch:  16 | Flops (T): 0.124039851552, Tokens (M) 0.895526, Train Loss 2.092,
Val Loss 2.173 | Gen: oooooooooooeeeeeeeee
Epoch:  17 | Flops (T): 0.131336313408, Tokens (M) 0.948204, Train Loss 2.047,
Val Loss 2.135 | Gen: ooooooooooooeeeeeee
Epoch:  18 | Flops (T): 0.138632775264, Tokens (M) 1.000882, Train Loss 2.020,
Val Loss 2.158 | Gen: uuuuuuuuuuuuuuuuuuuu
Epoch:  19 | Flops (T): 0.14592923712, Tokens (M) 1.05356, Train Loss 1.981, Val
Loss 2.142 | Gen: uuuuuuuuuuuuuuuuuuuu
Epoch:  20 | Flops (T): 0.153225698976, Tokens (M) 1.106238, Train Loss 1.956,
Val Loss 2.037 | Gen: oooooooooooooeeeeaaa
Epoch:  21 | Flops (T): 0.160522160832, Tokens (M) 1.158916, Train Loss 1.893,
Val Loss 1.985 | Gen: oooooooooooooosssaaa
Epoch:  22 | Flops (T): 0.167818622688, Tokens (M) 1.211594, Train Loss 1.791,
Val Loss 2.054 | Gen: oooooooooooooosaaaaaa
Epoch:  23 | Flops (T): 0.175115084544, Tokens (M) 1.264272, Train Loss 1.870,
Val Loss 2.265 | Gen: ooooooonnaaaayyyyyyy
Epoch:  24 | Flops (T): 0.1824115464, Tokens (M) 1.31695, Train Loss 1.831, Val
Loss 1.770 | Gen: oooooooooooooooeaaaaa
Epoch:  25 | Flops (T): 0.189708008256, Tokens (M) 1.369628, Train Loss 1.693,
Val Loss 1.714 | Gen: oooooooooooooooeaaaaa
Epoch:  26 | Flops (T): 0.197004470112, Tokens (M) 1.422306, Train Loss 1.640,
Val Loss 1.683 | Gen: oooooooooooeaaaaaaaaa
Epoch:  27 | Flops (T): 0.204300931968, Tokens (M) 1.474984, Train Loss 1.610,
Val Loss 1.673 | Gen: oooooooooooooeaaaaaa
Epoch:  28 | Flops (T): 0.211597393824, Tokens (M) 1.527662, Train Loss 1.592,
Val Loss 1.659 | Gen: oooooooooooooooeaaaaa
Epoch:  29 | Flops (T): 0.21889385568, Tokens (M) 1.58034, Train Loss 1.587, Val
Loss 1.687 | Gen: oooooooooooooooeaaaa
Epoch:  30 | Flops (T): 0.226190317536, Tokens (M) 1.633018, Train Loss 1.570,
Val Loss 1.654 | Gen: oooooooooooooooeaaaa
Epoch:  31 | Flops (T): 0.233486779392, Tokens (M) 1.685696, Train Loss 1.562,
Val Loss 1.640 | Gen: oooooooooooooooooeaaa
Epoch:  32 | Flops (T): 0.240783241248, Tokens (M) 1.738374, Train Loss 1.462,
Val Loss 1.829 | Gen: oooooooooooooooooneaa
Epoch:  33 | Flops (T): 0.248079703104, Tokens (M) 1.791052, Train Loss 1.497,
Val Loss 1.578 | Gen: ooooooooooooooooootaa
Epoch:  34 | Flops (T): 0.25537616496, Tokens (M) 1.84373, Train Loss 1.355, Val
Loss 1.315 | Gen: oooooooooooooooooo-o-
Epoch:  35 | Flops (T): 0.262672626816, Tokens (M) 1.896408, Train Loss 1.335,
Val Loss 1.476 | Gen: --------------------
Epoch:  36 | Flops (T): 0.269969088672, Tokens (M) 1.949086, Train Loss 1.314,
Val Loss 1.893 | Gen: nnn-n---------------
Epoch:  37 | Flops (T): 0.277265550528, Tokens (M) 2.001764, Train Loss 1.401,
Val Loss 1.303 | Gen: oooooooooooooooo----
Epoch:  38 | Flops (T): 0.284562012384, Tokens (M) 2.054442, Train Loss 1.255,
Val Loss 1.260 | Gen: --------------------
Epoch:  39 | Flops (T): 0.29185847424, Tokens (M) 2.10712, Train Loss 1.243, Val
```

```
Loss 1.943 | Gen: --------------------
Epoch:  40 | Flops (T): 0.299154936096, Tokens (M) 2.159798, Train Loss 1.299,
Val Loss 1.262 | Gen: --------------------
Epoch:  41 | Flops (T): 0.306451397952, Tokens (M) 2.212476, Train Loss 1.182,
Val Loss 1.269 | Gen: --------------------
Epoch:  42 | Flops (T): 0.313747859808, Tokens (M) 2.265154, Train Loss 1.181,
Val Loss 1.212 | Gen: --------------------
Epoch:  43 | Flops (T): 0.321044321664, Tokens (M) 2.317832, Train Loss 1.149,
Val Loss 1.193 | Gen: --------------------
Epoch:  44 | Flops (T): 0.32834078352, Tokens (M) 2.37051, Train Loss 1.108, Val
Loss 1.166 | Gen: --------------------
Epoch:  45 | Flops (T): 0.335637245376, Tokens (M) 2.423188, Train Loss 1.086,
Val Loss 1.179 | Gen: --------------------
Epoch:  46 | Flops (T): 0.342933707232, Tokens (M) 2.475866, Train Loss 1.095,
Val Loss 1.255 | Gen: --------------------
Epoch:  47 | Flops (T): 0.350230169088, Tokens (M) 2.528544, Train Loss 1.027,
Val Loss 1.286 | Gen: --------------------
Epoch:  48 | Flops (T): 0.357526630944, Tokens (M) 2.581222, Train Loss 1.024,
Val Loss 1.273 | Gen: ---------------ttttt
Epoch:  49 | Flops (T): 0.3648230928, Tokens (M) 2.6339, Train Loss 0.955, Val
Loss 1.306 | Gen: --------------------
Epoch:  50 | Flops (T): 0.372119554656, Tokens (M) 2.686578, Train Loss 0.969,
Val Loss 1.273 | Gen: --------------------
Epoch:  51 | Flops (T): 0.379416016512, Tokens (M) 2.739256, Train Loss 0.944,
Val Loss 0.908 | Gen: --------------------
Epoch:  52 | Flops (T): 0.386712478368, Tokens (M) 2.791934, Train Loss 0.804,
Val Loss 0.897 | Gen: --------------------
Epoch:  53 | Flops (T): 0.394008940224, Tokens (M) 2.844612, Train Loss 0.758,
Val Loss 0.819 | Gen: --------------------
Epoch:  54 | Flops (T): 0.40130540208, Tokens (M) 2.89729, Train Loss 0.752, Val
Loss 1.149 | Gen: --------------------
Epoch:  55 | Flops (T): 0.408601863936, Tokens (M) 2.949968, Train Loss 0.761,
Val Loss 0.790 | Gen: --------------------
Epoch:  56 | Flops (T): 0.415898325792, Tokens (M) 3.002646, Train Loss 0.663,
Val Loss 0.777 | Gen: --------------------
Epoch:  57 | Flops (T): 0.423194787648, Tokens (M) 3.055324, Train Loss 0.617,
Val Loss 0.705 | Gen: nnnnnn-tddddddddddd
Epoch:  58 | Flops (T): 0.430491249504, Tokens (M) 3.108002, Train Loss 0.589,
Val Loss 0.692 | Gen: nnnnnnnnnnnnnnnnnnn
Epoch:  59 | Flops (T): 0.43778771136, Tokens (M) 3.16068, Train Loss 0.560, Val
Loss 0.697 | Gen: nnnnnnnnnnnn-dddddd
Epoch:  60 | Flops (T): 0.445084173216, Tokens (M) 3.213358, Train Loss 0.542,
Val Loss 0.776 | Gen: --------------------
Epoch:  61 | Flops (T): 0.452380635072, Tokens (M) 3.266036, Train Loss 0.530,
Val Loss 0.586 | Gen: nnnnnnnnnnnnnnnnnnn
Epoch:  62 | Flops (T): 0.459677096928, Tokens (M) 3.318714, Train Loss 0.489,
Val Loss 0.609 | Gen: --------------------
Epoch:  63 | Flops (T): 0.466973558784, Tokens (M) 3.371392, Train Loss 0.479,
```

```
Val Loss 0.599 | Gen: nnnnnnnnnnnnnnnnnnn
Epoch:  64 | Flops (T): 0.47427002064, Tokens (M) 3.42407, Train Loss 0.471, Val
Loss 0.642 | Gen: nnnnnnnnnnnnnnnnnnn
Epoch:  65 | Flops (T): 0.481566482496, Tokens (M) 3.476748, Train Loss 0.464,
Val Loss 0.522 | Gen: nnnnnnnnnnnnnnnnnnn
Epoch:  66 | Flops (T): 0.488862944352, Tokens (M) 3.529426, Train Loss 0.458,
Val Loss 0.612 | Gen: nnnnnnnnnnnnnnnnnnn
Epoch:  67 | Flops (T): 0.496159406208, Tokens (M) 3.582104, Train Loss 0.456,
Val Loss 0.590 | Gen: nnnnnnnnnnnnnnnnnnn
Epoch:  68 | Flops (T): 0.503455868064, Tokens (M) 3.634782, Train Loss 0.443,
Val Loss 0.485 | Gen: nnnnnnnnnnnnnnnnnnn
Epoch:  69 | Flops (T): 0.51075232992, Tokens (M) 3.68746, Train Loss 0.417, Val
Loss 0.454 | Gen: nnnnnnnnnnnnnnnnnnn
Epoch:  70 | Flops (T): 0.518048791776, Tokens (M) 3.740138, Train Loss 0.423,
Val Loss 0.588 | Gen: --------------------
Epoch:  71 | Flops (T): 0.525345253632, Tokens (M) 3.792816, Train Loss 0.450,
Val Loss 0.563 | Gen: --------------------
Epoch:  72 | Flops (T): 0.532641715488, Tokens (M) 3.845494, Train Loss 0.416,
Val Loss 0.435 | Gen: oooooooooooooooooooo
Epoch:  73 | Flops (T): 0.539938177344, Tokens (M) 3.898172, Train Loss 0.403,
Val Loss 0.406 | Gen: oooooooooonnnnnnnnn
Epoch:  74 | Flops (T): 0.5472346392, Tokens (M) 3.95085, Train Loss 0.386, Val
Loss 0.423 | Gen: oooooooooooooooooooo
Epoch:  75 | Flops (T): 0.554531101056, Tokens (M) 4.003528, Train Loss 0.369,
Val Loss 0.385 | Gen: oooooooooooooooooooon
Epoch:  76 | Flops (T): 0.561827562912, Tokens (M) 4.056206, Train Loss 0.367,
Val Loss 0.457 | Gen: nnnnnnnnnn-daaaaaaaa
Epoch:  77 | Flops (T): 0.569124024768, Tokens (M) 4.108884, Train Loss 0.366,
Val Loss 0.382 | Gen: oooooooooooooooooonn
Epoch:  78 | Flops (T): 0.576420486624, Tokens (M) 4.161562, Train Loss 0.348,
Val Loss 0.401 | Gen: oooooooooooooooooooon
Epoch:  79 | Flops (T): 0.58371694848, Tokens (M) 4.21424, Train Loss 0.357, Val
Loss 0.556 | Gen: oooooooooooooooooooo
Epoch:  80 | Flops (T): 0.591013410336, Tokens (M) 4.266918, Train Loss 0.361,
Val Loss 0.491 | Gen: oooooooooooooooooooo
Epoch:  81 | Flops (T): 0.598309872192, Tokens (M) 4.319596, Train Loss 0.357,
Val Loss 0.406 | Gen: oooooooooooooooooonn
Epoch:  82 | Flops (T): 0.605606334048, Tokens (M) 4.372274, Train Loss 0.344,
Val Loss 0.373 | Gen: ooooooonnnnnnnnnnnn-
Epoch:  83 | Flops (T): 0.612902795904, Tokens (M) 4.424952, Train Loss 0.332,
Val Loss 0.323 | Gen: oooooooooooooooooonn
Epoch:  84 | Flops (T): 0.62019925776, Tokens (M) 4.47763, Train Loss 0.315, Val
Loss 0.299 | Gen: oooooooooooooooooon-
Epoch:  85 | Flops (T): 0.627495719616, Tokens (M) 4.530308, Train Loss 0.296,
Val Loss 0.304 | Gen: oooooooooooooooooon-
Epoch:  86 | Flops (T): 0.634792181472, Tokens (M) 4.582986, Train Loss 0.289,
Val Loss 0.291 | Gen: oooooooooooooooooon-
Epoch:  87 | Flops (T): 0.642088643328, Tokens (M) 4.635664, Train Loss 0.285,
```

```
Val Loss 0.297 | Gen: ooooooooooooooooon-
Epoch:  88 | Flops (T): 0.649385105184, Tokens (M) 4.688342, Train Loss 0.274,
Val Loss 0.359 | Gen: oooooooooooooooon-t
Epoch:  89 | Flops (T): 0.65668156704, Tokens (M) 4.74102, Train Loss 0.281, Val
Loss 0.278 | Gen: oooooooooooooooon-t
Epoch:  90 | Flops (T): 0.663978028896, Tokens (M) 4.793698, Train Loss 0.255,
Val Loss 0.265 | Gen: oooooooooooooooon-t
Epoch:  91 | Flops (T): 0.671274490752, Tokens (M) 4.846376, Train Loss 0.244,
Val Loss 0.263 | Gen: oooooooooooooooon-
Epoch:  92 | Flops (T): 0.678570952608, Tokens (M) 4.899054, Train Loss 0.249,
Val Loss 0.282 | Gen: oooooooooooooooon-
Epoch:  93 | Flops (T): 0.685867414464, Tokens (M) 4.951732, Train Loss 0.236,
Val Loss 0.256 | Gen: oooooooooooooooon-t
Epoch:  94 | Flops (T): 0.69316387632, Tokens (M) 5.00441, Train Loss 0.229, Val
Loss 0.251 | Gen: oooooooooooooooon-
Epoch:  95 | Flops (T): 0.700460338176, Tokens (M) 5.057088, Train Loss 0.223,
Val Loss 0.260 | Gen: oooooooooooooooon-
Epoch:  96 | Flops (T): 0.707756800032, Tokens (M) 5.109766, Train Loss 0.268,
Val Loss 0.297 | Gen: oooooooooooooooon-
Epoch:  97 | Flops (T): 0.715053261888, Tokens (M) 5.162444, Train Loss 0.277,
Val Loss 0.359 | Gen: oooooooooooooooont
Epoch:  98 | Flops (T): 0.722349723744, Tokens (M) 5.215122, Train Loss 0.341,
Val Loss 0.261 | Gen: ooooooooooooon-tdddd
Epoch:  99 | Flops (T): 0.7296461856, Tokens (M) 5.2678, Train Loss 0.238, Val
Loss 0.528 | Gen: oooooooooooooooon--
Obtained lowest validation loss of: 0.2514265688477344
Avg tokens per example (Seq Len): 20.59343236903831, Avg flops per example:
2852408.8569194684
```

[ ]:
```
TEST_SENTENCE = "floor"
trans48_args_s = AttrDict()
args_dict_48 = {
    "data_file_name": "pig_latin_small", #pig_latin_small debug
    "cuda": use_cuda,
    "nepochs": 100,
    "checkpoint_dir": "checkpoints",
    "learning_rate": 5e-4,
    "early_stopping_patience": 100,
    "lr_decay": 0.99,
    "batch_size": 64,
    "hidden_size": 48,
    "num_transformer_layers": 4,
    "loss_seq_span": "target_only",
    "num_layers": 4,
    "vocab_size": 30
}
trans48_args_s.update(args_dict_48)
```

```
print_opts(trans48_args_s)
trans48_decoder_s, trans48_losses_s, idx_dict = train(trans48_args_s)
```

```
================================================================================
                                       Opts
--------------------------------------------------------------------------------
                         data_file_name: pig_latin_small
                                   cuda: 1
                                 nepochs: 100
                         checkpoint_dir: checkpoints
                          learning_rate: 0.0005
                early_stopping_patience: 100
                                lr_decay: 0.99
                             batch_size: 64
                            hidden_size: 48
                 num_transformer_layers: 4
                          loss_seq_span: target_only
                             num_layers: 4
                             vocab_size: 30
================================================================================
================================================================================
                                     Data Stats
--------------------------------------------------------------------------------
('irresistible', 'irresistibleway')
('repentance', 'epentanceray')
('pretty', 'ettypray')
('parcels', 'arcelspay')
('survey', 'urveysay')
Num unique word pairs: 3198
Vocabulary: dict_keys(['-', 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j',
'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z',
'SOS', 'EOS', 'EOP'])
Vocab size: 30
================================================================================
defaultdict(<class 'list'>, {(12, 15): [('irresistible', 'irresistibleway'),
('advantageous', 'advantageousway'), ('impartiality', 'impartialityway'),
('interruption', 'interruptionway'), ('anticipation', 'anticipationway'),
('impenetrable', 'impenetrableway'), ('excellencies', 'excellenciesway'),
('intermission', 'intermissionway'), ('interspersed', 'interspersedway'),
('unaffectedly', 'unaffectedlyway'), ('insinuations', 'insinuationsway'),
('unpardonable', 'unpardonableway'), ('arrangements', 'arrangementsway'),
('unexpectedly', 'unexpectedlyway'), ('introduction', 'introductionway'),
('occasionally', 'occasionallyway'), ('establishing', 'establishingway'),
('unanswerable', 'unanswerableway'), ('independence', 'independenceway'),
('inclinations', 'inclinationsway'), ('instructions', 'instructionsway'),
('undoubtingly', 'undoubtinglyway'), ('unfathomable', 'unfathomableway'),
('impoverished', 'impoverishedway'), ('outstretched', 'outstretchedway'),
```

```
('unrestrained', 'unrestrainedway'), ('ungraciously', 'ungraciouslyway'),
('imperfection', 'imperfectionway'), ('unreasonably', 'unreasonablyway'),
('interference', 'interferenceway'), ('inconvenient', 'inconvenientway'),
('enthusiastic', 'enthusiasticway'), ('extinguished', 'extinguishedway'),
('entanglement', 'entanglementway'), ('affectionate', 'affectionateway')], (10,
12): [('repentance', 'epentanceray'), ('capability', 'apabilitycay'),
('beneficial', 'eneficialbay'), ('retrenched', 'etrenchedray'), ('devonshire',
'evonshireday'), ('palanquins', 'alanquinspay'), ('contribute', 'ontributecay'),
('minuteness', 'inutenessmay'), ('dissimilar', 'issimilarday'), ('flattering',
'atteringflay'), ('confounded', 'onfoundedcay'), ('compromise', 'ompromisecay'),
('contrasted', 'ontrastedcay'), ('precaution', 'ecautionpray'), ('fastidious',
'astidiousfay'), ('blackguard', 'ackguardblay'), ('heroically', 'eroicallyhay'),
('lengthened', 'engthenedlay'), ('pocketbook', 'ocketbookpay'), ('companions',
'ompanionscay'), ('displeased', 'ispleasedday'), ('pianoforte', 'ianofortepay'),
('succession', 'uccessionsay'), ('composedly', 'omposedlycay'), ('pronounced',
'onouncedpray'), ('honourable', 'onourablehay'), ('solicitude', 'olicitudesay'),
('justifying', 'ustifyingjay'), ('collecting', 'ollectingcay'), ('headstrong',
'eadstronghay'), ('forbidding', 'orbiddingfay'), ('successful', 'uccessfulsay'),
('confidence', 'onfidencecay'), ('prescribed', 'escribedpray'), ('scrutinies',
'utiniesscray'), ('melancholy', 'elancholymay'), ('handsomest', 'andsomesthay'),
('prescience', 'esciencepray'), ('conclusion', 'onclusioncay'), ('remarkably',
'emarkablyray'), ('gratefully', 'atefullygray'), ('michaelmas', 'ichaelmasmay'),
('containing', 'ontainingcay'), ('discussion', 'iscussionday'), ('resentment',
'esentmentray'), ('restraints', 'estraintsray'), ('tablecloth', 'ableclothtay'),
('projecting', 'ojectingpray'), ('frightened', 'ightenedfray'), ('pardonable',
'ardonablepay'), ('variations', 'ariationsvay'), ('sensations', 'ensationssay'),
('rheumatism', 'eumatismrhay'), ('dispirited', 'ispiritedday'), ('politeness',
'olitenesspay'), ('suggesting', 'uggestingsay'), ('willoughby', 'illoughbyway'),
('reasonable', 'easonableray'), ('discerning', 'iscerningday'), ('previously',
'eviouslypray'), ('passionate', 'assionatepay'), ('prophecies', 'opheciespray'),
('possessing', 'ossessingpay'), ('perfection', 'erfectionpay'), ('heightened',
'eightenedhay'), ('generously', 'enerouslygay'), ('faithfully', 'aithfullyfay'),
('reconciled', 'econciledray'), ('distressed', 'istressedday'), ('respective',
'espectiveray'), ('liberality', 'iberalitylay'), ('disengaged', 'isengagedday'),
('honourably', 'onourablyhay'), ('persuading', 'ersuadingpay'), ('cultivated',
'ultivatedcay'), ('projection', 'ojectionpray'), ('disapprove', 'isapproveday'),
('conciliate', 'onciliatecay'), ('satisfying', 'atisfyingsay'), ('dispatches',
'ispatchesday'), ('repeatedly', 'epeatedlyray'), ('suspicions', 'uspicionssay'),
('perplexity', 'erplexitypay'), ('strengthen', 'engthenstray'), ('dependence',
'ependenceday'), ('dissipated', 'issipatedday'), ('conformity', 'onformitycay'),
('hysterical', 'ystericalhay'), ('throughout', 'oughoutthray'), ('sacrificed',
'acrificedsay'), ('dimensions', 'imensionsday'), ('flatteries', 'atteriesflay'),
('completely', 'ompletelycay'), ('reproaches', 'eproachesray'), ('contriving',
'ontrivingcay'), ('consequent', 'onsequentcay'), ('liveliness', 'ivelinesslay'),
('resistance', 'esistanceray'), ('caricature', 'aricaturecay'), ('stimulated',
'imulatedstay'), ('brightened', 'ightenedbray'), ('prevailing', 'evailingpray'),
('disclosure', 'isclosureday'), ('bequeathed', 'equeathedbay'), ('disappoint',
'isappointday'), ('complexion', 'omplexioncay'), ('dismounted', 'ismountedday'),
```

('remembered', 'ememberedray'), ('delineated', 'elineatedday'), ('subsequent',
'ubsequentsay'), ('persevered', 'erseveredpay'), ('doubtingly', 'oubtinglyday'),
('characters', 'aracterschay'), ('neighbours', 'eighboursnay'), ('witticisms',
'itticismsway'), ('materially', 'ateriallymay'), ('playthings', 'aythingsplay'),
('suspecting', 'uspectingsay'), ('conditions', 'onditionscay'), ('discovered',
'iscoveredday'), ('deficiency', 'eficiencyday'), ('regularity', 'egularityray'),
('subjection', 'ubjectionsay'), ('brightness', 'ightnessbray'), ('pleasantly',
'easantlyplay'), ('delicacies', 'elicaciesday'), ('preference', 'eferencepray'),
('difference', 'ifferenceday'), ('compassion', 'ompassioncay'), ('competence',
'ompetencecay'), ('generosity', 'enerositygay'), ('retirement', 'etirementray'),
('strikingly', 'ikinglystray'), ('subsisting', 'ubsistingsay'), ('conscience',
'onsciencecay'), ('diabolical', 'iabolicalday'), ('temptation', 'emptationtay'),
('formidable', 'ormidablefay'), ('reconsider', 'econsiderray'), ('prejudiced',
'ejudicedpray'), ('comprehend', 'omprehendcay'), ('compliment', 'omplimentcay'),
('sufferings', 'ufferingssay'), ('concluding', 'oncludingcay'), ('compliance',
'ompliancecay'), ('mosquitoes', 'osquitoesmay'), ('thoughtful', 'oughtfulthay'),
('bitterness', 'itternessbay'), ('mentioning', 'entioningmay'), ('toleration',
'olerationtay'), ('profession', 'ofessionpray'), ('discompose', 'iscomposeday'),
('waistcoats', 'aistcoatsway'), ('reflecting', 'eflectingray'), ('positively',
'ositivelypay'), ('diminished', 'iminishedday')], (6, 8): [('pretty',
'ettypray'), ('survey', 'urveysay'), ('demure', 'emureday'), ('secure',
'ecuresay'), ('sobbed', 'obbedsay'), ('moment', 'omentmay'), ('guilty',
'uiltygay'), ('combat', 'ombatcay'), ('modern', 'odernmay'), ('father',
'atherfay'), ('horses', 'orseshay'), ('closer', 'oserclay'), ('nicety',
'icetynay'), ('shaken', 'akenshay'), ('centre', 'entrecay'), ('powers',
'owerspay'), ('simple', 'implesay'), ('beings', 'eingsbay'), ('recent',
'ecentray'), ('sketch', 'etchskay'), ('loving', 'ovinglay'), ('dagger',
'aggerday'), ('hearts', 'eartshay'), ('copied', 'opiedcay'), ('gazing',
'azinggay'), ('sooner', 'oonersay'), ('disarm', 'isarmday'), ('suited',
'uitedsay'), ('doubts', 'oubtsday'), ('lifted', 'iftedlay'), ('vacant',
'acantvay'), ('palmer', 'almerpay'), ('grudge', 'udgegray'), ('folded',
'oldedfay'), ('remove', 'emoveray'), ('borrow', 'orrowbay'), ('scheme',
'emeschay'), ('solemn', 'olemnsay'), ('denote', 'enoteday'), ('bustle',
'ustlebay'), ('turned', 'urnedtay'), ('mutton', 'uttonmay'), ('tender',
'endertay'), ('hearer', 'earerhay'), ('begins', 'eginsbay'), ('taught',
'aughttay'), ('honour', 'onourhay'), ('robert', 'obertray'), ('gibson',
'ibsongay'), ('virtue', 'irtuevay'), ('vanish', 'anishvay'), ('writes',
'iteswray'), ('silent', 'ilentsay'), ('singly', 'inglysay'), ('jargon',
'argonjay'), ('grieve', 'ievegray'), ('design', 'esignday'), ('linger',
'ingerlay'), ('bathed', 'athedbay'), ('pardon', 'ardonpay'), ('rumour',
'umourray'), ('dances', 'ancesday'), ('remark', 'emarkray'), ('worded',
'ordedway'), ('honest', 'onesthay'), ('chosen', 'osenchay'), ('boyish',
'oyishbay'), ('stress', 'essstray'), ('caught', 'aughtcay'), ('potent',
'otentpay'), ('during', 'uringday'), ('stocks', 'ocksstay'), ('relics',
'elicsray'), ('tongue', 'onguetay'), ('street', 'eetstray'), ('whiled',
'iledwhay'), ('manner', 'annermay'), ('truths', 'uthstray'), ('travel',
'aveltray'), ('couple', 'ouplecay'), ('gaiety', 'aietygay'), ('giving',
'ivinggay'), ('county', 'ountycay'), ('cramps', 'ampscray'), ('wealth',

'ealthway'), ('divine', 'ivineday'), ('behold', 'eholdbay'), ('feared',
'earedfay'), ('sparks', 'arksspay'), ('finger', 'ingerfay'), ('talent',
'alenttay'), ('demand', 'emandday'), ('coffee', 'offeecay'), ('merest',
'erestmay'), ('detail', 'etailday'), ('casino', 'asinocay'), ('choose',
'oosechay'), ('warmed', 'armedway'), ('relate', 'elateray'), ('sashes',
'ashessay'), ('chance', 'ancechay'), ('rushed', 'ushedray'), ('health',
'ealthhay'), ('preyed', 'eyedpray'), ('mutual', 'utualmay'), ('months',
'onthsmay'), ('raises', 'aisesray'), ('things', 'ingsthay'), ('master',
'astermay'), ('thanks', 'anksthay'), ('cheeks', 'eekschay'), ('hazard',
'azardhay'), ('slight', 'ightslay'), ('guests', 'uestsgay'), ('halves',
'alveshay'), ('lessen', 'essenlay'), ('notice', 'oticenay'), ('higher',
'igherhay'), ('truest', 'uesttray'), ('porter', 'orterpay'), ('fickle',
'icklefay'), ('behave', 'ehavebay'), ('credit', 'editcray'), ('number',
'umbernay'), ('mother', 'othermay'), ('tricks', 'ickstray'), ('traced',
'acedtray'), ('subdue', 'ubduesay'), ('ruined', 'uinedray'), ('famous',
'amousfay'), ('danced', 'ancedday'), ('clever', 'everclay'), ('drives',
'ivesdray'), ('resist', 'esistray'), ('forget', 'orgetfay'), ('stable',
'ablestay'), ('bigger', 'iggerbay'), ('betray', 'etraybay'), ('senses',
'ensessay'), ('smiled', 'iledsmay'), ('hating', 'atinghay'), ('bolder',
'olderbay'), ('banish', 'anishbay'), ('remain', 'emainray'), ('period',
'eriodpay'), ('gossip', 'ossipgay'), ('basket', 'asketbay'), ('genius',
'eniusgay'), ('seized', 'eizedsay'), ('widely', 'idelyway'), ('pursue',
'ursuepay'), ('favour', 'avourfay'), ('sighed', 'ighedsay'), ('plants',
'antsplay'), ('proved', 'ovedpray'), ('season', 'easonsay'), ('circle',
'irclecay'), ('taking', 'akingtay'), ('screen', 'eenscray'), ('talker',
'alkertay'), ('teazed', 'eazedtay'), ('suffer', 'uffersay'), ('curate',
'uratecay'), ('traits', 'aitstray'), ('defect', 'efectday'), ('called',
'alledcay'), ('dining', 'iningday'), ('direct', 'irectday'), ('dinner',
'innerday'), ('joking', 'okingjay'), ('whiter', 'iterwhay'), ('limits',
'imitslay'), ('papers', 'aperspay'), ('public', 'ublicpay'), ('losing',
'osinglay'), ('handed', 'andedhay'), ('wisest', 'isestway'), ('delays',
'elaysday'), ('leaves', 'eaveslay'), ('points', 'ointspay'), ('hither',
'itherhay'), ('horrid', 'orridhay'), ('middle', 'iddlemay'), ('vastly',
'astlyvay'), ('luxury', 'uxurylay'), ('failed', 'ailedfay'), ('lodges',
'odgeslay'), ('female', 'emalefay'), ('walked', 'alkedway'), ('nipped',
'ippednay'), ('nephew', 'ephewnay'), ('settle', 'ettlesay'), ('kinder',
'inderkay'), ('bowing', 'owingbay'), ('smooth', 'oothsmay'), ('roared',
'oaredray'), ('wiping', 'ipingway'), ('jewels', 'ewelsjay'), ('breath',
'eathbray'), ('steady', 'eadystay'), ('breach', 'eachbray'), ('polite',
'olitepay'), ('colder', 'oldercay'), ('though', 'oughthay'), ('poured',
'ouredpay'), ('paused', 'ausedpay'), ('nought', 'oughtnay'), ('shrewd',
'ewdshray'), ('fourth', 'ourthfay'), ('parish', 'arishpay'), ('traded',
'adedtray'), ('glance', 'anceglay'), ('lesser', 'esserlay'), ('curacy',
'uracycay'), ('labour', 'abourlay'), ('became', 'ecamebay'), ('decide',
'ecideday'), ('kisses', 'isseskay'), ('styled', 'yledstay'), ('sprung',
'ungspray'), ('priory', 'iorypray'), ('derive', 'eriveday'), ('school',
'oolschay'), ('liking', 'ikinglay'), ('dawned', 'awnedday'), ('weaken',
'eakenway'), ('winter', 'interway'), ('rivals', 'ivalsray'), ('sallow',

```
'allowsay'), ('defend', 'efendday'), ('seated', 'eatedsay'), ('penury',
'enurypay'), ('repair', 'epairray'), ('chaise', 'aisechay'), ('church',
'urchchay'), ('beyond', 'eyondbay'), ('wisdom', 'isdomway'), ('retain',
'etainray'), ('toilet', 'oilettay'), ('talked', 'alkedtay'), ('dearly',
'earlyday'), ('reward', 'ewardray'), ('safely', 'afelysay'), ('making',
'akingmay'), ('debate', 'ebateday'), ('future', 'uturefay'), ('broken',
'okenbray'), ('begged', 'eggedbay'), ('thirty', 'irtythay'), ('muslin',
'uslinmay'), ('marked', 'arkedmay'), ('hinder', 'inderhay'), ('beheld',
'eheldbay'), ('maxims', 'aximsmay'), ('theirs', 'eirsthay'), ('corner',
'ornercay'), ('piqued', 'iquedpay'), ('rested', 'estedray'), ('needed',
'eedednay'), ('fitted', 'ittedfay'), ('candid', 'andidcay'), ('laughs',
'aughslay'), ('window', 'indowway'), ('tastes', 'astestay'), ('spread',
'eadspray'), ('hollow', 'ollowhay'), ('sorrow', 'orrowsay'), ('partly',
'artlypay'), ('served', 'ervedsay'), ('flowed', 'owedflay'), ('remedy',
'emedyray'), ('trying', 'yingtray'), ('latest', 'atestlay'), ('manage',
'anagemay'), ('gained', 'ainedgay'), ('plenty', 'entyplay'), ('garden',
'ardengay'), ('frosts', 'ostsfray'), ('thirds', 'irdsthay'), ('handle',
'andlehay'), ('verbal', 'erbalvay'), ('sheath', 'eathshay'), ('covert',
'overtcay'), ('darker', 'arkerday'), ('harris', 'arrishay'), ('houses',
'ouseshay'), ('nature', 'aturenay'), ('thomas', 'omasthay'), ('submit',
'ubmitsay'), ('danger', 'angerday'), ('lawyer', 'awyerlay'), ('matter',
'attermay'), ('rugged', 'uggedray'), ('highly', 'ighlyhay'), ('whence',
'encewhay'), ('martha', 'arthamay'), ('hearty', 'eartyhay'), ('recall',
'ecallray'), ('raised', 'aisedray'), ('follow', 'ollowfay'), ('nearer',
'earernay'), ('gentle', 'entlegay'), ('pitied', 'itiedpay'), ('nurses',
'ursesnay'), ('punish', 'unishpay')], (7, 9): [('parcels', 'arcelspay'),
('welcome', 'elcomeway'), ('posture', 'osturepay'), ('mothers', 'othersmay'),
('tempers', 'emperstay'), ('carries', 'arriescay'), ('neglect', 'eglectnay'),
('prevail', 'evailpray'), ('declare', 'eclareday'), ('crowned', 'ownedcray'),
('mindful', 'indfulmay'), ('blessed', 'essedblay'), ('kitchen', 'itchenkay'),
('younger', 'youngeray'), ('hanover', 'anoverhay'), ('lowered', 'oweredlay'),
('summons', 'ummonssay'), ('suffice', 'ufficesay'), ('rousing', 'ousingray'),
('feeling', 'eelingfay'), ('parlors', 'arlorspay'), ('gravely', 'avelygray'),
('connect', 'onnectcay'), ('species', 'eciesspay'), ('revived', 'evivedray'),
('tribute', 'ibutetray'), ('pitying', 'ityingpay'), ('brother', 'otherbray'),
('retired', 'etiredray'), ('suffers', 'ufferssay'), ('spurned', 'urnedspay'),
('dropped', 'oppeddray'), ('galling', 'allinggay'), ('seducer', 'educersay'),
('foibles', 'oiblesfay'), ('tuesday', 'uesdaytay'), ('patient', 'atientpay'),
('worsted', 'orstedway'), ('trifled', 'ifledtray'), ('waiting', 'aitingway'),
('widower', 'idowerway'), ('hardily', 'ardilyhay'), ('glimpse', 'impseglay'),
('whether', 'etherwhay'), ('reverie', 'everieray'), ('resolve', 'esolveray'),
('propose', 'oposepray'), ('tallest', 'allesttay'), ('nominal', 'ominalnay'),
('languid', 'anguidlay'), ('wronged', 'ongedwray'), ('nothing', 'othingnay'),
('promote', 'omotepray'), ('novelty', 'oveltynay'), ('dooming', 'oomingday'),
('display', 'isplayday'), ('correct', 'orrectcay'), ('feigned', 'eignedfay'),
('schemes', 'emesschay'), ('deepest', 'eepestday'), ('flannel', 'annelflay'),
('hauteur', 'auteurhay'), ('sharing', 'aringshay'), ('getting', 'ettinggay'),
('screams', 'eamsscray'), ('variety', 'arietyvay'), ('yielded', 'yieldeday'),
```

```
('brought', 'oughtbray'), ('haughty', 'aughtyhay'), ('bristol', 'istolbray'),
('silence', 'ilencesay'), ('staying', 'ayingstay'), ('confess', 'onfesscay'),
('placing', 'acingplay'), ('noisier', 'oisiernay'), ('nursing', 'ursingnay'),
('tuition', 'uitiontay'), ('welfare', 'elfareway'), ('hunters', 'untershay'),
('poplars', 'oplarspay'), ('deprive', 'epriveday'), ('mansion', 'ansionmay'),
('reasons', 'easonsray'), ('conceit', 'onceitcay'), ('burgess', 'urgessbay'),
('receipt', 'eceiptray'), ('befalls', 'efallsbay'), ('fetches', 'etchesfay'),
('ranging', 'angingray'), ('passion', 'assionpay'), ('smirked', 'irkedsmay'),
('honours', 'onourshay'), ('covered', 'overedcay'), ('dancing', 'ancingday'),
('hundred', 'undredhay'), ('forgave', 'orgavefay'), ('fifteen', 'ifteenfay'),
('lasting', 'astinglay'), ('minuter', 'inutermay'), ('tremour', 'emourtray'),
('passing', 'assingpay'), ('streets', 'eetsstray'), ('resists', 'esistsray'),
('cleared', 'earedclay'), ('miracle', 'iraclemay'), ('several', 'everalsay'),
('sighing', 'ighingsay'), ('resumed', 'esumedray'), ('merited', 'eritedmay'),
('without', 'ithoutway'), ('removed', 'emovedray'), ('behaved', 'ehavedbay'),
('succour', 'uccoursay'), ('bribing', 'ibingbray'), ('charity', 'aritychay'),
('shortly', 'ortlyshay'), ('tempted', 'emptedtay'), ('briefly', 'ieflybray'),
('parties', 'artiespay'), ('touched', 'ouchedtay'), ('content', 'ontentcay'),
('proceed', 'oceedpray'), ('windows', 'indowsway'), ('ribbons', 'ibbonsray'),
('stepped', 'eppedstay'), ('pursuit', 'ursuitpay'), ('quarter', 'arterquay'),
('prudent', 'udentpray'), ('dignity', 'ignityday'), ('payment', 'aymentpay'),
('grounds', 'oundsgray'), ('bequest', 'equestbay'), ('donavan', 'onavanday'),
('renewed', 'enewedray'), ('smaller', 'allersmay'), ('fearful', 'earfulfay'),
('civilly', 'ivillycay'), ('romance', 'omanceray'), ('females', 'emalesfay'),
('grieved', 'ievedgray'), ('created', 'eatedcray'), ('command', 'ommandcay'),
('defined', 'efinedday'), ('bluntly', 'untlyblay'), ('recital', 'ecitalray'),
('sorrows', 'orrowssay'), ('whither', 'itherwhay'), ('spotted', 'ottedspay'),
('justify', 'ustifyjay'), ('conduit', 'onduitcay'), ('revenge', 'evengeray'),
('scenery', 'eneryscay'), ('started', 'artedstay'), ('degrees', 'egreesday'),
('whisper', 'isperwhay'), ('strange', 'angestray'), ('bidding', 'iddingbay'),
('musical', 'usicalmay'), ('happens', 'appenshay'), ('compare', 'omparecay'),
('presume', 'esumepray'), ('sitting', 'ittingsay'), ('regards', 'egardsray'),
('violins', 'iolinsvay'), ('rightly', 'ightlyray'), ('memento', 'ementomay'),
('courted', 'ourtedcay'), ('servant', 'ervantsay'), ('possess', 'ossesspay'),
('herself', 'erselfhay'), ('present', 'esentpray'), ('natural', 'aturalnay'),
('managed', 'anagedmay'), ('hostess', 'ostesshay'), ('leaving', 'eavinglay'),
('mention', 'entionmay'), ('circuit', 'ircuitcay'), ('counsel', 'ounselcay'),
('guineas', 'uineasgay'), ('solidly', 'olidlysay'), ('dashing', 'ashingday'),
('station', 'ationstay'), ('written', 'ittenwray'), ('writing', 'itingwray'),
('country', 'ountrycay'), ('delight', 'elightday'), ('westons', 'estonsway'),
('debated', 'ebatedday'), ('lengths', 'engthslay'), ('torment', 'ormenttay'),
('reverse', 'everseray'), ('bedroom', 'edroombay'), ('forbade', 'orbadefay'),
('comings', 'omingscay'), ('tonight', 'onighttay'), ('similar', 'imilarsay'),
('calling', 'allingcay'), ('cassino', 'assinocay'), ('strains', 'ainsstray'),
('scratch', 'atchscray'), ('gardens', 'ardensgay'), ('freedom', 'eedomfray'),
('compact', 'ompactcay'), ('message', 'essagemay'), ('heavily', 'eavilyhay'),
('manners', 'annersmay'), ('systems', 'ystemssay'), ('forbore', 'orborefay'),
('wrapped', 'appedwray'), ('defects', 'efectsday'), ('villain', 'illainvay'),
```

('studied', 'udiedstay'), ('playing', 'ayingplay'), ('diction', 'ictionday'),
('tending', 'endingtay'), ('seeking', 'eekingsay'), ('rapture', 'aptureray'),
('dawdled', 'awdledday'), ('wearing', 'earingway'), ('comfort', 'omfortcay'),
('finally', 'inallyfay'), ('circles', 'irclescay'), ('putting', 'uttingpay'),
('twisted', 'istedtway'), ('destroy', 'estroyday'), ('whereas', 'ereaswhay'),
('dismiss', 'ismissday'), ('visibly', 'isiblyvay'), ('talents', 'alentstay'),
('finding', 'indingfay'), ('removes', 'emovesray'), ('replied', 'epliedray'),
('bluster', 'usterblay'), ('raising', 'aisingray'), ('plainly', 'ainlyplay'),
('worried', 'orriedway'), ('meaning', 'eaningmay'), ('blushes', 'ushesblay'),
('relapse', 'elapseray'), ('careful', 'arefulcay'), ('beloved', 'elovedbay'),
('degrade', 'egradeday'), ('foresaw', 'oresawfay'), ('procure', 'ocurepray'),
('kindest', 'indestkay'), ('secures', 'ecuressay'), ('hanging', 'anginghay'),
('persons', 'ersonspay'), ('taverns', 'avernstay'), ('walking', 'alkingway'),
('quicker', 'ickerquay'), ('guessed', 'uessedgay'), ('confirm', 'onfirmcay'),
('venture', 'enturevay'), ('project', 'ojectpray'), ('prayers', 'ayerspray'),
('heights', 'eightshay'), ('subject', 'ubjectsay'), ('summits', 'ummitssay'),
('towards', 'owardstay'), ('closing', 'osingclay'), ('journey', 'ourneyjay'),
('pointer', 'ointerpay'), ('chained', 'ainedchay'), ('clothes', 'othesclay'),
('require', 'equireray'), ('chusing', 'usingchay'), ('reproof', 'eproofray'),
('reckons', 'eckonsray'), ('pressed', 'essedpray'), ('talking', 'alkingtay'),
('harbour', 'arbourhay'), ('garrets', 'arretsgay'), ('foresee', 'oreseefay'),
('lustily', 'ustilylay'), ('leagued', 'eaguedlay'), ('judging', 'udgingjay'),
('rejoice', 'ejoiceray'), ('honiton', 'onitonhay'), ('shorten', 'ortenshay'),
('stating', 'atingstay'), ('privacy', 'ivacypray'), ('steeles', 'eelesstay'),
('thickly', 'icklythay'), ('shocked', 'ockedshay'), ('limited', 'imitedlay'),
('blights', 'ightsblay'), ('seclude', 'ecludesay'), ('cordial', 'ordialcay'),
('request', 'equestray'), ('denoted', 'enotedday'), ('darling', 'arlingday'),
('coxcomb', 'oxcombcay'), ('however', 'oweverhay'), ('perform', 'erformpay'),
('society', 'ocietysay'), ('deserts', 'esertsday'), ('palmers', 'almerspay'),
('manager', 'anagermay'), ('rupture', 'uptureray'), ('concise', 'oncisecay'),
('scolded', 'oldedscay'), ('seduced', 'educedsay'), ('doubted', 'oubtedday'),
('forward', 'orwardfay'), ('leisure', 'eisurelay'), ('robbing', 'obbingray'),
('horizon', 'orizonhay'), ('blunder', 'underblay'), ('flushed', 'ushedflay'),
('divided', 'ividedday'), ('charged', 'argedchay'), ('disturb', 'isturbday'),
('smitten', 'ittensmay'), ('seconds', 'econdssay'), ('letters', 'etterslay'),
('provide', 'ovidepray'), ('conduct', 'onductcay'), ('plagued', 'aguedplay'),
('stretch', 'etchstray'), ('hoarded', 'oardedhay'), ('blushed', 'ushedblay'),
('nodding', 'oddingnay'), ('mankind', 'ankindmay'), ('caprice', 'apricecay'),
('quitted', 'ittedquay')], (5, 7): [('china', 'inachay'), ('deter', 'eterday'),
('woman', 'omanway'), ('clock', 'ockclay'), ('borne', 'ornebay'), ('noise',
'oisenay'), ('midst', 'idstmay'), ('dwell', 'elldway'), ('smith', 'ithsmay'),
('gowns', 'ownsgay'), ('named', 'amednay'), ('glass', 'assglay'), ('price',
'icepray'), ('close', 'oseclay'), ('guide', 'uidegay'), ('newly', 'ewlynay'),
('sweep', 'eepsway'), ('swept', 'eptsway'), ('those', 'osethay'), ('lower',
'owerlay'), ('plump', 'umpplay'), ('downs', 'ownsday'), ('purse', 'ursepay'),
('linen', 'inenlay'), ('later', 'aterlay'), ('broad', 'oadbray'), ('miles',
'ilesmay'), ('facts', 'actsfay'), ('sleep', 'eepslay'), ('steps', 'epsstay'),
('wants', 'antsway'), ('forth', 'orthfay'), ('drove', 'ovedray'), ('mouth',

'outhmay'), ('worry', 'orryway'), ('dried', 'ieddray'), ('stood', 'oodstay'),
('brush', 'ushbray'), ('sting', 'ingstay'), ('mirth', 'irthmay'), ('heads',
'eadshay'), ('sight', 'ightsay'), ('whose', 'osewhay'), ('drawn', 'awndray'),
('score', 'orescay'), ('party', 'artypay'), ('tears', 'earstay'), ('heart',
'earthay'), ('words', 'ordsway'), ('hunts', 'untshay'), ('trace', 'acetray'),
('trial', 'ialtray'), ('swoon', 'oonsway'), ('knees', 'eesknay'), ('bring',
'ingbray'), ('solid', 'olidsay'), ('sharp', 'arpshay'), ('basis', 'asisbay'),
('cavil', 'avilcay'), ('rated', 'atedray'), ('grave', 'avegray'), ('serve',
'ervesay'), ('betty', 'ettybay'), ('judge', 'udgejay'), ('boast', 'oastbay'),
('march', 'archmay'), ('built', 'uiltbay'), ('clear', 'earclay'), ('happy',
'appyhay'), ('fruit', 'uitfray'), ('witty', 'ittyway'), ('total', 'otaltay'),
('dared', 'aredday'), ('defer', 'eferday'), ('tried', 'iedtray'), ('vexed',
'exedvay'), ('walls', 'allsway'), ('liked', 'ikedlay'), ('prove', 'ovepray'),
('sneer', 'eersnay'), ('merit', 'eritmay'), ('scott', 'ottscay'), ('madam',
'adammay'), ('truth', 'uthtray'), ('minor', 'inormay'), ('stare', 'arestay'),
('calls', 'allscay'), ('lucky', 'uckylay'), ('hence', 'encehay'), ('press',
'esspray'), ('comes', 'omescay'), ('dined', 'inedday'), ('think', 'inkthay'),
('plate', 'ateplay'), ('grief', 'iefgray'), ('names', 'amesnay'), ('stock',
'ockstay'), ('short', 'ortshay'), ('moved', 'ovedmay'), ('teach', 'eachtay'),
('fears', 'earsfay'), ('tiled', 'iledtay'), ('cried', 'iedcray'), ('mound',
'oundmay'), ('small', 'allsmay'), ('twice', 'icetway'), ('lines', 'ineslay'),
('looks', 'ookslay'), ('rival', 'ivalray'), ('first', 'irstfay'), ('wound',
'oundway'), ('saves', 'avessay'), ('nurse', 'ursenay'), ('treat', 'eattray'),
('could', 'ouldcay'), ('rider', 'iderray'), ('lobby', 'obbylay'), ('table',
'abletay'), ('touch', 'ouchtay'), ('topic', 'opictay'), ('stand', 'andstay'),
('leant', 'eantlay'), ('grass', 'assgray'), ('check', 'eckchay'), ('quite',
'itequay'), ('wives', 'ivesway'), ('walks', 'alksway'), ('labor', 'aborlay'),
('house', 'ousehay'), ('drops', 'opsdray'), ('loose', 'ooselay'), ('tempt',
'empttay'), ('place', 'aceplay'), ('local', 'ocallay'), ('seats', 'eatssay'),
('gates', 'atesgay'), ('dropt', 'optdray'), ('cares', 'arescay'), ('falls',
'allsfay'), ('speak', 'eakspay'), ('shops', 'opsshay'), ('groom', 'oomgray'),
('sugar', 'ugarsay'), ('front', 'ontfray'), ('visit', 'isitvay'), ('raise',
'aiseray'), ('fetch', 'etchfay'), ('silks', 'ilkssay'), ('beset', 'esetbay'),
('rocks', 'ocksray'), ('start', 'artstay'), ('winks', 'inksway'), ('sends',
'endssay'), ('piano', 'ianopay'), ('fully', 'ullyfay'), ('widow', 'idowway'),
('wiser', 'iserway'), ('dirty', 'irtyday'), ('below', 'elowbay'), ('manor',
'anormay'), ('scale', 'alescay'), ('whole', 'olewhay'), ('minds', 'indsmay'),
('glebe', 'ebeglay'), ('fifty', 'iftyfay'), ('years', 'yearsay'), ('story',
'orystay'), ('fresh', 'eshfray'), ('begun', 'egunbay'), ('dance', 'anceday'),
('jokes', 'okesjay'), ('turns', 'urnstay'), ('honor', 'onorhay'), ('spite',
'itespay'), ('green', 'eengray'), ('saint', 'aintsay'), ('hated', 'atedhay'),
('doors', 'oorsday'), ('shown', 'ownshay'), ('power', 'owerpay'), ('today',
'odaytay'), ('floor', 'oorflay'), ('force', 'orcefay'), ('right', 'ightray'),
('trick', 'icktray'), ('dread', 'eaddray'), ('final', 'inalfay'), ('shake',
'akeshay'), ('lamps', 'ampslay'), ('dying', 'yingday'), ('black', 'ackblay'),
('grand', 'andgray'), ('field', 'ieldfay'), ('plead', 'eadplay'), ('loved',
'ovedlay'), ('terms', 'ermstay'), ('spend', 'endspay'), ('mohrs', 'ohrsmay'),
('young', 'youngay'), ('great', 'eatgray'), ('guess', 'uessgay'), ('ninny',

'innynay'), ('hasty', 'astyhay'), ('nerve', 'ervenay'), ('meals', 'ealsmay'), ('three', 'eethray'), ('shade', 'adeshay'), ('store', 'orestay'), ('daily', 'ailyday'), ('taste', 'astetay'), ('gross', 'ossgray'), ('dares', 'aresday'), ('grate', 'ategray'), ('stuff', 'uffstay'), ('sheet', 'eetshay')], (14, 18): [('breakfast-room', 'eakfastbray-oomray'), ('lavender-water', 'avenderlay-aterway'), ('broken-hearted', 'okenbray-eartedhay'), ('spunging-house', 'ungingspay-ousehay')], (9, 12): [('unwelcome', 'unwelcomeway'), ('exhausted', 'exhaustedway'), ('intrusion', 'intrusionway'), ('untouched', 'untouchedway'), ('interview', 'interviewway'), ('effusions', 'effusionsway'), ('impartial', 'impartialway'), ('inventive', 'inventiveway'), ('emergency', 'emergencyway'), ('officious', 'officiousway'), ('everybody', 'everybodyway'), ('observant', 'observantway'), ('emptiness', 'emptinessway'), ('occasions', 'occasionsway'), ('assembled', 'assembledway'), ('admirable', 'admirableway'), ('uselessly', 'uselesslyway'), ('excellent', 'excellentway'), ('associate', 'associateway'), ('affecting', 'affectingway'), ('absurdity', 'absurdityway'), ('allowable', 'allowableway'), ('appointed', 'appointedway'), ('overheard', 'overheardway'), ('improving', 'improvingway'), ('eagerness', 'eagernessway'), ('unluckily', 'unluckilyway'), ('allowance', 'allowanceway'), ('estimable', 'estimableway'), ('obviating', 'obviatingway'), ('entrusted', 'entrustedway'), ('arranging', 'arrangingway'), ('exclaimed', 'exclaimedway'), ('ingenuity', 'ingenuityway'), ('outwardly', 'outwardlyway'), ('expensive', 'expensiveway'), ('entreated', 'entreatedway'), ('unconcern', 'unconcernway'), ('obtaining', 'obtainingway'), ('interests', 'interestsway'), ('assiduous', 'assiduousway'), ('adjusting', 'adjustingway'), ('ungenteel', 'ungenteelway'), ('encourage', 'encourageway'), ('apartment', 'apartmentway'), ('inheritor', 'inheritorway'), ('unceasing', 'unceasingway'), ('insulting', 'insultingway'), ('unchanged', 'unchangedway'), ('animation', 'animationway'), ('enclosure', 'enclosureway'), ('apologize', 'apologizeway'), ('earnestly', 'earnestlyway'), ('exquisite', 'exquisiteway'), ('attending', 'attendingway'), ('invention', 'inventionway'), ('enquiries', 'enquiriesway'), ('appetites', 'appetitesway'), ('additions', 'additionsway'), ('impudence', 'impudenceway'), ('insincere', 'insincereway'), ('abundance', 'abundanceway'), ('exchanged', 'exchangedway'), ('extremely', 'extremelyway'), ('inflicted', 'inflictedway'), ('attracted', 'attractedway'), ('inelegant', 'inelegantway'), ('undoubted', 'undoubtedway'), ('etiquette', 'etiquetteway'), ('unfeigned', 'unfeignedway'), ('agreeable', 'agreeableway'), ('unsubdued', 'unsubduedway'), ('inspiring', 'inspiringway'), ('unguarded', 'unguardedway'), ('advancing', 'advancingway'), ('undressed', 'undressedway'), ('excursive', 'excursiveway'), ('extending', 'extendingway'), ('extolling', 'extollingway'), ('offending', 'offendingway'), ('eloquence', 'eloquenceway'), ('ornaments', 'ornamentsway'), ('intruding', 'intrudingway'), ('incessant', 'incessantway'), ('unwilling', 'unwillingway'), ('annamaria', 'annamariaway'), ('unbounded', 'unboundedway'), ('incurable', 'incurableway'), ('unnoticed', 'unnoticedway'), ('expressly', 'expresslyway'), ('impetuous', 'impetuousway'), ('appearing', 'appearingway'), ('ingenious', 'ingeniousway'), ('amazement', 'amazementway'), ('animating', 'animatingway'), ('interfere', 'interfereway'), ('ascending', 'ascendingway'), ('affection', 'affectionway'), ('undergone', 'undergoneway'), ('unstudied', 'unstudiedway'), ('attempted', 'attemptedway'), ('infirmity', 'infirmityway'), ('apologies', 'apologiesway'), ('elsewhere', 'elsewhereway'),

```
('expecting', 'expectingway'), ('informing', 'informingway'), ('admission',
'admissionway'), ('extremity', 'extremityway')], (6, 9): [('unread',
'unreadway'), ('owning', 'owningway'), ('odious', 'odiousway'), ('either',
'eitherway'), ('insult', 'insultway'), ('awhile', 'awhileway'), ('assist',
'assistway'), ('injure', 'injureway'), ('eluded', 'eludedway'), ('others',
'othersway'), ('untied', 'untiedway'), ('excess', 'excessway'), ('urgent',
'urgentway'), ('appear', 'appearway'), ('aspect', 'aspectway'), ('issued',
'issuedway'), ('evince', 'evinceway'), ('eating', 'eatingway'), ('easter',
'easterway'), ('arisen', 'arisenway'), ('earned', 'earnedway'), ('esteem',
'esteemway'), ('accuse', 'accuseway'), ('owners', 'ownersway'), ('events',
'eventsway'), ('energy', 'energyway'), ('affair', 'affairway'), ('effect',
'effectway'), ('unlike', 'unlikeway'), ('unison', 'unisonway'), ('enough',
'enoughway'), ('invite', 'inviteway'), ('income', 'incomeway'), ('extent',
'extentway'), ('office', 'officeway'), ('ankles', 'anklesway'), ('admire',
'admireway'), ('induce', 'induceway'), ('infant', 'infantway'), ('expand',
'expandway'), ('orders', 'ordersway'), ('occupy', 'occupyway'), ('attend',
'attendway'), ('unable', 'unableway'), ('olives', 'olivesway'), ('awaken',
'awakenway'), ('engage', 'engageway'), ('eldest', 'eldestway'), ('eyeing',
'eyeingway'), ('advice', 'adviceway'), ('always', 'alwaysway'), ('insist',
'insistway'), ('accept', 'acceptway'), ('indies', 'indiesway'), ('excite',
'exciteway'), ('openly', 'openlyway'), ('oblige', 'obligeway'), ('amidst',
'amidstway'), ('across', 'acrossway'), ('afraid', 'afraidway'), ('uproar',
'uproarway')], (8, 10): [('villages', 'illagesvay'), ('doubtful', 'oubtfulday'),
('deceived', 'eceivedday'), ('foreseen', 'oreseenfay'), ('dennison',
'ennisonday'), ('steadily', 'eadilystay'), ('startled', 'artledstay'),
('drawings', 'awingsdray'), ('rejected', 'ejectedray'), ('deserves',
'eservesday'), ('curricle', 'urriclecay'), ('domestic', 'omesticday'),
('believed', 'elievedbay'), ('packages', 'ackagespay'), ('kneeling',
'eelingknay'), ('degraded', 'egradedday'), ('medicine', 'edicinemay'),
('promoted', 'omotedpray'), ('wondered', 'onderedway'), ('reproach',
'eproachray'), ('somerset', 'omersetsay'), ('contents', 'ontentscay'),
('children', 'ildrenchay'), ('politics', 'oliticspay'), ('hastened',
'astenedhay'), ('relative', 'elativeray'), ('henshawe', 'enshawehay'),
('material', 'aterialmay'), ('welcomed', 'elcomedway'), ('weddings',
'eddingsway'), ('diverted', 'ivertedday'), ('patience', 'atiencepay'),
('delivery', 'eliveryday'), ('hazarded', 'azardedhay'), ('happiest',
'appiesthay'), ('trifling', 'iflingtray'), ('casement', 'asementcay'),
('reformed', 'eformedray'), ('demanded', 'emandedday'), ('ceremony',
'eremonycay'), ('vigorous', 'igorousvay'), ('mornings', 'orningsmay'),
('hilarity', 'ilarityhay'), ('daughter', 'aughterday'), ('vexation',
'exationvay'), ('shocking', 'ockingshay'), ('previous', 'eviouspray'),
('pleasant', 'easantplay'), ('candidly', 'andidlycay'), ('cheerful',
'eerfulchay'), ('festival', 'estivalfay'), ('changing', 'angingchay'),
('conveyed', 'onveyedcay'), ('lombardy', 'ombardylay'), ('militate',
'ilitatemay'), ('humanity', 'umanityhay'), ('giggling', 'igglinggay'),
('standard', 'andardstay'), ('returned', 'eturnedray'), ('proclaim',
'oclaimpray'), ('ridicule', 'idiculeray'), ('pointers', 'ointerspay'),
('grounded', 'oundedgray'), ('contrast', 'ontrastcay'), ('feelings',
```

'eelingsfay'), ('mistakes', 'istakesmay'), ('prepared', 'eparedpray'),
('stranger', 'angerstray'), ('habitual', 'abitualhay'), ('firmness',
'irmnessfay'), ('rational', 'ationalray'), ('regulate', 'egulateray'),
('claiming', 'aimingclay'), ('knocking', 'ockingknay'), ('servants',
'ervantssay'), ('goodness', 'oodnessgay'), ('mutually', 'utuallymay'),
('legacies', 'egacieslay'), ('sisterly', 'isterlysay'), ('punctual',
'unctualpay'), ('deciding', 'ecidingday'), ('suitable', 'uitablesay'),
('guarding', 'uardinggay'), ('converse', 'onversecay'), ('slighter',
'ighterslay'), ('dissuade', 'issuadeday'), ('despatch', 'espatchday'),
('blackest', 'ackestblay'), ('recalled', 'ecalledray'), ('loitered',
'oiteredlay'), ('dreading', 'eadingdray'), ('relieved', 'elievedray'),
('westward', 'estwardway'), ('thistles', 'istlesthay'), ('defended',
'efendedday'), ('contempt', 'ontemptcay'), ('sensibly', 'ensiblysay'),
('romantic', 'omanticray'), ('comforts', 'omfortscay'), ('jealousy',
'ealousyjay'), ('repeated', 'epeatedray'), ('banditti', 'andittibay'),
('greatest', 'eatestgray'), ('mistrust', 'istrustmay'), ('forwards',
'orwardsfay'), ('rewarded', 'ewardedray'), ('watching', 'atchingway'),
('negative', 'egativenay'), ('sympathy', 'ympathysay'), ('helpless',
'elplesshay'), ('quickest', 'ickestquay'), ('portrait', 'ortraitpay'),
('consoled', 'onsoledcay'), ('fourteen', 'ourteenfay'), ('retarded',
'etardedray'), ('comments', 'ommentscay'), ('jennings', 'enningsjay'),
('securing', 'ecuringsay'), ('withdrew', 'ithdrewway'), ('youthful',
'youthfulay'), ('bachelor', 'achelorbay'), ('convince', 'onvincecay'),
('measured', 'easuredmay'), ('goodwill', 'oodwillgay'), ('branched',
'anchedbray'), ('purchase', 'urchasepay'), ('covenant', 'ovenantcay'),
('reseated', 'eseatedray'), ('concerns', 'oncernscay'), ('hardship',
'ardshiphay'), ('lessened', 'essenedlay'), ('delaford', 'elafordday'),
('governed', 'overnedgay'), ('listened', 'istenedlay'), ('constant',
'onstantcay'), ('fatigued', 'atiguedfay'), ('persuade', 'ersuadepay'),
('critique', 'itiquecray'), ('remedies', 'emediesray'), ('stepping',
'eppingstay'), ('kindness', 'indnesskay'), ('sociable', 'ociablesay'),
('meetings', 'eetingsmay'), ('deserved', 'eservedday'), ('thoughts',
'oughtsthay'), ('becoming', 'ecomingbay'), ('heartily', 'eartilyhay'),
('promised', 'omisedpray'), ('shutting', 'uttingshay'), ('reproved',
'eprovedray'), ('ladyship', 'adyshiplay'), ('pursuing', 'ursuingpay'),
('residing', 'esidingray'), ('sentence', 'entencesay'), ('revealed',
'evealedray'), ('decrease', 'ecreaseday'), ('speedily', 'eedilyspay'),
('commands', 'ommandscay'), ('mistaken', 'istakenmay'), ('settling',
'ettlingsay'), ('polished', 'olishedpay'), ('stanhill', 'anhillstay'),
('retailed', 'etailedray'), ('pointing', 'ointingpay'), ('strongly',
'onglystray'), ('subjects', 'ubjectssay'), ('calmness', 'almnesscay'),
('doctrine', 'octrineday'), ('handsome', 'andsomehay'), ('profited',
'ofitedpray'), ('planning', 'anningplay'), ('denoting', 'enotingday'),
('commonly', 'ommonlycay'), ('bestowed', 'estowedbay'), ('marriage',
'arriagemay'), ('presents', 'esentspray'), ('complete', 'ompletecay'),
('striking', 'ikingstray'), ('needless', 'eedlessnay'), ('plighted',
'ightedplay'), ('tranquil', 'anquiltray'), ('spacious', 'aciousspay'),
('nonsense', 'onsensenay'), ('likeness', 'ikenesslay'), ('valuable',

```
'aluablevay'), ('laudable', 'audablelay'), ('delirium', 'eliriumday'),
('believes', 'elievesbay'), ('premises', 'emisespray'), ('presumed',
'esumedpray'), ('coldness', 'oldnesscay'), ('metaphor', 'etaphormay'),
('thankful', 'ankfulthay'), ('suffered', 'ufferedsay'), ('preserve',
'eservepray'), ('resorted', 'esortedray'), ('felicity', 'elicityfay'),
('betrayed', 'etrayedbay'), ('throwing', 'owingthray'), ('powerful',
'owerfulpay'), ('humoured', 'umouredhay'), ('practise', 'actisepray'),
('coloured', 'olouredcay'), ('regiment', 'egimentray'), ('stubborn',
'ubbornstay'), ('huswifes', 'uswifeshay'), ('painting', 'aintingpay'),
('friendly', 'iendlyfray'), ('happened', 'appenedhay'), ('moderate',
'oderatemay'), ('solitary', 'olitarysay'), ('wildness', 'ildnessway'),
('marianne', 'ariannemay'), ('delicacy', 'elicacyday'), ('progress',
'ogresspray'), ('coachman', 'oachmancay'), ('grieving', 'ievinggray'),
('troubles', 'oublestray'), ('coolness', 'oolnesscay'), ('possibly',
'ossiblypay'), ('relating', 'elatingray'), ('patterns', 'atternspay'),
('consists', 'onsistscay'), ('creating', 'eatingcray'), ('confused',
'onfusedcay'), ('critical', 'iticalcray'), ('laboured', 'abouredlay'),
('minutely', 'inutelymay'), ('tendency', 'endencytay'), ('distrust',
'istrustday'), ('distress', 'istressday'), ('conclude', 'oncludecay'),
('sixpence', 'ixpencesay'), ('somewhat', 'omewhatsay'), ('coincide',
'oincidecay'), ('harmless', 'armlesshay'), ('juvenile', 'uvenilejay'),
('pleasing', 'easingplay'), ('precious', 'eciouspray'), ('restrain',
'estrainray'), ('produced', 'oducedpray'), ('renewing', 'enewingray'),
('pinching', 'inchingpay'), ('severely', 'everelysay'), ('blossoms',
'ossomsblay'), ('heighten', 'eightenhay'), ('wandered', 'anderedway'),
('properly', 'operlypray'), ('veracity', 'eracityvay'), ('released',
'eleasedray'), ('transact', 'ansacttray'), ('helpmate', 'elpmatehay'),
('business', 'usinessbay'), ('concerto', 'oncertocay'), ('reserved',
'eservedray'), ('reliance', 'elianceray'), ('headache', 'eadachehay'),
('families', 'amiliesfay'), ('centered', 'enteredcay'), ('mulberry',
'ulberrymay'), ('sensible', 'ensiblesay'), ('steepest', 'eepeststay'),
('compared', 'omparedcay'), ('speaking', 'eakingspay'), ('censured',
'ensuredcay'), ('screened', 'eenedscray'), ('pictures', 'icturespay'),
('reaching', 'eachingray'), ('wonderer', 'ondererway'), ('whenever',
'eneverwhay'), ('detested', 'etestedday'), ('dispatch', 'ispatchday'),
('shutters', 'uttersshay'), ('disgrace', 'isgraceday'), ('dejected',
'ejectedday'), ('conquest', 'onquestcay'), ('foretell', 'oretellfay'),
('weakened', 'eakenedway'), ('brooding', 'oodingbray'), ('shedding',
'eddingshay'), ('poignant', 'oignantpay'), ('trembled', 'embledtray'),
('resolute', 'esoluteray'), ('reverted', 'evertedray'), ('silenced',
'ilencedsay'), ('jointure', 'ointurejay')], (10, 13): [('illaudable',
'illaudableway'), ('incivility', 'incivilityway'), ('oppression',
'oppressionway'), ('acquainted', 'acquaintedway'), ('economical',
'economicalway'), ('ebullition', 'ebullitionway'), ('affronting',
'affrontingway'), ('unforeseen', 'unforeseenway'), ('auspicious',
'auspiciousway'), ('indecorous', 'indecorousway'), ('attractive',
'attractiveway'), ('exercising', 'exercisingway'), ('unsatiable',
'unsatiableway'), ('individual', 'individualway'), ('enthusiasm',
```

'enthusiasmway'), ('announcing', 'announcingway'), ('indistinct',
'indistinctway'), ('affliction', 'afflictionway'), ('imaginable',
'imaginableway'), ('obstructed', 'obstructedway'), ('improperly',
'improperlyway'), ('illiterate', 'illiterateway'), ('interested',
'interestedway'), ('infamously', 'infamouslyway'), ('amusements',
'amusementsway'), ('endeavours', 'endeavoursway'), ('inhabiting',
'inhabitingway'), ('indisposed', 'indisposedway'), ('acquiesced',
'acquiescedway'), ('anticipate', 'anticipateway'), ('accurately',
'accuratelyway'), ('influenced', 'influencedway'), ('assemblies',
'assembliesway'), ('approached', 'approachedway'), ('accusation',
'accusationway'), ('apparently', 'apparentlyway'), ('intimately',
'intimatelyway'), ('uneasiness', 'uneasinessway'), ('alteration',
'alterationway'), ('artificial', 'artificialway'), ('admiration',
'admirationway'), ('accidental', 'accidentalway'), ('unexpected',
'unexpectedway'), ('insensible', 'insensibleway'), ('expressing',
'expressingway'), ('accelerate', 'accelerateway'), ('everything',
'everythingway'), ('increasing', 'increasingway'), ('undervalue',
'undervalueway'), ('invaluable', 'invaluableway'), ('affections',
'affectionsway'), ('infinitely', 'infinitelyway'), ('unsuitable',
'unsuitableway'), ('authorised', 'authorisedway'), ('impudently',
'impudentlyway'), ('unpardoned', 'unpardonedway'), ('inevitable',
'inevitableway'), ('ungenerous', 'ungenerousway'), ('impression',
'impressionway'), ('indelicacy', 'indelicacyway'), ('inducement',
'inducementway'), ('unaffected', 'unaffectedway'), ('instigated',
'instigatedway'), ('inquietude', 'inquietudeway'), ('attempting',
'attemptingway'), ('experience', 'experienceway'), ('impatience',
'impatienceway'), ('indulgence', 'indulgenceway'), ('affability',
'affabilityway'), ('unavailing', 'unavailingway'), ('incommoded',
'incommodedway'), ('altogether', 'altogetherway')], (4, 6): [('done', 'oneday'),
('wife', 'ifeway'), ('plea', 'eaplay'), ('mall', 'allmay'), ('sang', 'angsay'),
('clue', 'ueclay'), ('bell', 'ellbay'), ('bind', 'indbay'), ('join', 'oinjay'),
('went', 'entway'), ('vile', 'ilevay'), ('hens', 'enshay'), ('tore', 'oretay'),
('rage', 'ageray'), ('flow', 'owflay'), ('gate', 'ategay'), ('palm', 'almpay'),
('give', 'ivegay'), ('lips', 'ipslay'), ('sink', 'inksay'), ('rent', 'entray'),
('gave', 'avegay'), ('joke', 'okejay'), ('miss', 'issmay'), ('mere', 'eremay'),
('made', 'ademay'), ('dogs', 'ogsday'), ('bent', 'entbay'), ('duly', 'ulyday'),
('when', 'enwhay'), ('rate', 'ateray'), ('sure', 'uresay'), ('land', 'andlay'),
('pink', 'inkpay'), ('hint', 'inthay'), ('cast', 'astcay'), ('park', 'arkpay'),
('bold', 'oldbay'), ('door', 'oorday'), ('bulk', 'ulkbay'), ('doat', 'oatday'),
('coat', 'oatcay'), ('date', 'ateday'), ('foot', 'ootfay'), ('hand', 'andhay'),
('damp', 'ampday'), ('pall', 'allpay'), ('goes', 'oesgay'), ('rung', 'ungray'),
('none', 'onenay'), ('game', 'amegay'), ('nods', 'odsnay'), ('read', 'eadray'),
('gold', 'oldgay'), ('sing', 'ingsay'), ('puts', 'utspay'), ('cats', 'atscay'),
('many', 'anymay'), ('dare', 'areday'), ('noon', 'oonnay'), ('said', 'aidsay'),
('mark', 'arkmay'), ('case', 'asecay'), ('hate', 'atehay'), ('word', 'ordway'),
('beat', 'eatbay'), ('lose', 'oselay'), ('fish', 'ishfay'), ('laid', 'aidlay'),
('fine', 'inefay'), ('snug', 'ugsnay'), ('moss', 'ossmay'), ('heir', 'eirhay'),
('sigh', 'ighsay'), ('mend', 'endmay'), ('sake', 'akesay'), ('pass', 'asspay'),

```
('debt', 'ebtday'), ('lane', 'anelay'), ('days', 'aysday'), ('deep', 'eepday'),
('song', 'ongsay'), ('spot', 'otspay'), ('took', 'ooktay'), ('both', 'othbay'),
('year', 'yearay'), ('mama', 'amamay'), ('hill', 'illhay'), ('beds', 'edsbay'),
('less', 'esslay'), ('gown', 'owngay'), ('kill', 'illkay'), ('free', 'eefray'),
('hall', 'allhay'), ('room', 'oomray'), ('gray', 'aygray'), ('beef', 'eefbay'),
('kept', 'eptkay'), ('what', 'atwhay'), ('blue', 'ueblay'), ('fell', 'ellfay'),
('draw', 'awdray'), ('play', 'ayplay'), ('baby', 'abybay'), ('rude', 'uderay'),
('trim', 'imtray'), ('chat', 'atchay'), ('bely', 'elybay'), ('fled', 'edflay'),
('past', 'astpay'), ('wait', 'aitway'), ('meal', 'ealmay'), ('wont', 'ontway'),
('week', 'eekway'), ('best', 'estbay'), ('line', 'inelay'), ('lawn', 'awnlay'),
('tree', 'eetray'), ('tend', 'endtay'), ('find', 'indfay'), ('five', 'ivefay'),
('gone', 'onegay'), ('need', 'eednay'), ('drew', 'ewdray'), ('more', 'oremay'),
('cool', 'oolcay'), ('suit', 'uitsay'), ('from', 'omfray'), ('cure', 'urecay'),
('will', 'illway'), ('slow', 'owslay'), ('they', 'eythay'), ('name', 'amenay'),
('shot', 'otshay'), ('come', 'omecay'), ('tied', 'iedtay'), ('came', 'amecay'),
('boys', 'oysbay'), ('wish', 'ishway'), ('hour', 'ourhay'), ('kind', 'indkay'),
('mile', 'ilemay'), ('quit', 'itquay'), ('john', 'ohnjay'), ('hazy', 'azyhay'),
('fall', 'allfay'), ('sell', 'ellsay'), ('tale', 'aletay'), ('grew', 'ewgray'),
('seek', 'eeksay'), ('muff', 'uffmay'), ('soft', 'oftsay'), ('same', 'amesay'),
('body', 'odybay'), ('head', 'eadhay'), ('like', 'ikelay'), ('here', 'erehay'),
('help', 'elphay'), ('sail', 'ailsay'), ('pope', 'opepay'), ('deny', 'enyday'),
('bond', 'ondbay'), ('bath', 'athbay'), ('blow', 'owblay'), ('hold', 'oldhay'),
('safe', 'afesay'), ('path', 'athpay'), ('feed', 'eedfay'), ('thus', 'usthay'),
('four', 'ourfay'), ('lest', 'estlay'), ('thin', 'inthay')], (11, 14):
[('approaching', 'approachingway'), ('unequivocal', 'unequivocalway'),
('exceedingly', 'exceedinglyway'), ('application', 'applicationway'),
('observation', 'observationway'), ('unhappiness', 'unhappinessway'),
('interrupted', 'interruptedway'), ('enlightened', 'enlightenedway'),
('appearances', 'appearancesway'), ('unconnected', 'unconnectedway'),
('unconcerned', 'unconcernedway'), ('indignation', 'indignationway'),
('accommodate', 'accommodateway'), ('inattention', 'inattentionway'),
('undoubtedly', 'undoubtedlyway'), ('intercourse', 'intercourseway'),
('imperfectly', 'imperfectlyway'), ('imprudently', 'imprudentlyway'),
('affectation', 'affectationway'), ('instigation', 'instigationway'),
('inclination', 'inclinationway'), ('undervalued', 'undervaluedway'),
('affirmative', 'affirmativeway'), ('inconstancy', 'inconstancyway'),
('universally', 'universallyway'), ('engagements', 'engagementsway'),
('exclamation', 'exclamationway'), ('elucidation', 'elucidationway'),
('accustomary', 'accustomaryway'), ('approbation', 'approbationway'),
('unnecessary', 'unnecessaryway'), ('associating', 'associatingway'),
('attributing', 'attributingway'), ('unobtrusive', 'unobtrusiveway'),
('impropriety', 'improprietyway'), ('examination', 'examinationway'),
('alterations', 'alterationsway'), ('unexpressed', 'unexpressedway'),
('abridgement', 'abridgementway'), ('exhilarated', 'exhilaratedway'),
('unsuspected', 'unsuspectedway'), ('independent', 'independentway'),
('alternately', 'alternatelyway'), ('enforcement', 'enforcementway'),
('undertaking', 'undertakingway'), ('outstripped', 'outstrippedway'),
('appointment', 'appointmentway'), ('explanation', 'explanationway'),
```

```
('experienced', 'experiencedway'), ('unknowingly', 'unknowinglyway'),
('undesirable', 'undesirableway'), ('ascertained', 'ascertainedway'),
('enumeration', 'enumerationway'), ('alleviation', 'alleviationway'),
('instruction', 'instructionway'), ('undeserving', 'undeservingway'),
('earnestness', 'earnestnessway'), ('insincerity', 'insincerityway'),
('accompanied', 'accompaniedway'), ('employments', 'employmentsway'),
('inferiority', 'inferiorityway'), ('inexcusable', 'inexcusableway'),
('introducing', 'introducingway'), ('extravagant', 'extravagantway'),
('impatiently', 'impatientlyway'), ('attentively', 'attentivelyway'),
('eligibility', 'eligibilityway'), ('uncertainty', 'uncertaintyway')], (7, 10):
[('anxiety', 'anxietyway'), ('attacks', 'attacksway'), ('ailment',
'ailmentway'), ('awkward', 'awkwardway'), ('effects', 'effectsway'), ('induced',
'inducedway'), ('uncivil', 'uncivilway'), ('against', 'againstway'), ('apology',
'apologyway'), ('endless', 'endlessway'), ('alluded', 'alludedway'), ('ability',
'abilityway'), ('address', 'addressway'), ('engaged', 'engagedway'), ('eagerly',
'eagerlyway'), ('expects', 'expectsway'), ('elliott', 'elliottway'), ('oneself',
'oneselfway'), ('anybody', 'anybodyway'), ('augment', 'augmentway'), ('utility',
'utilityway'), ('unquiet', 'unquietway'), ('infancy', 'infancyway'), ('altered',
'alteredway'), ('unequal', 'unequalway'), ('adapted', 'adaptedway'), ('useless',
'uselessway'), ('opposed', 'opposedway'), ('artless', 'artlessway'), ('imagery',
'imageryway'), ('amiably', 'amiablyway'), ('uncouth', 'uncouthway'), ('affixed',
'affixedway'), ('impulse', 'impulseway'), ('opinion', 'opinionway'), ('imitate',
'imitateway'), ('admires', 'admiresway'), ('affront', 'affrontway'), ('amusing',
'amusingway'), ('accents', 'accentsway'), ('implied', 'impliedway'), ('ecstasy',
'ecstasyway'), ('illness', 'illnessway'), ('entreat', 'entreatway'), ('another',
'anotherway'), ('opening', 'openingway'), ('england', 'englandway'), ('ensured',
'ensuredway'), ('imputed', 'imputedway'), ('enabled', 'enabledway'), ('offence',
'offenceway'), ('insured', 'insuredway'), ('urgency', 'urgencyway'), ('amazing',
'amazingway'), ('uniform', 'uniformway'), ('avarice', 'avariceway'), ('ellison',
'ellisonway'), ('endowed', 'endowedway'), ('ashamed', 'ashamedway'), ('eloping',
'elopingway'), ('inhabit', 'inhabitway'), ('advised', 'advisedway'), ('attract',
'attractway'), ('already', 'alreadyway'), ('enquire', 'enquireway'), ('affects',
'affectsway'), ('invalid', 'invalidway')], (9, 11): [('genuinely',
'enuinelygay'), ('columella', 'olumellacay'), ('vouchsafe', 'ouchsafevay'),
('forwarded', 'orwardedfay'), ('scattered', 'atteredscay'), ('christmas',
'istmaschray'), ('crimsoned', 'imsonedcray'), ('contrives', 'ontrivescay'),
('miserable', 'iserablemay'), ('preceding', 'ecedingpray'), ('healthful',
'ealthfulhay'), ('confiding', 'onfidingcay'), ('perfectly', 'erfectlypay'),
('prospects', 'ospectspray'), ('withdrawn', 'ithdrawnway'), ('lingering',
'ingeringlay'), ('contained', 'ontainedcay'), ('depressed', 'epressedday'),
('disgusted', 'isgustedday'), ('promoting', 'omotingpray'), ('supported',
'upportedsay'), ('gradually', 'aduallygray'), ('disturbed', 'isturbedday'),
('perpetual', 'erpetualpay'), ('pronounce', 'onouncepray'), ('sincerely',
'incerelysay'), ('seventeen', 'eventeensay'), ('presented', 'esentedpray'),
('medicines', 'edicinesmay'), ('syllables', 'yllablessay'), ('behaviour',
'ehaviourbay'), ('niggardly', 'iggardlynay'), ('commonest', 'ommonestcay'),
('necessary', 'ecessarynay'), ('prettiest', 'ettiestpray'), ('condemned',
'ondemnedcay'), ('conquests', 'onquestscay'), ('fashioned', 'ashionedfay'),
```

('hysterics', 'ystericshay'), ('smallness', 'allnesssmay'), ('professed',
'ofessedpray'), ('conceited', 'onceitedcay'), ('compliant', 'ompliantcay'),
('frankness', 'anknessfray'), ('merriment', 'errimentmay'), ('resembled',
'esembledray'), ('leisurely', 'eisurelylay'), ('dispersed', 'ispersedday'),
('departure', 'epartureday'), ('pecuniary', 'ecuniarypay'), ('frivolous',
'ivolousfray'), ('situation', 'ituationsay'), ('discourse', 'iscourseday'),
('resettled', 'esettledray'), ('beholding', 'eholdingbay'), ('gentleman',
'entlemangay'), ('chagrined', 'agrinedchay'), ('struggled', 'uggledstray'),
('preparing', 'eparingpray'), ('respected', 'espectedray'), ('newspaper',
'ewspapernay'), ('justified', 'ustifiedjay'), ('sacrifice', 'acrificesay'),
('stammered', 'ammeredstay'), ('smartness', 'artnesssmay'), ('continued',
'ontinuedcay'), ('contrived', 'ontrivedcay'), ('reconcile', 'econcileray'),
('modestest', 'odestestmay'), ('cordially', 'ordiallycay'), ('sentiment',
'entimentsay'), ('shameless', 'amelessshay'), ('reflected', 'eflectedray'),
('feelingly', 'eelinglyfay'), ('confident', 'onfidentcay'), ('proposals',
'oposalspray'), ('warehouse', 'arehouseway'), ('deviation', 'eviationday'),
('statement', 'atementstay'), ('submitted', 'ubmittedsay'), ('believing',
'elievingbay'), ('gentility', 'entilitygay'), ('resources', 'esourcesray'),
('composure', 'omposurecay'), ('deserving', 'eservingday'), ('vehemence',
'ehemencevay'), ('disgraced', 'isgracedday'), ('perplexed', 'erplexedpay'),
('prejudice', 'ejudicepray'), ('quickened', 'ickenedquay'), ('blundered',
'underedblay'), ('strongest', 'ongeststray'), ('prevailed', 'evailedpray'),
('languages', 'anguageslay'), ('reanimate', 'eanimateray'), ('stationer',
'ationerstay'), ('reminding', 'emindingray'), ('blameless', 'amelessblay'),
('principle', 'inciplepray'), ('hardening', 'ardeninghay'), ('suggested',
'uggestedsay'), ('desperate', 'esperateday'), ('landscape', 'andscapelay'),
('paragraph', 'aragraphpay'), ('provision', 'ovisionpray'), ('relations',
'elationsray'), ('fruitless', 'uitlessfray'), ('christian', 'istianchray'),
('stipulate', 'ipulatestay'), ('constancy', 'onstancycay'), ('gentlemen',
'entlemengay'), ('venturing', 'enturingvay'), ('northward', 'orthwardnay'),
('neighbour', 'eighbournay'), ('weakening', 'eakeningway'), ('persisted',
'ersistedpay'), ('malignant', 'alignantmay'), ('wandering', 'anderingway'),
('witnesses', 'itnessesway'), ('dangerous', 'angerousday'), ('safeguard',
'afeguardsay'), ('treasured', 'easuredtray'), ('cleveland', 'evelandclay'),
('patronage', 'atronagepay'), ('despising', 'espisingday'), ('foolishly',
'oolishlyfay'), ('seriously', 'eriouslysay'), ('restraint', 'estraintray'),
('september', 'eptembersay'), ('distrusts', 'istrustsday'), ('temperate',
'emperatetay'), ('finishing', 'inishingfay'), ('persuaded', 'ersuadedpay'),
('stupified', 'upifiedstay'), ('declaring', 'eclaringday'), ('dejection',
'ejectionday'), ('soothings', 'oothingssay'), ('conveying', 'onveyingcay'),
('possessed', 'ossessedpay'), ('described', 'escribedday'), ('spreading',
'eadingspray'), ('foresight', 'oresightfay'), ('political', 'oliticalpay'),
('concluded', 'oncludedcay'), ('doubtless', 'oubtlessday'), ('confusion',
'onfusioncay'), ('disliking', 'islikingday'), ('carriages', 'arriagescay'),
('favourite', 'avouritefay'), ('brilliant', 'illiantbray'), ('hastening',
'asteninghay'), ('precisely', 'eciselypray'), ('wondering', 'onderingway'),
('shoulders', 'oulderssshay'), ('honorably', 'onorablyhay'), ('privilege',
'ivilegepray'), ('deference', 'eferenceday'), ('vulgarity', 'ulgarityvay'),

```
('reprimand', 'eprimandray'), ('surveying', 'urveyingsay'), ('handsomer',
'andsomerhay'), ('shivering', 'iveringshay'), ('tolerable', 'olerabletay'),
('requiring', 'equiringray'), ('precision', 'ecisionpray'), ('courtland',
'ourtlandcay'), ('sportsman', 'ortsmanspay'), ('happiness', 'appinesshay'),
('pointedly', 'ointedlypay'), ('benefited', 'enefitedbay'), ('substance',
'ubstancesay'), ('beginning', 'eginningbay'), ('gratitude', 'atitudegray'),
('negligent', 'egligentnay'), ('belonging', 'elongingbay'), ('mortified',
'ortifiedmay'), ('neglected', 'eglectednay'), ('shillings', 'illingsshay'),
('warranted', 'arrantedway'), ('performed', 'erformedpay'), ('necessity',
'ecessitynay'), ('slightest', 'ightestslay'), ('guardians', 'uardiansgay'),
('preferred', 'eferredpray'), ('forgotten', 'orgottenfay'), ('refreshed',
'efreshedray'), ('barouches', 'arouchesbay'), ('grievance', 'ievancegray'),
('seduction', 'eductionsay'), ('comprised', 'omprisedcay'), ('listening',
'isteninglay'), ('detection', 'etectionday'), ('narration', 'arrationnay'),
('deceiving', 'eceivingday'), ('provoking', 'ovokingpray'), ('cessation',
'essationcay'), ('questions', 'estionsquay'), ('continual', 'ontinualcay'),
('faintness', 'aintnessfay'), ('diffident', 'iffidentday'), ('referring',
'eferringray'), ('blessings', 'essingsblay'), ('therefore', 'ereforethay'),
('frugality', 'ugalityfray'), ('reprobate', 'eprobateray')], (11, 13):
[('tranquilize', 'anquilizetray'), ('fashionable', 'ashionablefay'),
('respectable', 'espectableray'), ('publication', 'ublicationpay'),
('superintend', 'uperintendsay'), ('recollected', 'ecollectedray'),
('cheerfuller', 'eerfullerchay'), ('concurrence', 'oncurrencecay'),
('restorative', 'estorativeray'), ('reproaching', 'eproachingray'),
('declaration', 'eclarationday'), ('discharging', 'ischargingday'),
('proclaiming', 'oclaimingpray'), ('discovering', 'iscoveringday'),
('possibility', 'ossibilitypay'), ('contributed', 'ontributedcay'),
('captivating', 'aptivatingcay'), ('misinformed', 'isinformedmay'),
('carefulness', 'arefulnesscay'), ('philosophic', 'ilosophicphay'),
('calculation', 'alculationcay'), ('consumption', 'onsumptioncay'),
('confinement', 'onfinementcay'), ('certainties', 'ertaintiescay'),
('destination', 'estinationday'), ('separations', 'eparationssay'),
('voluntarily', 'oluntarilyvay'), ('connoisseur', 'onnoisseurcay'),
('refinements', 'efinementsray'), ('subservient', 'ubservientsay'),
('marlborough', 'arlboroughmay'), ('sagaciously', 'agaciouslysay'),
('fascinating', 'ascinatingfay'), ('comfortable', 'omfortablecay'),
('reflections', 'eflectionsray'), ('sensibility', 'ensibilitysay'),
('forbearance', 'orbearancefay'), ('mischievous', 'ischievousmay'),
('disgraceful', 'isgracefulday'), ('temporizing', 'emporizingtay'),
('sacrificing', 'acrificingsay'), ('spontaneous', 'ontaneousspay'),
('flourishing', 'ourishingflay'), ('lamentation', 'amentationlay'),
('meditations', 'editationsmay'), ('quarrelling', 'arrellingquay'),
('discussions', 'iscussionsday'), ('supposition', 'uppositionsay'),
('comfortably', 'omfortablycay'), ('disappeared', 'isappearedday'),
('preparation', 'eparationpray'), ('consistency', 'onsistencycay'),
('compliments', 'omplimentscay'), ('promptitude', 'omptitudepray'),
('determining', 'eterminingday'), ('countenance', 'ountenancecay'),
('maintenance', 'aintenancemay'), ('contraction', 'ontractioncay'),
```

```
('dissipation', 'issipationday'), ('comparative', 'omparativecay'),
('desperation', 'esperationday'), ('displeasure', 'ispleasureday'),
('despatching', 'espatchingday'), ('playfellows', 'ayfellowsplay'),
('plantations', 'antationsplay'), ('vindication', 'indicationvay'),
('significant', 'ignificantsay'), ('discontents', 'iscontentsday'),
('housekeeper', 'ousekeeperhay'), ('playfulness', 'ayfulnessplay'),
('threatening', 'eateningthray'), ('meritorious', 'eritoriousmay'),
('tediousness', 'ediousnesstay'), ('convenience', 'onveniencecay'),
('possessions', 'ossessionspay'), ('consolation', 'onsolationcay'),
('composition', 'ompositioncay'), ('proposition', 'opositionpray'),
('correctness', 'orrectnesscay'), ('consequence', 'onsequencecay'),
('barbarously', 'arbarouslybay'), ('booksellers', 'ooksellersbay'),
('conjectures', 'onjecturescay'), ('violoncello', 'ioloncellovay'),
('dissembling', 'issemblingday'), ('rapturously', 'apturouslyray'),
('speculative', 'eculativespay'), ('remembering', 'ememberingray'),
('probability', 'obabilitypray'), ('punctuality', 'unctualitypay'),
('forgiveness', 'orgivenessfay'), ('disapproves', 'isapprovesday'),
('troublesome', 'oublesometray')], (3, 5): [('say', 'aysay'), ('for', 'orfay'),
('get', 'etgay'), ('pin', 'inpay'), ('not', 'otnay'), ('fit', 'itfay'), ('rap',
'apray'), ('lip', 'iplay'), ('pry', 'ypray'), ('rob', 'obray'), ('fir',
'irfay'), ('can', 'ancay'), ('two', 'otway'), ('him', 'imhay'), ('son',
'onsay'), ('let', 'etlay'), ('try', 'ytray'), ('hot', 'othay'), ('sum',
'umsay'), ('got', 'otgay'), ('law', 'awlay'), ('how', 'owhay'), ('far',
'arfay'), ('the', 'ethay'), ('sky', 'yskay'), ('sly', 'yslay'), ('cut',
'utcay'), ('hop', 'ophay'), ('had', 'adhay'), ('was', 'asway'), ('bow',
'owbay'), ('hue', 'uehay'), ('now', 'ownay'), ('vex', 'exvay'), ('yet',
'yetay'), ('sit', 'itsay'), ('due', 'ueday'), ('cap', 'apcay'), ('sat',
'atsay'), ('hit', 'ithay'), ('see', 'eesay'), ('pay', 'aypay'), ('nor',
'ornay'), ('fly', 'yflay'), ('dog', 'ogday'), ('too', 'ootay'), ('six',
'ixsay'), ('lap', 'aplay'), ('new', 'ewnay'), ('led', 'edlay'), ('sin',
'insay'), ('ten', 'entay'), ('his', 'ishay'), ('day', 'ayday')], (14, 17):
[('insurmountable', 'insurmountableway'), ('inconveniences',
'inconveniencesway'), ('aggrandizement', 'aggrandizementway'),
('irreconcilable', 'irreconcilableway'), ('unacknowledged',
'unacknowledgedway'), ('unpleasantness', 'unpleasantnessway')], (13, 16):
[('understanding', 'understandingway'), ('intentionally', 'intentionallyway'),
('embarrassment', 'embarrassmentway'), ('unembarrassed', 'unembarrassedway'),
('administering', 'administeringway'), ('instantaneous', 'instantaneousway'),
('intrinsically', 'intrinsicallyway'), ('indispensable', 'indispensableway'),
('impoverishing', 'impoverishingway'), ('opportunities', 'opportunitiesway'),
('impertinently', 'impertinentlyway'), ('independently', 'independentlyway'),
('encroachments', 'encroachmentsway'), ('uninteresting', 'uninterestingway'),
('investigation', 'investigationway'), ('unaccountable', 'unaccountableway'),
('irrepressible', 'irrepressibleway'), ('impossibility', 'impossibilityway')],
(10, 14): [('thirty-six', 'irtythay-ixsay')], (8, 11): [('increase',
'increaseway'), ('agitated', 'agitatedway'), ('unkindly', 'unkindlyway'),
('occurred', 'occurredway'), ('arranged', 'arrangedway'), ('appetite',
'appetiteway'), ('inclined', 'inclinedway'), ('indolent', 'indolentway'),
```

('inquired', 'inquiredway'), ('evenings', 'eveningsway'), ('obliging',
'obligingway'), ('equalled', 'equalledway'), ('accepted', 'acceptedway'),
('extended', 'extendedway'), ('entreaty', 'entreatyway'), ('exigence',
'exigenceway'), ('election', 'electionway'), ('examined', 'examinedway'),
('inviting', 'invitingway'), ('embitter', 'embitterway'), ('idleness',
'idlenessway'), ('admirers', 'admirersway'), ('editions', 'editionsway'),
('extorted', 'extortedway'), ('outweigh', 'outweighway'), ('attended',
'attendedway'), ('addition', 'additionway'), ('entitled', 'entitledway'),
('awakened', 'awakenedway'), ('attacked', 'attackedway'), ('arriving',
'arrivingway'), ('overlook', 'overlookway'), ('eminence', 'eminenceway'),
('adhering', 'adheringway'), ('obtained', 'obtainedway'), ('openness',
'opennessway'), ('allusion', 'allusionway'), ('anything', 'anythingway'),
('overcome', 'overcomeway'), ('announce', 'announceway'), ('enjoying',
'enjoyingway'), ('outlived', 'outlivedway'), ('emphasis', 'emphasisway'),
('insulted', 'insultedway'), ('esteemed', 'esteemedway'), ('alphabet',
'alphabetway'), ('enquired', 'enquiredway'), ('accident', 'accidentway'),
('occupied', 'occupiedway'), ('ellisons', 'ellisonsway'), ('avoiding',
'avoidingway'), ('attained', 'attainedway'), ('unwarily', 'unwarilyway'),
('employed', 'employedway'), ('altering', 'alteringway'), ('intimate',
'intimateway'), ('effected', 'effectedway'), ('offences', 'offencesway'),
('entirely', 'entirelyway'), ('abounded', 'aboundedway'), ('actually',
'actuallyway'), ('identify', 'identifyway'), ('applying', 'applyingway'),
('exterior', 'exteriorway'), ('auditors', 'auditorsway'), ('unfairly',
'unfairlyway'), ('eighteen', 'eighteenway'), ('embraced', 'embracedway'),
('unfolded', 'unfoldedway'), ('injuries', 'injuriesway'), ('elevated',
'elevatedway'), ('alarming', 'alarmingway'), ('adequate', 'adequateway'),
('unabated', 'unabatedway'), ('incurred', 'incurredway'), ('uncommon',
'uncommonway'), ('unawares', 'unawaresway'), ('unsuited', 'unsuitedway'),
('instance', 'instanceway'), ('interval', 'intervalway'), ('audacity',
'audacityway')], (5, 8): [('other', 'otherway'), ('ideas', 'ideasway'),
('agony', 'agonyway'), ('avoid', 'avoidway'), ('awful', 'awfulway'), ('april',
'aprilway'), ('enter', 'enterway'), ('infer', 'inferway'), ('abuse',
'abuseway'), ('ivory', 'ivoryway'), ('adopt', 'adoptway'), ('image',
'imageway'), ('agree', 'agreeway'), ('alike', 'alikeway'), ('earth',
'earthway'), ('ended', 'endedway'), ('owner', 'ownerway'), ('using',
'usingway'), ('ought', 'oughtway'), ('abhor', 'abhorway'), ('altar',
'altarway'), ('often', 'oftenway'), ('atone', 'atoneway'), ('angel',
'angelway'), ('arise', 'ariseway'), ('under', 'underway'), ('abode',
'abodeway'), ('exert', 'exertway'), ('usual', 'usualway'), ('aches',
'achesway'), ('enjoy', 'enjoyway'), ('urged', 'urgedway'), ('admit',
'admitway'), ('aside', 'asideway'), ('equal', 'equalway'), ('alive',
'aliveway'), ('arose', 'aroseway'), ('eight', 'eightway'), ('opera',
'operaway'), ('ennui', 'ennuiway'), ('utter', 'utterway'), ('ankle',
'ankleway')], (2, 5): [('is', 'isway'), ('ay', 'ayway'), ('ah', 'ahway'), ('ii',
'iiway'), ('or', 'orway'), ('am', 'amway'), ('if', 'ifway'), ('an', 'anway')],
(12, 14): [('tranquillity', 'anquillitytray'), ('humiliations',
'umiliationshay'), ('contributing', 'ontributingcay'), ('significancy',
'ignificancysay'), ('wretchedness', 'etchednesswray'), ('prodigiously',

'odigiouslypray'), ('particularly', 'articularlypay'), ('carelessness', 'arelessnesscay'), ('refreshments', 'efreshmentsray'), ('satisfaction', 'atisfactionsay'), ('construction', 'onstructioncay'), ('contribution', 'ontributioncay'), ('gracefulness', 'acefulnessgray'), ('consequences', 'onsequencescay'), ('grandmothers', 'andmothersgray'), ('proportioned', 'oportionedpray'), ('perverseness', 'erversenesspay'), ('conciliation', 'onciliationcay'), ('representing', 'epresentingray'), ('watchfulness', 'atchfulnessway'), ('recollecting', 'ecollectingray'), ('hesitatingly', 'esitatinglyhay'), ('pleasantness', 'easantnessplay'), ('benefactress', 'enefactressbay'), ('distractedly', 'istractedlyday'), ('perseverance', 'erseverancepay'), ('performances', 'erformancespay'), ('disapproving', 'isapprovingday'), ('recollection', 'ecollectionray'), ('comprehended', 'omprehendedcay'), ('relinquished', 'elinquishedray'), ('conjecturing', 'onjecturingcay'), ('reservedness', 'eservednessray'), ('promontories', 'omontoriespray'), ('discontented', 'iscontentedday'), ('persecutions', 'ersecutionspay'), ('relationship', 'elationshipray'), ('triumphantly', 'iumphantlytray'), ('deliberation', 'eliberationday'), ('supplication', 'upplicationsay'), ('dishonorable', 'ishonorableday'), ('propensities', 'opensitiespray'), ('contemptuous', 'ontemptuouscay'), ('transporting', 'ansportingtray')], (13, 15): [('distinguished', 'istinguishedday'), ('resuscitation', 'esuscitationray'), ('particularity', 'articularitypay'), ('superannuated', 'uperannuatedsay'), ('disagreements', 'isagreementsday'), ('gentlemanlike', 'entlemanlikegay'), ('prepossessing', 'epossessingpray'), ('conscientious', 'onscientiouscay'), ('considerately', 'onsideratelycay'), ('felicitations', 'elicitationsfay'), ('prepossession', 'epossessionpray'), ('consternation', 'onsternationcay'), ('disproportion', 'isproportionday'), ('commiseration', 'ommiserationcay'), ('mortification', 'ortificationmay'), ('solicitations', 'olicitationssay'), ('qualification', 'alificationquay'), ('philanthropic', 'ilanthropicphay'), ('disappointing', 'isappointingday'), ('prescriptions', 'escriptionspray'), ('strengthening', 'engtheningstray')], (13, 17): [('mulberry-tree', 'ulberrymay-eetray'), ('good-humoured', 'oodgay-umouredhay'), ('mean-spirited', 'eanmay-iritedspay'), ('self-reproach', 'elfsay-eproachray'), ('drawing-table', 'awingdray-abletay'), ('well-disposed', 'ellway-isposedday'), ('proud-looking', 'oudpray-ookinglay'), ('flower-garden', 'owerflay-ardengay'), ('mansion-house', 'ansionmay-ousehay'), ('landing-place', 'andinglay-aceplay'), ('music-sellers', 'usicmay-ellerssay')], (4, 7): [('acts', 'actsway'), ('anne', 'anneway'), ('upon', 'uponway'), ('able', 'ableway'), ('inch', 'inchway'), ('ages', 'agesway'), ('envy', 'envyway'), ('idea', 'ideaway'), ('ends', 'endsway'), ('-and', '-andway'), ('eyed', 'eyedway'), ('arms', 'armsway'), ('over', 'overway'), ('easy', 'easyway'), ('ever', 'everway'), ('ours', 'oursway'), ('ears', 'earsway'), ('away', 'awayway')], (16, 20): [('pleasure-grounds', 'easureplay-oundsgray'), ('self-provocation', 'elfsay-ovocationpray')], (11, 15): [('bed-chamber', 'edbay-amberchay'), ('high-minded', 'ighhay-indedmay'), ('before-hand', 'eforebay-andhay'), ('watch-tower', 'atchway-owertay'), ('print-shops', 'intpray-opsshay'), ('card-tables', 'ardcay-ablestay'), ('post-horses', 'ostpay-orseshay'), ('good-nature', 'oodgay-aturenay'), ('dining-room', 'iningday-oomray'), ('high-church', 'ighhay-urchchay'), ('well-wisher', 'ellway-isherway'), ('bond-street', 'ondbay-

```
eetstray'), ('carpet-work', 'arpetcay-orkway')], (2, 4): [('to', 'otay'), ('he',
'ehay'), ('no', 'onay'), ('me', 'emay')], (12, 16): [('good-hearted', 'oodgay-
eartedhay'), ('well-judging', 'ellway-udgingjay'), ('drawing-room', 'awingdray-
oomray'), ('common-place', 'ommoncay-aceplay'), ('self-control', 'elfsay-
ontrolcay'), ('hand-writing', 'andhay-itingwray'), ('well-behaved', 'ellway-
ehavedbay'), ('mantel-piece', 'antelmay-iecepay')], (10, 15): [('ill-humour',
'illway-umourhay'), ('ill-judged', 'illway-udgedjay'), ('ill-suited', 'illway-
uitedsay')], (1, 3): [('c', 'cay')], (8, 13): [('ill-wind', 'illway-indway'),
('ill-will', 'illway-illway'), ('over-set', 'overway-etsay')], (17, 19):
[('disqualifications', 'isqualificationsday')], (9, 16): [('by-and-by', 'ybay-
andway-ybay')], (3, 6): [('aye', 'ayeway'), ('use', 'useway'), ('owe',
'oweway'), ('our', 'ourway'), ('old', 'oldway'), ('ear', 'earway'), ('its',
'itsway'), ('ill', 'illway')], (12, 17): [('ill-breeding', 'illway-eedingbray'),
('re-assurance', 'eray-assuranceway'), ('over-powered', 'overway-oweredpay'),
('ill-disposed', 'illway-isposedday')], (11, 16): [('ill-natured', 'illway-
aturednay')], (14, 16): [('reconciliation', 'econciliationray'),
('recommendation', 'ecommendationray'), ('reasonableness', 'easonablenessray'),
('disinclination', 'isinclinationday'), ('respectability', 'espectabilityray'),
('circumspection', 'ircumspectioncay'), ('considerations', 'onsiderationscay'),
('qualifications', 'alificationsquay'), ('congratulating', 'ongratulatingcay')],
(15, 17): [('dissatisfaction', 'issatisfactionday'), ('disrespectfully',
'isrespectfullyday'), ('conscientiously', 'onscientiouslycay'),
('misconstruction', 'isconstructionmay')], (15, 18): [('instantaneously',
'instantaneouslyway')], (15, 19): [('shooting-jacket', 'ootingshay-acketjay')],
(14, 21): [('sisters-in-law', 'isterssay-inway-awlay')], (17, 21): [('self-
condemnation', 'elfsay-ondemnationcay')], (13, 20): [('sister-in-law',
'istersay-inway-awlay')], (9, 14): [('ill-timed', 'illway-imedtay')], (13, 18):
[('ill-treatment', 'illway-eatmenttray'), ('old-fashioned', 'oldway-
ashionedfay'), ('self-interest', 'elfsay-interestway')], (16, 22): [('good-for-
nothing', 'oodgay-orfay-othingnay')], (9, 13): [('dove-cote', 'oveday-otecay'),
('tea-table', 'eatay-abletay'), ('work-bags', 'orkway-agsbay')], (1, 4): [('i',
'iway')], (16, 19): [('incomprehensible', 'incomprehensibleway')], (8, 12):
[('to-night', 'otay-ightnay'), ('land-tax', 'andlay-axtay')], (11, 18): [('tete-
a-tete', 'etetay-away-etetay')], (5, 9): [('to-do', 'otay-oday')]})
Moved models to GPU!
Epoch:   0 | Flops (T): 0.014807758752, Tokens (M) 0.052678, Train Loss 2.995,
Val Loss 2.921 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   1 | Flops (T): 0.029615517504, Tokens (M) 0.105356, Train Loss 2.875,
Val Loss 2.879 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   2 | Flops (T): 0.044423276256, Tokens (M) 0.158034, Train Loss 2.832,
Val Loss 2.829 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   3 | Flops (T): 0.059231035008, Tokens (M) 0.210712, Train Loss 2.788,
Val Loss 2.787 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   4 | Flops (T): 0.07403879376, Tokens (M) 0.26339, Train Loss 2.750, Val
Loss 2.787 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   5 | Flops (T): 0.088846552512, Tokens (M) 0.316068, Train Loss 2.727,
Val Loss 2.756 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   6 | Flops (T): 0.103654311264, Tokens (M) 0.368746, Train Loss 2.525,
```

```
Val Loss 2.475 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   7 | Flops (T): 0.118462070016, Tokens (M) 0.421424, Train Loss 2.267,
Val Loss 2.295 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   8 | Flops (T): 0.133269828768, Tokens (M) 0.474102, Train Loss 2.173,
Val Loss 2.220 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   9 | Flops (T): 0.14807758752, Tokens (M) 0.52678, Train Loss 2.123, Val
Loss 2.163 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  10 | Flops (T): 0.162885346272, Tokens (M) 0.579458, Train Loss 2.089,
Val Loss 2.103 | Gen: affffffffffffffffff
Epoch:  11 | Flops (T): 0.177693105024, Tokens (M) 0.632136, Train Loss 2.068,
Val Loss 2.116 | Gen: ffffffffffffffffff
Epoch:  12 | Flops (T): 0.192500863776, Tokens (M) 0.684814, Train Loss 2.076,
Val Loss 2.111 | Gen: ffffffffffffffffff
Epoch:  13 | Flops (T): 0.207308622528, Tokens (M) 0.737492, Train Loss 2.072,
Val Loss 2.138 | Gen: ffffffffffffffffff
Epoch:  14 | Flops (T): 0.22211638128, Tokens (M) 0.79017, Train Loss 2.063, Val
Loss 2.076 | Gen: ffffffffffffffffff
Epoch:  15 | Flops (T): 0.236924140032, Tokens (M) 0.842848, Train Loss 2.001,
Val Loss 2.072 | Gen: ffffffffffffffffff
Epoch:  16 | Flops (T): 0.251731898784, Tokens (M) 0.895526, Train Loss 2.011,
Val Loss 2.053 | Gen: affffffffffffffffff
Epoch:  17 | Flops (T): 0.266539657536, Tokens (M) 0.948204, Train Loss 1.982,
Val Loss 2.024 | Gen: affffffffffffffffff
Epoch:  18 | Flops (T): 0.281347416288, Tokens (M) 1.000882, Train Loss 1.964,
Val Loss 2.015 | Gen: affffffffffffffffff
Epoch:  19 | Flops (T): 0.29615517504, Tokens (M) 1.05356, Train Loss 2.013, Val
Loss 2.089 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  20 | Flops (T): 0.310962933792, Tokens (M) 1.106238, Train Loss 1.952,
Val Loss 1.949 | Gen: ffffffffffffffffff
Epoch:  21 | Flops (T): 0.325770692544, Tokens (M) 1.158916, Train Loss 1.909,
Val Loss 1.952 | Gen: ffffffffffffffffff
Epoch:  22 | Flops (T): 0.340578451296, Tokens (M) 1.211594, Train Loss 1.886,
Val Loss 1.904 | Gen: affffffffffffffffff
Epoch:  23 | Flops (T): 0.355386210048, Tokens (M) 1.264272, Train Loss 1.831,
Val Loss 1.876 | Gen: affffffffffffffffff
Epoch:  24 | Flops (T): 0.3701939688, Tokens (M) 1.31695, Train Loss 1.798, Val
Loss 1.844 | Gen: ffffffffffffffffff
Epoch:  25 | Flops (T): 0.385001727552, Tokens (M) 1.369628, Train Loss 1.781,
Val Loss 1.814 | Gen: ffffffffffffffffff
Epoch:  26 | Flops (T): 0.399809486304, Tokens (M) 1.422306, Train Loss 1.747,
Val Loss 1.808 | Gen: ffffffffffffffffff
Epoch:  27 | Flops (T): 0.414617245056, Tokens (M) 1.474984, Train Loss 1.727,
Val Loss 1.774 | Gen: ffffffffffffffffff
Epoch:  28 | Flops (T): 0.429425003808, Tokens (M) 1.527662, Train Loss 1.711,
Val Loss 1.773 | Gen: ffffffffffffffffff
Epoch:  29 | Flops (T): 0.44423276256, Tokens (M) 1.58034, Train Loss 1.704, Val
Loss 1.790 | Gen: ffffffffffffffffff
Epoch:  30 | Flops (T): 0.459040521312, Tokens (M) 1.633018, Train Loss 1.687,
```

```
Val Loss 1.769 | Gen: fffffffffffffffffff
Epoch:  31 | Flops (T): 0.473848280064, Tokens (M) 1.685696, Train Loss 1.680,
Val Loss 1.808 | Gen: fffffffffffffffffff
Epoch:  32 | Flops (T): 0.488656038816, Tokens (M) 1.738374, Train Loss 1.691,
Val Loss 1.755 | Gen: fffffffffffffffffff
Epoch:  33 | Flops (T): 0.503463797568, Tokens (M) 1.791052, Train Loss 1.674,
Val Loss 1.737 | Gen: fffffffffffffffffff
Epoch:  34 | Flops (T): 0.51827155632, Tokens (M) 1.84373, Train Loss 1.636, Val
Loss 1.708 | Gen: fffffffffffffffffff
Epoch:  35 | Flops (T): 0.533079315072, Tokens (M) 1.896408, Train Loss 1.633,
Val Loss 1.709 | Gen: fffffffffffffffffff
Epoch:  36 | Flops (T): 0.547887073824, Tokens (M) 1.949086, Train Loss 1.607,
Val Loss 1.677 | Gen: fffffffffffffffffff
Epoch:  37 | Flops (T): 0.562694832576, Tokens (M) 2.001764, Train Loss 1.577,
Val Loss 1.778 | Gen: fffffffffffffffffff
Epoch:  38 | Flops (T): 0.577502591328, Tokens (M) 2.054442, Train Loss 1.536,
Val Loss 1.735 | Gen: fffffffffffffffffff
Epoch:  39 | Flops (T): 0.59231035008, Tokens (M) 2.10712, Train Loss 1.489, Val
Loss 1.729 | Gen: rrrrrrrrrrrrrrrrrrr
Epoch:  40 | Flops (T): 0.607118108832, Tokens (M) 2.159798, Train Loss 1.458,
Val Loss 1.568 | Gen: rrrrrrrrrrrrrrrrrrr
Epoch:  41 | Flops (T): 0.621925867584, Tokens (M) 2.212476, Train Loss 1.388,
Val Loss 1.563 | Gen: rrrrrrrrrrrrrrrrrrr
Epoch:  42 | Flops (T): 0.636733626336, Tokens (M) 2.265154, Train Loss 1.438,
Val Loss 1.862 | Gen: rrrrrrrrrrrrrrrrrrr
Epoch:  43 | Flops (T): 0.651541385088, Tokens (M) 2.317832, Train Loss 1.422,
Val Loss 1.518 | Gen: fffffffffffffffffff
Epoch:  44 | Flops (T): 0.66634914384, Tokens (M) 2.37051, Train Loss 1.348, Val
Loss 1.713 | Gen: fffffffffffffffffff
Epoch:  45 | Flops (T): 0.681156902592, Tokens (M) 2.423188, Train Loss 1.333,
Val Loss 1.373 | Gen: fffffffffffffffffff
Epoch:  46 | Flops (T): 0.695964661344, Tokens (M) 2.475866, Train Loss 1.296,
Val Loss 1.383 | Gen: oooooooooooooooooooo
Epoch:  47 | Flops (T): 0.710772420096, Tokens (M) 2.528544, Train Loss 1.282,
Val Loss 1.329 | Gen: rrrrrrrrrrrrrrrrrrr
Epoch:  48 | Flops (T): 0.725580178848, Tokens (M) 2.581222, Train Loss 1.256,
Val Loss 1.483 | Gen: rrrrrrrrrrrrrrrrrrr
Epoch:  49 | Flops (T): 0.7403879376, Tokens (M) 2.6339, Train Loss 1.250, Val
Loss 1.436 | Gen: rrrrrrrrrrrrrrrrrrr
Epoch:  50 | Flops (T): 0.755195696352, Tokens (M) 2.686578, Train Loss 1.235,
Val Loss 1.323 | Gen: rrrrrrrrrrrrrrrrrrr
Epoch:  51 | Flops (T): 0.770003455104, Tokens (M) 2.739256, Train Loss 1.232,
Val Loss 1.414 | Gen: rrrrrrrrrrrrrrrrrrr
Epoch:  52 | Flops (T): 0.784811213856, Tokens (M) 2.791934, Train Loss 1.272,
Val Loss 1.370 | Gen: oooooooooooooooooooo
Epoch:  53 | Flops (T): 0.799618972608, Tokens (M) 2.844612, Train Loss 1.245,
Val Loss 1.342 | Gen: oooooooooooooooooooo
Epoch:  54 | Flops (T): 0.81442673136, Tokens (M) 2.89729, Train Loss 1.226, Val
```

```
Loss 1.359 | Gen: fffffffffffffffffffff
Epoch:  55 | Flops (T): 0.829234490112, Tokens (M) 2.949968, Train Loss 1.191,
Val Loss 1.201 | Gen: oooooooooooooooooooo
Epoch:  56 | Flops (T): 0.844042248864, Tokens (M) 3.002646, Train Loss 1.161,
Val Loss 1.203 | Gen: rrrrrrrrrrrrrrrrrrrr
Epoch:  57 | Flops (T): 0.858850007616, Tokens (M) 3.055324, Train Loss 1.130,
Val Loss 1.331 | Gen: rrrrrrrrrrrrrrrrrrrr
Epoch:  58 | Flops (T): 0.873657766368, Tokens (M) 3.108002, Train Loss 1.151,
Val Loss 1.398 | Gen: oooooooooooooooooooo
Epoch:  59 | Flops (T): 0.88846552512, Tokens (M) 3.16068, Train Loss 1.168, Val
Loss 1.269 | Gen: rrrrrrrrrrrrrrrrrrrr
Epoch:  60 | Flops (T): 0.903273283872, Tokens (M) 3.213358, Train Loss 1.113,
Val Loss 1.385 | Gen: rrrrrrrrrrrrrrrrrrrr
Epoch:  61 | Flops (T): 0.918081042624, Tokens (M) 3.266036, Train Loss 1.104,
Val Loss 1.091 | Gen: oooooooooooooooooooo
Epoch:  62 | Flops (T): 0.932888801376, Tokens (M) 3.318714, Train Loss 1.048,
Val Loss 1.155 | Gen: oooooooooooooooooooo
Epoch:  63 | Flops (T): 0.947696560128, Tokens (M) 3.371392, Train Loss 1.041,
Val Loss 1.183 | Gen: oooooooooooooooooooo
Epoch:  64 | Flops (T): 0.96250431888, Tokens (M) 3.42407, Train Loss 1.020, Val
Loss 1.128 | Gen: oooooooooooooooooooo
Epoch:  65 | Flops (T): 0.977312077632, Tokens (M) 3.476748, Train Loss 0.986,
Val Loss 1.057 | Gen: oooooooooooooooooooo
Epoch:  66 | Flops (T): 0.992119836384, Tokens (M) 3.529426, Train Loss 0.975,
Val Loss 1.098 | Gen: oooooooooooooooooooo
Epoch:  67 | Flops (T): 1.006927595136, Tokens (M) 3.582104, Train Loss 0.981,
Val Loss 1.053 | Gen: oooooooooooooooooooo
Epoch:  68 | Flops (T): 1.021735353888, Tokens (M) 3.634782, Train Loss 0.946,
Val Loss 1.011 | Gen: oooooooooooooooooooo
Epoch:  69 | Flops (T): 1.03654311264, Tokens (M) 3.68746, Train Loss 0.932, Val
Loss 1.047 | Gen: oooooooooooooooooooo
Epoch:  70 | Flops (T): 1.051350871392, Tokens (M) 3.740138, Train Loss 0.937,
Val Loss 1.106 | Gen: oooooooooooooooooooo
Epoch:  71 | Flops (T): 1.066158630144, Tokens (M) 3.792816, Train Loss 0.920,
Val Loss 1.070 | Gen: oooooooooooooooooooo
Epoch:  72 | Flops (T): 1.080966388896, Tokens (M) 3.845494, Train Loss 0.904,
Val Loss 1.070 | Gen: oooooooooooooooooooo
Epoch:  73 | Flops (T): 1.095774147648, Tokens (M) 3.898172, Train Loss 0.875,
Val Loss 1.047 | Gen: oooooooooooooooooooo
Epoch:  74 | Flops (T): 1.1105819064, Tokens (M) 3.95085, Train Loss 0.885, Val
Loss 1.236 | Gen: oooooooooooooooooooo
Epoch:  75 | Flops (T): 1.125389665152, Tokens (M) 4.003528, Train Loss 0.896,
Val Loss 1.010 | Gen: oooooooooooooooooooo
Epoch:  76 | Flops (T): 1.140197423904, Tokens (M) 4.056206, Train Loss 0.855,
Val Loss 1.026 | Gen: oooooooooooooooooooo
Epoch:  77 | Flops (T): 1.155005182656, Tokens (M) 4.108884, Train Loss 0.861,
Val Loss 0.960 | Gen: ooooooooooooooooooof
Epoch:  78 | Flops (T): 1.169812941408, Tokens (M) 4.161562, Train Loss 0.829,
```

```
Val Loss 0.926 | Gen: oooooooooooooooooooo
Epoch:  79 | Flops (T): 1.18462070016, Tokens (M) 4.21424, Train Loss 0.807, Val
Loss 0.994 | Gen: oooooooooooooooooooo
Epoch:  80 | Flops (T): 1.199428458912, Tokens (M) 4.266918, Train Loss 0.817,
Val Loss 1.046 | Gen: ffffffffffffffffffff
Epoch:  81 | Flops (T): 1.214236217664, Tokens (M) 4.319596, Train Loss 0.829,
Val Loss 0.962 | Gen: cffffffffffffffffff
Epoch:  82 | Flops (T): 1.229043976416, Tokens (M) 4.372274, Train Loss 0.792,
Val Loss 0.976 | Gen: ffffffffffffffffffff
Epoch:  83 | Flops (T): 1.243851735168, Tokens (M) 4.424952, Train Loss 0.791,
Val Loss 0.989 | Gen: ffffffffffffffffffff
Epoch:  84 | Flops (T): 1.25865949392, Tokens (M) 4.47763, Train Loss 0.777, Val
Loss 1.016 | Gen: cffffffffffffffffff
Epoch:  85 | Flops (T): 1.273467252672, Tokens (M) 4.530308, Train Loss 0.813,
Val Loss 1.109 | Gen: oooooooooooooooooooo
Epoch:  86 | Flops (T): 1.288275011424, Tokens (M) 4.582986, Train Loss 0.822,
Val Loss 0.936 | Gen: rrrrrrrrrrrrrrrrrrr
Epoch:  87 | Flops (T): 1.303082770176, Tokens (M) 4.635664, Train Loss 0.761,
Val Loss 0.908 | Gen: oooooooooooooooooooo
Epoch:  88 | Flops (T): 1.317890528928, Tokens (M) 4.688342, Train Loss 0.744,
Val Loss 0.833 | Gen: oooooooooooooooooooo
Epoch:  89 | Flops (T): 1.33269828768, Tokens (M) 4.74102, Train Loss 0.731, Val
Loss 1.035 | Gen: oooooooooooooooooooo
Epoch:  90 | Flops (T): 1.347506046432, Tokens (M) 4.793698, Train Loss 0.750,
Val Loss 0.840 | Gen: oooooooooooooooooooo
Epoch:  91 | Flops (T): 1.362313805184, Tokens (M) 4.846376, Train Loss 0.725,
Val Loss 0.829 | Gen: oooooooooooooooooooo
Epoch:  92 | Flops (T): 1.377121563936, Tokens (M) 4.899054, Train Loss 0.715,
Val Loss 0.826 | Gen: oooooooooooooooooooo
Epoch:  93 | Flops (T): 1.391929322688, Tokens (M) 4.951732, Train Loss 0.706,
Val Loss 0.847 | Gen: oooooooooooooooooooo
Epoch:  94 | Flops (T): 1.40673708144, Tokens (M) 5.00441, Train Loss 0.714, Val
Loss 0.858 | Gen: oooooooooooooooofff
Epoch:  95 | Flops (T): 1.421544840192, Tokens (M) 5.057088, Train Loss 0.692,
Val Loss 0.849 | Gen: oooooooooooooooooooo
Epoch:  96 | Flops (T): 1.436352598944, Tokens (M) 5.109766, Train Loss 0.707,
Val Loss 0.815 | Gen: oooooooooooooooooooo
Epoch:  97 | Flops (T): 1.451160357696, Tokens (M) 5.162444, Train Loss 0.694,
Val Loss 0.861 | Gen: ooooooooooooooooooff
Epoch:  98 | Flops (T): 1.465968116448, Tokens (M) 5.215122, Train Loss 0.678,
Val Loss 0.794 | Gen: ooooooooooooooooooff
Epoch:  99 | Flops (T): 1.4807758752, Tokens (M) 5.2678, Train Loss 0.656, Val
Loss 0.801 | Gen: oooooooooooooooooooof
Obtained lowest validation loss of: 0.7937000574067582
Avg tokens per example (Seq Len): 20.59343236903831, Avg flops per example:
5788803.265050821
```

[ ]:
```
TEST_SENTENCE = "floor"
trans64_args_s = AttrDict()
args_dict_64 = {
    "data_file_name": "pig_latin_small", #pig_latin_small debug
    "cuda": use_cuda,
    "nepochs": 100,
    "checkpoint_dir": "checkpoints",
    "learning_rate": 5e-4,
    "early_stopping_patience": 100,
    "lr_decay": 0.99,
    "batch_size": 64,
    "hidden_size": 64,
    "num_transformer_layers": 4,
    "loss_seq_span": "target_only",
    "num_layers": 4,
    "vocab_size": 30
}
trans64_args_s.update(args_dict_64)
print_opts(trans64_args_s)
trans64_decoder_s, trans64_losses_s, idx_dict = train(trans64_args_s)
```

```
================================================================================
                                       Opts
--------------------------------------------------------------------------------
                         data_file_name: pig_latin_small
                                   cuda: 1
                                 nepochs: 100
                         checkpoint_dir: checkpoints
                          learning_rate: 0.0005
                early_stopping_patience: 100
                                lr_decay: 0.99
                              batch_size: 64
                             hidden_size: 64
                 num_transformer_layers: 4
                          loss_seq_span: target_only
                              num_layers: 4
                              vocab_size: 30
================================================================================
================================================================================
                                   Data Stats
--------------------------------------------------------------------------------
('irresistible', 'irresistibleway')
('repentance', 'epentanceray')
('pretty', 'ettypray')
('parcels', 'arcelspay')
('survey', 'urveysay')
Num unique word pairs: 3198
```

```
Vocabulary: dict_keys(['-', 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j',
'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z',
'SOS', 'EOS', 'EOP'])
Vocab size: 30
================================================================================
defaultdict(<class 'list'>, {(12, 15): [('irresistible', 'irresistibleway'),
('advantageous', 'advantageousway'), ('impartiality', 'impartialityway'),
('interruption', 'interruptionway'), ('anticipation', 'anticipationway'),
('impenetrable', 'impenetrableway'), ('excellencies', 'excellenciesway'),
('intermission', 'intermissionway'), ('interspersed', 'interspersedway'),
('unaffectedly', 'unaffectedlyway'), ('insinuations', 'insinuationsway'),
('unpardonable', 'unpardonableway'), ('arrangements', 'arrangementsway'),
('unexpectedly', 'unexpectedlyway'), ('introduction', 'introductionway'),
('occasionally', 'occasionallyway'), ('establishing', 'establishingway'),
('unanswerable', 'unanswerableway'), ('independence', 'independenceway'),
('inclinations', 'inclinationsway'), ('instructions', 'instructionsway'),
('undoubtingly', 'undoubtinglyway'), ('unfathomable', 'unfathomableway'),
('impoverished', 'impoverishedway'), ('outstretched', 'outstretchedway'),
('unrestrained', 'unrestrainedway'), ('ungraciously', 'ungraciouslyway'),
('imperfection', 'imperfectionway'), ('unreasonably', 'unreasonablyway'),
('interference', 'interferenceway'), ('inconvenient', 'inconvenientway'),
('enthusiastic', 'enthusiasticway'), ('extinguished', 'extinguishedway'),
('entanglement', 'entanglementway'), ('affectionate', 'affectionateway')], (10,
12): [('repentance', 'epentanceray'), ('capability', 'apabilitycay'),
('beneficial', 'eneficialbay'), ('retrenched', 'etrenchedray'), ('devonshire',
'evonshireday'), ('palanquins', 'alanquinspay'), ('contribute', 'ontributecay'),
('minuteness', 'inutenessmay'), ('dissimilar', 'issimilarday'), ('flattering',
'atteringflay'), ('confounded', 'onfoundedcay'), ('compromise', 'ompromisecay'),
('contrasted', 'ontrastedcay'), ('precaution', 'ecautionpray'), ('fastidious',
'astidiousfay'), ('blackguard', 'ackguardblay'), ('heroically', 'eroicallyhay'),
('lengthened', 'engthenedlay'), ('pocketbook', 'ocketbookpay'), ('companions',
'ompanionscay'), ('displeased', 'ispleasedday'), ('pianoforte', 'ianofortepay'),
('succession', 'uccessionsay'), ('composedly', 'omposedlycay'), ('pronounced',
'onouncedpray'), ('honourable', 'onourablehay'), ('solicitude', 'olicitudesay'),
('justifying', 'ustifyingjay'), ('collecting', 'ollectingcay'), ('headstrong',
'eadstronghay'), ('forbidding', 'orbiddingfay'), ('successful', 'uccessfulsay'),
('confidence', 'onfidencecay'), ('prescribed', 'escribedpray'), ('scrutinies',
'utiniesscray'), ('melancholy', 'elancholymay'), ('handsomest', 'andsomesthay'),
('prescience', 'esciencepray'), ('conclusion', 'onclusioncay'), ('remarkably',
'emarkablyray'), ('gratefully', 'atefullygray'), ('michaelmas', 'ichaelmasmay'),
('containing', 'ontainingcay'), ('discussion', 'iscussionday'), ('resentment',
'esentmentray'), ('restraints', 'estraintsray'), ('tablecloth', 'ableclothtay'),
('projecting', 'ojectingpray'), ('frightened', 'ightenedfray'), ('pardonable',
'ardonablepay'), ('variations', 'ariationsvay'), ('sensations', 'ensationssay'),
('rheumatism', 'eumatismrhay'), ('dispirited', 'ispiritedday'), ('politeness',
'olitenesspay'), ('suggesting', 'uggestingsay'), ('willoughby', 'illoughbyway'),
('reasonable', 'easonableray'), ('discerning', 'iscerningday'), ('previously',
'eviouslypray'), ('passionate', 'assionatepay'), ('prophecies', 'opheciespray'),
```

('possessing', 'ossessingpay'), ('perfection', 'erfectionpay'), ('heightened',
'eightenedhay'), ('generously', 'enerouslygay'), ('faithfully', 'aithfullyfay'),
('reconciled', 'econciledray'), ('distressed', 'istressedday'), ('respective',
'espectiveray'), ('liberality', 'iberalitylay'), ('disengaged', 'isengagedday'),
('honourably', 'onourablyhay'), ('persuading', 'ersuadingpay'), ('cultivated',
'ultivatedcay'), ('projection', 'ojectionpray'), ('disapprove', 'isapproveday'),
('conciliate', 'onciliatecay'), ('satisfying', 'atisfyingsay'), ('dispatches',
'ispatchesday'), ('repeatedly', 'epeatedlyray'), ('suspicions', 'uspicionssay'),
('perplexity', 'erplexitypay'), ('strengthen', 'engthenstray'), ('dependence',
'ependenceday'), ('dissipated', 'issipatedday'), ('conformity', 'onformitycay'),
('hysterical', 'ystericalhay'), ('throughout', 'oughoutthray'), ('sacrificed',
'acrificedsay'), ('dimensions', 'imensionsday'), ('flatteries', 'atteriesflay'),
('completely', 'ompletelycay'), ('reproaches', 'eproachesray'), ('contriving',
'ontrivingcay'), ('consequent', 'onsequentcay'), ('liveliness', 'ivelinesslay'),
('resistance', 'esistanceray'), ('caricature', 'aricaturecay'), ('stimulated',
'imulatedstay'), ('brightened', 'ightenedbray'), ('prevailing', 'evailingpray'),
('disclosure', 'isclosureday'), ('bequeathed', 'equeathedbay'), ('disappoint',
'isappointday'), ('complexion', 'omplexioncay'), ('dismounted', 'ismountedday'),
('remembered', 'ememberedray'), ('delineated', 'elineatedday'), ('subsequent',
'ubsequentsay'), ('persevered', 'erseveredpay'), ('doubtingly', 'oubtinglyday'),
('characters', 'aracterschay'), ('neighbours', 'eighboursnay'), ('witticisms',
'itticismsway'), ('materially', 'ateriallymay'), ('playthings', 'aythingsplay'),
('suspecting', 'uspectingsay'), ('conditions', 'onditionscay'), ('discovered',
'iscoveredday'), ('deficiency', 'eficiencyday'), ('regularity', 'egularityray'),
('subjection', 'ubjectionsay'), ('brightness', 'ightnessbray'), ('pleasantly',
'easantlyplay'), ('delicacies', 'elicaciesday'), ('preference', 'eferencepray'),
('difference', 'ifferenceday'), ('compassion', 'ompassioncay'), ('competence',
'ompetencecay'), ('generosity', 'enerositygay'), ('retirement', 'etirementray'),
('strikingly', 'ikinglystray'), ('subsisting', 'ubsistingsay'), ('conscience',
'onsciencecay'), ('diabolical', 'iabolicalday'), ('temptation', 'emptationtay'),
('formidable', 'ormidablefay'), ('reconsider', 'econsiderray'), ('prejudiced',
'ejudicedpray'), ('comprehend', 'omprehendcay'), ('compliment', 'omplimentcay'),
('sufferings', 'ufferingssay'), ('concluding', 'oncludingcay'), ('compliance',
'ompliancecay'), ('mosquitoes', 'osquitoesmay'), ('thoughtful', 'oughtfulthay'),
('bitterness', 'itternessbay'), ('mentioning', 'entioningmay'), ('toleration',
'olerationtay'), ('profession', 'ofessionpray'), ('discompose', 'iscomposeday'),
('waistcoats', 'aistcoatsway'), ('reflecting', 'eflectingray'), ('positively',
'ositivelypay'), ('diminished', 'iminishedday')], (6, 8): [('pretty',
'ettypray'), ('survey', 'urveysay'), ('demure', 'emureday'), ('secure',
'ecuresay'), ('sobbed', 'obbedsay'), ('moment', 'omentmay'), ('guilty',
'uiltygay'), ('combat', 'ombatcay'), ('modern', 'odernmay'), ('father',
'atherfay'), ('horses', 'orseshay'), ('closer', 'oserclay'), ('nicety',
'icetynay'), ('shaken', 'akenshay'), ('centre', 'entrecay'), ('powers',
'owerspay'), ('simple', 'implesay'), ('beings', 'eingsbay'), ('recent',
'ecentray'), ('sketch', 'etchskay'), ('loving', 'ovinglay'), ('dagger',
'aggerday'), ('hearts', 'eartshay'), ('copied', 'opiedcay'), ('gazing',
'azinggay'), ('sooner', 'oonersay'), ('disarm', 'isarmday'), ('suited',
'uitedsay'), ('doubts', 'oubtsday'), ('lifted', 'iftedlay'), ('vacant',

'acantvay'), ('palmer', 'almerpay'), ('grudge', 'udgegray'), ('folded',
'oldedfay'), ('remove', 'emoveray'), ('borrow', 'orrowbay'), ('scheme',
'emeschay'), ('solemn', 'olemnsay'), ('denote', 'enoteday'), ('bustle',
'ustlebay'), ('turned', 'urnedtay'), ('mutton', 'uttonmay'), ('tender',
'endertay'), ('hearer', 'earerhay'), ('begins', 'eginsbay'), ('taught',
'aughttay'), ('honour', 'onourhay'), ('robert', 'obertray'), ('gibson',
'ibsongay'), ('virtue', 'irtuevay'), ('vanish', 'anishvay'), ('writes',
'iteswray'), ('silent', 'ilentsay'), ('singly', 'inglysay'), ('jargon',
'argonjay'), ('grieve', 'ievegray'), ('design', 'esignday'), ('linger',
'ingerlay'), ('bathed', 'athedbay'), ('pardon', 'ardonpay'), ('rumour',
'umourray'), ('dances', 'ancesday'), ('remark', 'emarkray'), ('worded',
'ordedway'), ('honest', 'onesthay'), ('chosen', 'osenchay'), ('boyish',
'oyishbay'), ('stress', 'essstray'), ('caught', 'aughtcay'), ('potent',
'otentpay'), ('during', 'uringday'), ('stocks', 'ocksstay'), ('relics',
'elicsray'), ('tongue', 'onguetay'), ('street', 'eetstray'), ('whiled',
'iledwhay'), ('manner', 'annermay'), ('truths', 'uthstray'), ('travel',
'aveltray'), ('couple', 'ouplecay'), ('gaiety', 'aietygay'), ('giving',
'ivinggay'), ('county', 'ountycay'), ('cramps', 'ampscray'), ('wealth',
'ealthway'), ('divine', 'ivineday'), ('behold', 'eholdbay'), ('feared',
'earedfay'), ('sparks', 'arksspay'), ('finger', 'ingerfay'), ('talent',
'alenttay'), ('demand', 'emandday'), ('coffee', 'offeecay'), ('merest',
'erestmay'), ('detail', 'etailday'), ('casino', 'asinocay'), ('choose',
'oosechay'), ('warmed', 'armedway'), ('relate', 'elateray'), ('sashes',
'ashessay'), ('chance', 'ancechay'), ('rushed', 'ushedray'), ('health',
'ealthhay'), ('preyed', 'eyedpray'), ('mutual', 'utualmay'), ('months',
'onthsmay'), ('raises', 'aisesray'), ('things', 'ingsthay'), ('master',
'astermay'), ('thanks', 'anksthay'), ('cheeks', 'eekschay'), ('hazard',
'azardhay'), ('slight', 'ightslay'), ('guests', 'uestsgay'), ('halves',
'alveshay'), ('lessen', 'essenlay'), ('notice', 'oticenay'), ('higher',
'igherhay'), ('truest', 'uesttray'), ('porter', 'orterpay'), ('fickle',
'icklefay'), ('behave', 'ehavebay'), ('credit', 'editcray'), ('number',
'umbernay'), ('mother', 'othermay'), ('tricks', 'ickstray'), ('traced',
'acedtray'), ('subdue', 'ubduesay'), ('ruined', 'uinedray'), ('famous',
'amousfay'), ('danced', 'ancedday'), ('clever', 'everclay'), ('drives',
'ivesdray'), ('resist', 'esistray'), ('forget', 'orgetfay'), ('stable',
'ablestay'), ('bigger', 'iggerbay'), ('betray', 'etraybay'), ('senses',
'ensessay'), ('smiled', 'iledsmay'), ('hating', 'atinghay'), ('bolder',
'olderbay'), ('banish', 'anishbay'), ('remain', 'emainray'), ('period',
'eriodpay'), ('gossip', 'ossipgay'), ('basket', 'asketbay'), ('genius',
'eniusgay'), ('seized', 'eizedsay'), ('widely', 'idelyway'), ('pursue',
'ursuepay'), ('favour', 'avourfay'), ('sighed', 'ighedsay'), ('plants',
'antsplay'), ('proved', 'ovedpray'), ('season', 'easonsay'), ('circle',
'irclecay'), ('taking', 'akingtay'), ('screen', 'eenscray'), ('talker',
'alkertay'), ('teazed', 'eazedtay'), ('suffer', 'uffersay'), ('curate',
'uratecay'), ('traits', 'aitstray'), ('defect', 'efectday'), ('called',
'alledcay'), ('dining', 'iningday'), ('direct', 'irectday'), ('dinner',
'innerday'), ('joking', 'okingjay'), ('whiter', 'iterwhay'), ('limits',
'imitslay'), ('papers', 'aperspay'), ('public', 'ublicpay'), ('losing',

'osinglay'), ('handed', 'andedhay'), ('wisest', 'isestway'), ('delays',
'elaysday'), ('leaves', 'eaveslay'), ('points', 'ointspay'), ('hither',
'itherhay'), ('horrid', 'orridhay'), ('middle', 'iddlemay'), ('vastly',
'astlyvay'), ('luxury', 'uxurylay'), ('failed', 'ailedfay'), ('lodges',
'odgeslay'), ('female', 'emalefay'), ('walked', 'alkedway'), ('nipped',
'ippednay'), ('nephew', 'ephewnay'), ('settle', 'ettlesay'), ('kinder',
'inderkay'), ('bowing', 'owingbay'), ('smooth', 'oothsmay'), ('roared',
'oaredray'), ('wiping', 'ipingway'), ('jewels', 'ewelsjay'), ('breath',
'eathbray'), ('steady', 'eadystay'), ('breach', 'eachbray'), ('polite',
'olitepay'), ('colder', 'oldercay'), ('though', 'oughthay'), ('poured',
'ouredpay'), ('paused', 'ausedpay'), ('nought', 'oughtnay'), ('shrewd',
'ewdshray'), ('fourth', 'ourthfay'), ('parish', 'arishpay'), ('traded',
'adedtray'), ('glance', 'anceglay'), ('lesser', 'esserlay'), ('curacy',
'uracycay'), ('labour', 'abourlay'), ('became', 'ecamebay'), ('decide',
'ecideday'), ('kisses', 'isseskay'), ('styled', 'yledstay'), ('sprung',
'ungspray'), ('priory', 'iorypray'), ('derive', 'eriveday'), ('school',
'oolschay'), ('liking', 'ikinglay'), ('dawned', 'awnedday'), ('weaken',
'eakenway'), ('winter', 'interway'), ('rivals', 'ivalsray'), ('sallow',
'allowsay'), ('defend', 'efendday'), ('seated', 'eatedsay'), ('penury',
'enurypay'), ('repair', 'epairray'), ('chaise', 'aisechay'), ('church',
'urchchay'), ('beyond', 'eyondbay'), ('wisdom', 'isdomway'), ('retain',
'etainray'), ('toilet', 'oilettay'), ('talked', 'alkedtay'), ('dearly',
'earlyday'), ('reward', 'ewardray'), ('safely', 'afelysay'), ('making',
'akingmay'), ('debate', 'ebateday'), ('future', 'uturefay'), ('broken',
'okenbray'), ('begged', 'eggedbay'), ('thirty', 'irtythay'), ('muslin',
'uslinmay'), ('marked', 'arkedmay'), ('hinder', 'inderhay'), ('beheld',
'eheldbay'), ('maxims', 'aximsmay'), ('theirs', 'eirsthay'), ('corner',
'ornercay'), ('piqued', 'iquedpay'), ('rested', 'estedray'), ('needed',
'eedednay'), ('fitted', 'ittedfay'), ('candid', 'andidcay'), ('laughs',
'aughslay'), ('window', 'indowway'), ('tastes', 'astestay'), ('spread',
'eadspray'), ('hollow', 'ollowhay'), ('sorrow', 'orrowsay'), ('partly',
'artlypay'), ('served', 'ervedsay'), ('flowed', 'owedflay'), ('remedy',
'emedyray'), ('trying', 'yingtray'), ('latest', 'atestlay'), ('manage',
'anagemay'), ('gained', 'ainedgay'), ('plenty', 'entyplay'), ('garden',
'ardengay'), ('frosts', 'ostsfray'), ('thirds', 'irdsthay'), ('handle',
'andlehay'), ('verbal', 'erbalvay'), ('sheath', 'eathshay'), ('covert',
'overtcay'), ('darker', 'arkerday'), ('harris', 'arrishay'), ('houses',
'ouseshay'), ('nature', 'aturenay'), ('thomas', 'omasthay'), ('submit',
'ubmitsay'), ('danger', 'angerday'), ('lawyer', 'awyerlay'), ('matter',
'attermay'), ('rugged', 'uggedray'), ('highly', 'ighlyhay'), ('whence',
'encewhay'), ('martha', 'arthamay'), ('hearty', 'eartyhay'), ('recall',
'ecallray'), ('raised', 'aisedray'), ('follow', 'ollowfay'), ('nearer',
'earernay'), ('gentle', 'entlegay'), ('pitied', 'itiedpay'), ('nurses',
'ursesnay'), ('punish', 'unishpay')], (7, 9): [('parcels', 'arcelspay'),
('welcome', 'elcomeway'), ('posture', 'osturepay'), ('mothers', 'othersmay'),
('tempers', 'emperstay'), ('carries', 'arriescay'), ('neglect', 'eglectnay'),
('prevail', 'evailpray'), ('declare', 'eclareday'), ('crowned', 'ownedcray'),
('mindful', 'indfulmay'), ('blessed', 'essedblay'), ('kitchen', 'itchenkay'),

```
('younger', 'youngeray'), ('hanover', 'anoverhay'), ('lowered', 'oweredlay'),
('summons', 'ummonssay'), ('suffice', 'ufficesay'), ('rousing', 'ousingray'),
('feeling', 'eelingfay'), ('parlors', 'arlorspay'), ('gravely', 'avelygray'),
('connect', 'onnectcay'), ('species', 'eciesspay'), ('revived', 'evivedray'),
('tribute', 'ibutetray'), ('pitying', 'ityingpay'), ('brother', 'otherbray'),
('retired', 'etiredray'), ('suffers', 'ufferssay'), ('spurned', 'urnedspay'),
('dropped', 'oppeddray'), ('galling', 'allinggay'), ('seducer', 'educersay'),
('foibles', 'oiblesfay'), ('tuesday', 'uesdaytay'), ('patient', 'atientpay'),
('worsted', 'orstedway'), ('trifled', 'ifledtray'), ('waiting', 'aitingway'),
('widower', 'idowerway'), ('hardily', 'ardilyhay'), ('glimpse', 'impseglay'),
('whether', 'etherwhay'), ('reverie', 'everieray'), ('resolve', 'esolveray'),
('propose', 'oposepray'), ('tallest', 'allesttay'), ('nominal', 'ominalnay'),
('languid', 'anguidlay'), ('wronged', 'ongedwray'), ('nothing', 'othingnay'),
('promote', 'omotepray'), ('novelty', 'oveltynay'), ('dooming', 'oomingday'),
('display', 'isplayday'), ('correct', 'orrectcay'), ('feigned', 'eignedfay'),
('schemes', 'emesschay'), ('deepest', 'eepestday'), ('flannel', 'annelflay'),
('hauteur', 'auteurhay'), ('sharing', 'aringshay'), ('getting', 'ettinggay'),
('screams', 'eamsscray'), ('variety', 'arietyvay'), ('yielded', 'yieldeday'),
('brought', 'oughtbray'), ('haughty', 'aughtyhay'), ('bristol', 'istolbray'),
('silence', 'ilencesay'), ('staying', 'ayingstay'), ('confess', 'onfesscay'),
('placing', 'acingplay'), ('noisier', 'oisiernay'), ('nursing', 'ursingnay'),
('tuition', 'uitiontay'), ('welfare', 'elfareway'), ('hunters', 'untershay'),
('poplars', 'oplarspay'), ('deprive', 'epriveday'), ('mansion', 'ansionmay'),
('reasons', 'easonsray'), ('conceit', 'onceitcay'), ('burgess', 'urgessbay'),
('receipt', 'eceiptray'), ('befalls', 'efallsbay'), ('fetches', 'etchesfay'),
('ranging', 'angingray'), ('passion', 'assionpay'), ('smirked', 'irkedsmay'),
('honours', 'onourshay'), ('covered', 'overedcay'), ('dancing', 'ancingday'),
('hundred', 'undredhay'), ('forgave', 'orgavefay'), ('fifteen', 'ifteenfay'),
('lasting', 'astinglay'), ('minuter', 'inutermay'), ('tremour', 'emourtray'),
('passing', 'assingpay'), ('streets', 'eetsstray'), ('resists', 'esistsray'),
('cleared', 'earedclay'), ('miracle', 'iraclemay'), ('several', 'everalsay'),
('sighing', 'ighingsay'), ('resumed', 'esumedray'), ('merited', 'eritedmay'),
('without', 'ithoutway'), ('removed', 'emovedray'), ('behaved', 'ehavedbay'),
('succour', 'uccoursay'), ('bribing', 'ibingbray'), ('charity', 'aritychay'),
('shortly', 'ortlyshay'), ('tempted', 'emptedtay'), ('briefly', 'ieflybray'),
('parties', 'artiespay'), ('touched', 'ouchedtay'), ('content', 'ontentcay'),
('proceed', 'oceedpray'), ('windows', 'indowsway'), ('ribbons', 'ibbonsray'),
('stepped', 'eppedstay'), ('pursuit', 'ursuitpay'), ('quarter', 'arterquay'),
('prudent', 'udentpray'), ('dignity', 'ignityday'), ('payment', 'aymentpay'),
('grounds', 'oundsgray'), ('bequest', 'equestbay'), ('donavan', 'onavanday'),
('renewed', 'enewedray'), ('smaller', 'allersmay'), ('fearful', 'earfulfay'),
('civilly', 'ivillycay'), ('romance', 'omanceray'), ('females', 'emalesfay'),
('grieved', 'ievedgray'), ('created', 'eatedcray'), ('command', 'ommandcay'),
('defined', 'efinedday'), ('bluntly', 'untlyblay'), ('recital', 'ecitalray'),
('sorrows', 'orrowssay'), ('whither', 'itherwhay'), ('spotted', 'ottedspay'),
('justify', 'ustifyjay'), ('conduit', 'onduitcay'), ('revenge', 'evengeray'),
('scenery', 'eneryscay'), ('started', 'artedstay'), ('degrees', 'egreesday'),
('whisper', 'isperwhay'), ('strange', 'angestray'), ('bidding', 'iddingbay'),
```

```
('musical', 'usicalmay'), ('happens', 'appenshay'), ('compare', 'omparecay'),
('presume', 'esumepray'), ('sitting', 'ittingsay'), ('regards', 'egardsray'),
('violins', 'iolinsvay'), ('rightly', 'ightlyray'), ('memento', 'ementomay'),
('courted', 'ourtedcay'), ('servant', 'ervantsay'), ('possess', 'ossesspay'),
('herself', 'erselfhay'), ('present', 'esentpray'), ('natural', 'aturalnay'),
('managed', 'anagedmay'), ('hostess', 'ostesshay'), ('leaving', 'eavinglay'),
('mention', 'entionmay'), ('circuit', 'ircuitcay'), ('counsel', 'ounselcay'),
('guineas', 'uineasgay'), ('solidly', 'olidlysay'), ('dashing', 'ashingday'),
('station', 'ationstay'), ('written', 'ittenwray'), ('writing', 'itingwray'),
('country', 'ountrycay'), ('delight', 'elightday'), ('westons', 'estonsway'),
('debated', 'ebatedday'), ('lengths', 'engthslay'), ('torment', 'ormenttay'),
('reverse', 'everseray'), ('bedroom', 'edroombay'), ('forbade', 'orbadefay'),
('comings', 'omingscay'), ('tonight', 'onighttay'), ('similar', 'imilarsay'),
('calling', 'allingcay'), ('cassino', 'assinocay'), ('strains', 'ainsstray'),
('scratch', 'atchscray'), ('gardens', 'ardensgay'), ('freedom', 'eedomfray'),
('compact', 'ompactcay'), ('message', 'essagemay'), ('heavily', 'eavilyhay'),
('manners', 'annersmay'), ('systems', 'ystemssay'), ('forbore', 'orborefay'),
('wrapped', 'appedwray'), ('defects', 'efectsday'), ('villain', 'illainvay'),
('studied', 'udiedstay'), ('playing', 'ayingplay'), ('diction', 'ictionday'),
('tending', 'endingtay'), ('seeking', 'eekingsay'), ('rapture', 'aptureray'),
('dawdled', 'awdledday'), ('wearing', 'earingway'), ('comfort', 'omfortcay'),
('finally', 'inallyfay'), ('circles', 'irclescay'), ('putting', 'uttingpay'),
('twisted', 'istedtway'), ('destroy', 'estroyday'), ('whereas', 'ereaswhay'),
('dismiss', 'ismissday'), ('visibly', 'isiblyvay'), ('talents', 'alentstay'),
('finding', 'indingfay'), ('removes', 'emovesray'), ('replied', 'epliedray'),
('bluster', 'usterblay'), ('raising', 'aisingray'), ('plainly', 'ainlyplay'),
('worried', 'orriedway'), ('meaning', 'eaningmay'), ('blushes', 'ushesblay'),
('relapse', 'elapseray'), ('careful', 'arefulcay'), ('beloved', 'elovedbay'),
('degrade', 'egradeday'), ('foresaw', 'oresawfay'), ('procure', 'ocurepray'),
('kindest', 'indestkay'), ('secures', 'ecuressay'), ('hanging', 'anginghay'),
('persons', 'ersonspay'), ('taverns', 'avernstay'), ('walking', 'alkingway'),
('quicker', 'ickerquay'), ('guessed', 'uessedgay'), ('confirm', 'onfirmcay'),
('venture', 'enturevay'), ('project', 'ojectpray'), ('prayers', 'ayerspray'),
('heights', 'eightshay'), ('subject', 'ubjectsay'), ('summits', 'ummitssay'),
('towards', 'owardstay'), ('closing', 'osingclay'), ('journey', 'ourneyjay'),
('pointer', 'ointerpay'), ('chained', 'ainedchay'), ('clothes', 'othesclay'),
('require', 'equireray'), ('chusing', 'usingchay'), ('reproof', 'eproofray'),
('reckons', 'eckonsray'), ('pressed', 'essedpray'), ('talking', 'alkingtay'),
('harbour', 'arbourhay'), ('garrets', 'arretsgay'), ('foresee', 'oreseefay'),
('lustily', 'ustilylay'), ('leagued', 'eaguedlay'), ('judging', 'udgingjay'),
('rejoice', 'ejoiceray'), ('honiton', 'onitonhay'), ('shorten', 'ortenshay'),
('stating', 'atingstay'), ('privacy', 'ivacypray'), ('steeles', 'eelesstay'),
('thickly', 'icklythay'), ('shocked', 'ockedshay'), ('limited', 'imitedlay'),
('blights', 'ightsblay'), ('seclude', 'ecludesay'), ('cordial', 'ordialcay'),
('request', 'equestray'), ('denoted', 'enotedday'), ('darling', 'arlingday'),
('coxcomb', 'oxcombcay'), ('however', 'oweverhay'), ('perform', 'erformpay'),
('society', 'ocietysay'), ('deserts', 'esertsday'), ('palmers', 'almerspay'),
('manager', 'anagermay'), ('rupture', 'uptureray'), ('concise', 'oncisecay'),
```

('scolded', 'oldedscay'), ('seduced', 'educedsay'), ('doubted', 'oubtedday'),
('forward', 'orwardfay'), ('leisure', 'eisurelay'), ('robbing', 'obbingray'),
('horizon', 'orizonhay'), ('blunder', 'underblay'), ('flushed', 'ushedflay'),
('divided', 'ividedday'), ('charged', 'argedchay'), ('disturb', 'isturbday'),
('smitten', 'ittensmay'), ('seconds', 'econdssay'), ('letters', 'etterslay'),
('provide', 'ovidepray'), ('conduct', 'onductcay'), ('plagued', 'aguedplay'),
('stretch', 'etchstray'), ('hoarded', 'oardedhay'), ('blushed', 'ushedblay'),
('nodding', 'oddingnay'), ('mankind', 'ankindmay'), ('caprice', 'apricecay'),
('quitted', 'ittedquay')], (5, 7): [('china', 'inachay'), ('deter', 'eterday'),
('woman', 'omanway'), ('clock', 'ockclay'), ('borne', 'ornebay'), ('noise',
'oisenay'), ('midst', 'idstmay'), ('dwell', 'elldway'), ('smith', 'ithsmay'),
('gowns', 'ownsgay'), ('named', 'amednay'), ('glass', 'assglay'), ('price',
'icepray'), ('close', 'oseclay'), ('guide', 'uidegay'), ('newly', 'ewlynay'),
('sweep', 'eepsway'), ('swept', 'eptsway'), ('those', 'osethay'), ('lower',
'owerlay'), ('plump', 'umpplay'), ('downs', 'ownsday'), ('purse', 'ursepay'),
('linen', 'inenlay'), ('later', 'aterlay'), ('broad', 'oadbray'), ('miles',
'ilesmay'), ('facts', 'actsfay'), ('sleep', 'eepslay'), ('steps', 'epsstay'),
('wants', 'antsway'), ('forth', 'orthfay'), ('drove', 'ovedray'), ('mouth',
'outhmay'), ('worry', 'orryway'), ('dried', 'ieddray'), ('stood', 'oodstay'),
('brush', 'ushbray'), ('sting', 'ingstay'), ('mirth', 'irthmay'), ('heads',
'eadshay'), ('sight', 'ightsay'), ('whose', 'osewhay'), ('drawn', 'awndray'),
('score', 'orescay'), ('party', 'artypay'), ('tears', 'earstay'), ('heart',
'earthay'), ('words', 'ordsway'), ('hunts', 'untshay'), ('trace', 'acetray'),
('trial', 'ialtray'), ('swoon', 'oonsway'), ('knees', 'eesknay'), ('bring',
'ingbray'), ('solid', 'olidsay'), ('sharp', 'arpshay'), ('basis', 'asisbay'),
('cavil', 'avilcay'), ('rated', 'atedray'), ('grave', 'avegray'), ('serve',
'ervesay'), ('betty', 'ettybay'), ('judge', 'udgejay'), ('boast', 'oastbay'),
('march', 'archmay'), ('built', 'uiltbay'), ('clear', 'earclay'), ('happy',
'appyhay'), ('fruit', 'uitfray'), ('witty', 'ittyway'), ('total', 'otaltay'),
('dared', 'aredday'), ('defer', 'eferday'), ('tried', 'iedtray'), ('vexed',
'exedvay'), ('walls', 'allsway'), ('liked', 'ikedlay'), ('prove', 'ovepray'),
('sneer', 'eersnay'), ('merit', 'eritmay'), ('scott', 'ottscay'), ('madam',
'adammay'), ('truth', 'uthtray'), ('minor', 'inormay'), ('stare', 'arestay'),
('calls', 'allscay'), ('lucky', 'uckylay'), ('hence', 'encehay'), ('press',
'esspray'), ('comes', 'omescay'), ('dined', 'inedday'), ('think', 'inkthay'),
('plate', 'ateplay'), ('grief', 'iefgray'), ('names', 'amesnay'), ('stock',
'ockstay'), ('short', 'ortshay'), ('moved', 'ovedmay'), ('teach', 'eachtay'),
('fears', 'earsfay'), ('tiled', 'iledtay'), ('cried', 'iedcray'), ('mound',
'oundmay'), ('small', 'allsmay'), ('twice', 'icetway'), ('lines', 'ineslay'),
('looks', 'ookslay'), ('rival', 'ivalray'), ('first', 'irstfay'), ('wound',
'oundway'), ('saves', 'avessay'), ('nurse', 'ursenay'), ('treat', 'eattray'),
('could', 'ouldcay'), ('rider', 'iderray'), ('lobby', 'obbylay'), ('table',
'abletay'), ('touch', 'ouchtay'), ('topic', 'opictay'), ('stand', 'andstay'),
('leant', 'eantlay'), ('grass', 'assgray'), ('check', 'eckchay'), ('quite',
'itequay'), ('wives', 'ivesway'), ('walks', 'alksway'), ('labor', 'aborlay'),
('house', 'ousehay'), ('drops', 'opsdray'), ('loose', 'ooselay'), ('tempt',
'empttay'), ('place', 'aceplay'), ('local', 'ocallay'), ('seats', 'eatssay'),
('gates', 'atesgay'), ('dropt', 'optdray'), ('cares', 'arescay'), ('falls',

'allsfay'), ('speak', 'eakspay'), ('shops', 'opsshay'), ('groom', 'oomgray'), ('sugar', 'ugarsay'), ('front', 'ontfray'), ('visit', 'isitvay'), ('raise', 'aiseray'), ('fetch', 'etchfay'), ('silks', 'ilkssay'), ('beset', 'esetbay'), ('rocks', 'ocksray'), ('start', 'artstay'), ('winks', 'inksway'), ('sends', 'endssay'), ('piano', 'ianopay'), ('fully', 'ullyfay'), ('widow', 'idowway'), ('wiser', 'iserway'), ('dirty', 'irtyday'), ('below', 'elowbay'), ('manor', 'anormay'), ('scale', 'alescay'), ('whole', 'olewhay'), ('minds', 'indsmay'), ('glebe', 'ebeglay'), ('fifty', 'iftyfay'), ('years', 'yearsay'), ('story', 'orystay'), ('fresh', 'eshfray'), ('begun', 'egunbay'), ('dance', 'anceday'), ('jokes', 'okesjay'), ('turns', 'urnstay'), ('honor', 'onorhay'), ('spite', 'itespay'), ('green', 'eengray'), ('saint', 'aintsay'), ('hated', 'atedhay'), ('doors', 'oorsday'), ('shown', 'ownshay'), ('power', 'owerpay'), ('today', 'odaytay'), ('floor', 'oorflay'), ('force', 'orcefay'), ('right', 'ightray'), ('trick', 'icktray'), ('dread', 'eaddray'), ('final', 'inalfay'), ('shake', 'akeshay'), ('lamps', 'ampslay'), ('dying', 'yingday'), ('black', 'ackblay'), ('grand', 'andgray'), ('field', 'ieldfay'), ('plead', 'eadplay'), ('loved', 'ovedlay'), ('terms', 'ermstay'), ('spend', 'endspay'), ('mohrs', 'ohrsmay'), ('young', 'youngay'), ('great', 'eatgray'), ('guess', 'uessgay'), ('ninny', 'innynay'), ('hasty', 'astyhay'), ('nerve', 'ervenay'), ('meals', 'ealsmay'), ('three', 'eethray'), ('shade', 'adeshay'), ('store', 'orestay'), ('daily', 'ailyday'), ('taste', 'astetay'), ('gross', 'ossgray'), ('dares', 'aresday'), ('grate', 'ategray'), ('stuff', 'uffstay'), ('sheet', 'eetshay')], (14, 18): [('breakfast-room', 'eakfastbray-oomray'), ('lavender-water', 'avenderlay-aterway'), ('broken-hearted', 'okenbray-eartedhay'), ('spunging-house', 'ungingspay-ousehay')], (9, 12): [('unwelcome', 'unwelcomeway'), ('exhausted', 'exhaustedway'), ('intrusion', 'intrusionway'), ('untouched', 'untouchedway'), ('interview', 'interviewway'), ('effusions', 'effusionsway'), ('impartial', 'impartialway'), ('inventive', 'inventiveway'), ('emergency', 'emergencyway'), ('officious', 'officiousway'), ('everybody', 'everybodyway'), ('observant', 'observantway'), ('emptiness', 'emptinessway'), ('occasions', 'occasionsway'), ('assembled', 'assembledway'), ('admirable', 'admirableway'), ('uselessly', 'uselesslyway'), ('excellent', 'excellentway'), ('associate', 'associateway'), ('affecting', 'affectingway'), ('absurdity', 'absurdityway'), ('allowable', 'allowableway'), ('appointed', 'appointedway'), ('overheard', 'overheardway'), ('improving', 'improvingway'), ('eagerness', 'eagernessway'), ('unluckily', 'unluckilyway'), ('allowance', 'allowanceway'), ('estimable', 'estimableway'), ('obviating', 'obviatingway'), ('entrusted', 'entrustedway'), ('arranging', 'arrangingway'), ('exclaimed', 'exclaimedway'), ('ingenuity', 'ingenuityway'), ('outwardly', 'outwardlyway'), ('expensive', 'expensiveway'), ('entreated', 'entreatedway'), ('unconcern', 'unconcernway'), ('obtaining', 'obtainingway'), ('interests', 'interestsway'), ('assiduous', 'assiduousway'), ('adjusting', 'adjustingway'), ('ungenteel', 'ungenteelway'), ('encourage', 'encourageway'), ('apartment', 'apartmentway'), ('inheritor', 'inheritorway'), ('unceasing', 'unceasingway'), ('insulting', 'insultingway'), ('unchanged', 'unchangedway'), ('animation', 'animationway'), ('enclosure', 'enclosureway'), ('apologize', 'apologizeway'), ('earnestly', 'earnestlyway'), ('exquisite', 'exquisiteway'), ('attending', 'attendingway'), ('invention', 'inventionway'), ('enquiries', 'enquiriesway'), ('appetites', 'appetitesway'), ('additions', 'additionsway'),

('impudence', 'impudenceway'), ('insincere', 'insincereway'), ('abundance', 'abundanceway'), ('exchanged', 'exchangedway'), ('extremely', 'extremelyway'), ('inflicted', 'inflictedway'), ('attracted', 'attractedway'), ('inelegant', 'inelegantway'), ('undoubted', 'undoubtedway'), ('etiquette', 'etiquetteway'), ('unfeigned', 'unfeignedway'), ('agreeable', 'agreeableway'), ('unsubdued', 'unsubduedway'), ('inspiring', 'inspiringway'), ('unguarded', 'unguardedway'), ('advancing', 'advancingway'), ('undressed', 'undressedway'), ('excursive', 'excursiveway'), ('extending', 'extendingway'), ('extolling', 'extollingway'), ('offending', 'offendingway'), ('eloquence', 'eloquenceway'), ('ornaments', 'ornamentsway'), ('intruding', 'intrudingway'), ('incessant', 'incessantway'), ('unwilling', 'unwillingway'), ('annamaria', 'annamariaway'), ('unbounded', 'unboundedway'), ('incurable', 'incurableway'), ('unnoticed', 'unnoticedway'), ('expressly', 'expresslyway'), ('impetuous', 'impetuousway'), ('appearing', 'appearingway'), ('ingenious', 'ingeniousway'), ('amazement', 'amazementway'), ('animating', 'animatingway'), ('interfere', 'interfereway'), ('ascending', 'ascendingway'), ('affection', 'affectionway'), ('undergone', 'undergoneway'), ('unstudied', 'unstudiedway'), ('attempted', 'attemptedway'), ('infirmity', 'infirmityway'), ('apologies', 'apologiesway'), ('elsewhere', 'elsewhereway'), ('expecting', 'expectingway'), ('informing', 'informingway'), ('admission', 'admissionway'), ('extremity', 'extremityway')], (6, 9): [('unread', 'unreadway'), ('owning', 'owningway'), ('odious', 'odiousway'), ('either', 'eitherway'), ('insult', 'insultway'), ('awhile', 'awhileway'), ('assist', 'assistway'), ('injure', 'injureway'), ('eluded', 'eludedway'), ('others', 'othersway'), ('untied', 'untiedway'), ('excess', 'excessway'), ('urgent', 'urgentway'), ('appear', 'appearway'), ('aspect', 'aspectway'), ('issued', 'issuedway'), ('evince', 'evinceway'), ('eating', 'eatingway'), ('easter', 'easterway'), ('arisen', 'arisenway'), ('earned', 'earnedway'), ('esteem', 'esteemway'), ('accuse', 'accuseway'), ('owners', 'ownersway'), ('events', 'eventsway'), ('energy', 'energyway'), ('affair', 'affairway'), ('effect', 'effectway'), ('unlike', 'unlikeway'), ('unison', 'unisonway'), ('enough', 'enoughway'), ('invite', 'inviteway'), ('income', 'incomeway'), ('extent', 'extentway'), ('office', 'officeway'), ('ankles', 'anklesway'), ('admire', 'admireway'), ('induce', 'induceway'), ('infant', 'infantway'), ('expand', 'expandway'), ('orders', 'ordersway'), ('occupy', 'occupyway'), ('attend', 'attendway'), ('unable', 'unableway'), ('olives', 'olivesway'), ('awaken', 'awakenway'), ('engage', 'engageway'), ('eldest', 'eldestway'), ('eyeing', 'eyeingway'), ('advice', 'adviceway'), ('always', 'alwaysway'), ('insist', 'insistway'), ('accept', 'acceptway'), ('indies', 'indiesway'), ('excite', 'exciteway'), ('openly', 'openlyway'), ('oblige', 'obligeway'), ('amidst', 'amidstway'), ('across', 'acrossway'), ('afraid', 'afraidway'), ('uproar', 'uproarway')], (8, 10): [('villages', 'illagesvay'), ('doubtful', 'oubtfulday'), ('deceived', 'eceivedday'), ('foreseen', 'oreseenfay'), ('dennison', 'ennisonday'), ('steadily', 'eadilystay'), ('startled', 'artledstay'), ('drawings', 'awingsdray'), ('rejected', 'ejectedray'), ('deserves', 'eservesday'), ('curricle', 'urriclecay'), ('domestic', 'omesticday'), ('believed', 'elievedbay'), ('packages', 'ackagespay'), ('kneeling', 'eelingknay'), ('degraded', 'egradedday'), ('medicine', 'edicinemay'), ('promoted', 'omotedpray'), ('wondered', 'onderedway'), ('reproach',

'eproachray'), ('somerset', 'omersetsay'), ('contents', 'ontentscay'),
('children', 'ildrenchay'), ('politics', 'oliticspay'), ('hastened',
'astenedhay'), ('relative', 'elativeray'), ('henshawe', 'enshawehay'),
('material', 'aterialmay'), ('welcomed', 'elcomedway'), ('weddings',
'eddingsway'), ('diverted', 'ivertedday'), ('patience', 'atiencepay'),
('delivery', 'eliveryday'), ('hazarded', 'azardedhay'), ('happiest',
'appiesthay'), ('trifling', 'iflingtray'), ('casement', 'asementcay'),
('reformed', 'eformedray'), ('demanded', 'emandedday'), ('ceremony',
'eremonycay'), ('vigorous', 'igorousvay'), ('mornings', 'orningsmay'),
('hilarity', 'ilarityhay'), ('daughter', 'aughterday'), ('vexation',
'exationvay'), ('shocking', 'ockingshay'), ('previous', 'eviouspray'),
('pleasant', 'easantplay'), ('candidly', 'andidlycay'), ('cheerful',
'eerfulchay'), ('festival', 'estivalfay'), ('changing', 'angingchay'),
('conveyed', 'onveyedcay'), ('lombardy', 'ombardylay'), ('militate',
'ilitatemay'), ('humanity', 'umanityhay'), ('giggling', 'igglinggay'),
('standard', 'andardstay'), ('returned', 'eturnedray'), ('proclaim',
'oclaimpray'), ('ridicule', 'idiculeray'), ('pointers', 'ointerspay'),
('grounded', 'oundedgray'), ('contrast', 'ontrastcay'), ('feelings',
'eelingsfay'), ('mistakes', 'istakesmay'), ('prepared', 'eparedpray'),
('stranger', 'angerstray'), ('habitual', 'abitualhay'), ('firmness',
'irmnessfay'), ('rational', 'ationalray'), ('regulate', 'egulateray'),
('claiming', 'aimingclay'), ('knocking', 'ockingknay'), ('servants',
'ervantssay'), ('goodness', 'oodnessgay'), ('mutually', 'utuallymay'),
('legacies', 'egacieslay'), ('sisterly', 'isterlysay'), ('punctual',
'unctualpay'), ('deciding', 'ecidingday'), ('suitable', 'uitablesay'),
('guarding', 'uardinggay'), ('converse', 'onversecay'), ('slighter',
'ighterslay'), ('dissuade', 'issuadeday'), ('despatch', 'espatchday'),
('blackest', 'ackestblay'), ('recalled', 'ecalledray'), ('loitered',
'oiteredlay'), ('dreading', 'eadingdray'), ('relieved', 'elievedray'),
('westward', 'estwardway'), ('thistles', 'istlesthay'), ('defended',
'efendedday'), ('contempt', 'ontemptcay'), ('sensibly', 'ensiblysay'),
('romantic', 'omanticray'), ('comforts', 'omfortscay'), ('jealousy',
'ealousyjay'), ('repeated', 'epeatedray'), ('banditti', 'andittibay'),
('greatest', 'eatestgray'), ('mistrust', 'istrustmay'), ('forwards',
'orwardsfay'), ('rewarded', 'ewardedray'), ('watching', 'atchingway'),
('negative', 'egativenay'), ('sympathy', 'ympathysay'), ('helpless',
'elplesshay'), ('quickest', 'ickestquay'), ('portrait', 'ortraitpay'),
('consoled', 'onsoledcay'), ('fourteen', 'ourteenfay'), ('retarded',
'etardedray'), ('comments', 'ommentscay'), ('jennings', 'enningsjay'),
('securing', 'ecuringsay'), ('withdrew', 'ithdrewway'), ('youthful',
'youthfulay'), ('bachelor', 'achelorbay'), ('convince', 'onvincecay'),
('measured', 'easuredmay'), ('goodwill', 'oodwillgay'), ('branched',
'anchedbray'), ('purchase', 'urchasepay'), ('covenant', 'ovenantcay'),
('reseated', 'eseatedray'), ('concerns', 'oncernscay'), ('hardship',
'ardshiphay'), ('lessened', 'essenedlay'), ('delaford', 'elafordday'),
('governed', 'overnedgay'), ('listened', 'istenedlay'), ('constant',
'onstantcay'), ('fatigued', 'atiguedfay'), ('persuade', 'ersuadepay'),
('critique', 'itiquecray'), ('remedies', 'emediesray'), ('stepping',

'eppingstay'), ('kindness', 'indnesskay'), ('sociable', 'ociablesay'),
('meetings', 'eetingsmay'), ('deserved', 'eservedday'), ('thoughts',
'oughtsthay'), ('becoming', 'ecomingbay'), ('heartily', 'eartilyhay'),
('promised', 'omisedpray'), ('shutting', 'uttingshay'), ('reproved',
'eprovedray'), ('ladyship', 'adyshiplay'), ('pursuing', 'ursuingpay'),
('residing', 'esidingray'), ('sentence', 'entencesay'), ('revealed',
'evealedray'), ('decrease', 'ecreaseday'), ('speedily', 'eedilyspay'),
('commands', 'ommandscay'), ('mistaken', 'istakenmay'), ('settling',
'ettlingsay'), ('polished', 'olishedpay'), ('stanhill', 'anhillstay'),
('retailed', 'etailedray'), ('pointing', 'ointingpay'), ('strongly',
'onglystray'), ('subjects', 'ubjectssay'), ('calmness', 'almnesscay'),
('doctrine', 'octrineday'), ('handsome', 'andsomehay'), ('profited',
'ofitedpray'), ('planning', 'anningplay'), ('denoting', 'enotingday'),
('commonly', 'ommonlycay'), ('bestowed', 'estowedbay'), ('marriage',
'arriagemay'), ('presents', 'esentspray'), ('complete', 'ompletecay'),
('striking', 'ikingstray'), ('needless', 'eedlessnay'), ('plighted',
'ightedplay'), ('tranquil', 'anquiltray'), ('spacious', 'aciousspay'),
('nonsense', 'onsensenay'), ('likeness', 'ikenesslay'), ('valuable',
'aluablevay'), ('laudable', 'audablelay'), ('delirium', 'eliriumday'),
('believes', 'elievesbay'), ('premises', 'emisespray'), ('presumed',
'esumedpray'), ('coldness', 'oldnesscay'), ('metaphor', 'etaphormay'),
('thankful', 'ankfulthay'), ('suffered', 'ufferedsay'), ('preserve',
'eservepray'), ('resorted', 'esortedray'), ('felicity', 'elicityfay'),
('betrayed', 'etrayedbay'), ('throwing', 'owingthray'), ('powerful',
'owerfulpay'), ('humoured', 'umouredhay'), ('practise', 'actisepray'),
('coloured', 'olouredcay'), ('regiment', 'egimentray'), ('stubborn',
'ubbornstay'), ('huswifes', 'uswifeshay'), ('painting', 'aintingpay'),
('friendly', 'iendlyfray'), ('happened', 'appenedhay'), ('moderate',
'oderatemay'), ('solitary', 'olitarysay'), ('wildness', 'ildnessway'),
('marianne', 'ariannemay'), ('delicacy', 'elicacyday'), ('progress',
'ogresspray'), ('coachman', 'oachmancay'), ('grieving', 'ievinggray'),
('troubles', 'oublestray'), ('coolness', 'oolnesscay'), ('possibly',
'ossiblypay'), ('relating', 'elatingray'), ('patterns', 'atternspay'),
('consists', 'onsistscay'), ('creating', 'eatingcray'), ('confused',
'onfusedcay'), ('critical', 'iticalcray'), ('laboured', 'abouredlay'),
('minutely', 'inutelymay'), ('tendency', 'endencytay'), ('distrust',
'istrustday'), ('distress', 'istressday'), ('conclude', 'oncludecay'),
('sixpence', 'ixpencesay'), ('somewhat', 'omewhatsay'), ('coincide',
'oincidecay'), ('harmless', 'armlesshay'), ('juvenile', 'uvenilejay'),
('pleasing', 'easingplay'), ('precious', 'eciouspray'), ('restrain',
'estrainray'), ('produced', 'oducedpray'), ('renewing', 'enewingray'),
('pinching', 'inchingpay'), ('severely', 'everelysay'), ('blossoms',
'ossomsblay'), ('heighten', 'eightenhay'), ('wandered', 'anderedway'),
('properly', 'operlypray'), ('veracity', 'eracityvay'), ('released',
'eleasedray'), ('transact', 'ansacttray'), ('helpmate', 'elpmatehay'),
('business', 'usinessbay'), ('concerto', 'oncertocay'), ('reserved',
'eservedray'), ('reliance', 'elianceray'), ('headache', 'eadachehay'),
('families', 'amiliesfay'), ('centered', 'enteredcay'), ('mulberry',

```
'ulberrymay'), ('sensible', 'ensiblesay'), ('steepest', 'eepeststay'),
('compared', 'omparedcay'), ('speaking', 'eakingspay'), ('censured',
'ensuredcay'), ('screened', 'eenedscray'), ('pictures', 'icturespay'),
('reaching', 'eachingray'), ('wonderer', 'ondererway'), ('whenever',
'eneverwhay'), ('detested', 'etestedday'), ('dispatch', 'ispatchday'),
('shutters', 'uttersshay'), ('disgrace', 'isgraceday'), ('dejected',
'ejectedday'), ('conquest', 'onquestcay'), ('foretell', 'oretellfay'),
('weakened', 'eakenedway'), ('brooding', 'oodingbray'), ('shedding',
'eddingshay'), ('poignant', 'oignantpay'), ('trembled', 'embledtray'),
('resolute', 'esoluteray'), ('reverted', 'evertedray'), ('silenced',
'ilencedsay'), ('jointure', 'ointurejay')], (10, 13): [('illaudable',
'illaudableway'), ('incivility', 'incivilityway'), ('oppression',
'oppressionway'), ('acquainted', 'acquaintedway'), ('economical',
'economicalway'), ('ebullition', 'ebullitionway'), ('affronting',
'affrontingway'), ('unforeseen', 'unforeseenway'), ('auspicious',
'auspiciousway'), ('indecorous', 'indecorousway'), ('attractive',
'attractiveway'), ('exercising', 'exercisingway'), ('unsatiable',
'unsatiableway'), ('individual', 'individualway'), ('enthusiasm',
'enthusiasmway'), ('announcing', 'announcingway'), ('indistinct',
'indistinctway'), ('affliction', 'afflictionway'), ('imaginable',
'imaginableway'), ('obstructed', 'obstructedway'), ('improperly',
'improperlyway'), ('illiterate', 'illiterateway'), ('interested',
'interestedway'), ('infamously', 'infamouslyway'), ('amusements',
'amusementsway'), ('endeavours', 'endeavoursway'), ('inhabiting',
'inhabitingway'), ('indisposed', 'indisposedway'), ('acquiesced',
'acquiescedway'), ('anticipate', 'anticipateway'), ('accurately',
'accuratelyway'), ('influenced', 'influencedway'), ('assemblies',
'assembliesway'), ('approached', 'approachedway'), ('accusation',
'accusationway'), ('apparently', 'apparentlyway'), ('intimately',
'intimatelyway'), ('uneasiness', 'uneasinessway'), ('alteration',
'alterationway'), ('artificial', 'artificialway'), ('admiration',
'admirationway'), ('accidental', 'accidentalway'), ('unexpected',
'unexpectedway'), ('insensible', 'insensibleway'), ('expressing',
'expressingway'), ('accelerate', 'accelerateway'), ('everything',
'everythingway'), ('increasing', 'increasingway'), ('undervalue',
'undervalueway'), ('invaluable', 'invaluableway'), ('affections',
'affectionsway'), ('infinitely', 'infinitelyway'), ('unsuitable',
'unsuitableway'), ('authorised', 'authorisedway'), ('impudently',
'impudentlyway'), ('unpardoned', 'unpardonedway'), ('inevitable',
'inevitableway'), ('ungenerous', 'ungenerousway'), ('impression',
'impressionway'), ('indelicacy', 'indelicacyway'), ('inducement',
'inducementway'), ('unaffected', 'unaffectedway'), ('instigated',
'instigatedway'), ('inquietude', 'inquietudeway'), ('attempting',
'attemptingway'), ('experience', 'experienceway'), ('impatience',
'impatienceway'), ('indulgence', 'indulgenceway'), ('affability',
'affabilityway'), ('unavailing', 'unavailingway'), ('incommoded',
'incommodedway'), ('altogether', 'altogetherway')], (4, 6): [('done', 'oneday'),
('wife', 'ifeway'), ('plea', 'eaplay'), ('mall', 'allmay'), ('sang', 'angsay'),
```

```
('clue', 'ueclay'), ('bell', 'ellbay'), ('bind', 'indbay'), ('join', 'oinjay'),
('went', 'entway'), ('vile', 'ilevay'), ('hens', 'enshay'), ('tore', 'oretay'),
('rage', 'ageray'), ('flow', 'owflay'), ('gate', 'ategay'), ('palm', 'almpay'),
('give', 'ivegay'), ('lips', 'ipslay'), ('sink', 'inksay'), ('rent', 'entray'),
('gave', 'avegay'), ('joke', 'okejay'), ('miss', 'issmay'), ('mere', 'eremay'),
('made', 'ademay'), ('dogs', 'ogsday'), ('bent', 'entbay'), ('duly', 'ulyday'),
('when', 'enwhay'), ('rate', 'ateray'), ('sure', 'uresay'), ('land', 'andlay'),
('pink', 'inkpay'), ('hint', 'inthay'), ('cast', 'astcay'), ('park', 'arkpay'),
('bold', 'oldbay'), ('door', 'oorday'), ('bulk', 'ulkbay'), ('doat', 'oatday'),
('coat', 'oatcay'), ('date', 'ateday'), ('foot', 'ootfay'), ('hand', 'andhay'),
('damp', 'ampday'), ('pall', 'allpay'), ('goes', 'oesgay'), ('rung', 'ungray'),
('none', 'onenay'), ('game', 'amegay'), ('nods', 'odsnay'), ('read', 'eadray'),
('gold', 'oldgay'), ('sing', 'ingsay'), ('puts', 'utspay'), ('cats', 'atscay'),
('many', 'anymay'), ('dare', 'areday'), ('noon', 'oonnay'), ('said', 'aidsay'),
('mark', 'arkmay'), ('case', 'asecay'), ('hate', 'atehay'), ('word', 'ordway'),
('beat', 'eatbay'), ('lose', 'oselay'), ('fish', 'ishfay'), ('laid', 'aidlay'),
('fine', 'inefay'), ('snug', 'ugsnay'), ('moss', 'ossmay'), ('heir', 'eirhay'),
('sigh', 'ighsay'), ('mend', 'endmay'), ('sake', 'akesay'), ('pass', 'asspay'),
('debt', 'ebtday'), ('lane', 'anelay'), ('days', 'aysday'), ('deep', 'eepday'),
('song', 'ongsay'), ('spot', 'otspay'), ('took', 'ooktay'), ('both', 'othbay'),
('year', 'yearay'), ('mama', 'amamay'), ('hill', 'illhay'), ('beds', 'edsbay'),
('less', 'esslay'), ('gown', 'owngay'), ('kill', 'illkay'), ('free', 'eefray'),
('hall', 'allhay'), ('room', 'oomray'), ('gray', 'aygray'), ('beef', 'eefbay'),
('kept', 'eptkay'), ('what', 'atwhay'), ('blue', 'ueblay'), ('fell', 'ellfay'),
('draw', 'awdray'), ('play', 'ayplay'), ('baby', 'abybay'), ('rude', 'uderay'),
('trim', 'imtray'), ('chat', 'atchay'), ('bely', 'elybay'), ('fled', 'edflay'),
('past', 'astpay'), ('wait', 'aitway'), ('meal', 'ealmay'), ('wont', 'ontway'),
('week', 'eekway'), ('best', 'estbay'), ('line', 'inelay'), ('lawn', 'awnlay'),
('tree', 'eetray'), ('tend', 'endtay'), ('find', 'indfay'), ('five', 'ivefay'),
('gone', 'onegay'), ('need', 'eednay'), ('drew', 'ewdray'), ('more', 'oremay'),
('cool', 'oolcay'), ('suit', 'uitsay'), ('from', 'omfray'), ('cure', 'urecay'),
('will', 'illway'), ('slow', 'owslay'), ('they', 'eythay'), ('name', 'amenay'),
('shot', 'otshay'), ('come', 'omecay'), ('tied', 'iedtay'), ('came', 'amecay'),
('boys', 'oysbay'), ('wish', 'ishway'), ('hour', 'ourhay'), ('kind', 'indkay'),
('mile', 'ilemay'), ('quit', 'itquay'), ('john', 'ohnjay'), ('hazy', 'azyhay'),
('fall', 'allfay'), ('sell', 'ellsay'), ('tale', 'aletay'), ('grew', 'ewgray'),
('seek', 'eeksay'), ('muff', 'uffmay'), ('soft', 'oftsay'), ('same', 'amesay'),
('body', 'odybay'), ('head', 'eadhay'), ('like', 'ikelay'), ('here', 'erehay'),
('help', 'elphay'), ('sail', 'ailsay'), ('pope', 'opepay'), ('deny', 'enyday'),
('bond', 'ondbay'), ('bath', 'athbay'), ('blow', 'owblay'), ('hold', 'oldhay'),
('safe', 'afesay'), ('path', 'athpay'), ('feed', 'eedfay'), ('thus', 'usthay'),
('four', 'ourfay'), ('lest', 'estlay'), ('thin', 'inthay')], (11, 14):
[('approaching', 'approachingway'), ('unequivocal', 'unequivocalway'),
('exceedingly', 'exceedinglyway'), ('application', 'applicationway'),
('observation', 'observationway'), ('unhappiness', 'unhappinessway'),
('interrupted', 'interruptedway'), ('enlightened', 'enlightenedway'),
('appearances', 'appearancesway'), ('unconnected', 'unconnectedway'),
('unconcerned', 'unconcernedway'), ('indignation', 'indignationway'),
```

```
('accommodate', 'accommodateway'), ('inattention', 'inattentionway'),
('undoubtedly', 'undoubtedlyway'), ('intercourse', 'intercourseway'),
('imperfectly', 'imperfectlyway'), ('imprudently', 'imprudentlyway'),
('affectation', 'affectationway'), ('instigation', 'instigationway'),
('inclination', 'inclinationway'), ('undervalued', 'undervaluedway'),
('affirmative', 'affirmativeway'), ('inconstancy', 'inconstancyway'),
('universally', 'universallyway'), ('engagements', 'engagementsway'),
('exclamation', 'exclamationway'), ('elucidation', 'elucidationway'),
('accustomary', 'accustomaryway'), ('approbation', 'approbationway'),
('unnecessary', 'unnecessaryway'), ('associating', 'associatingway'),
('attributing', 'attributingway'), ('unobtrusive', 'unobtrusiveway'),
('impropriety', 'improprietyway'), ('examination', 'examinationway'),
('alterations', 'alterationsway'), ('unexpressed', 'unexpressedway'),
('abridgement', 'abridgementway'), ('exhilarated', 'exhilaratedway'),
('unsuspected', 'unsuspectedway'), ('independent', 'independentway'),
('alternately', 'alternatelyway'), ('enforcement', 'enforcementway'),
('undertaking', 'undertakingway'), ('outstripped', 'outstrippedway'),
('appointment', 'appointmentway'), ('explanation', 'explanationway'),
('experienced', 'experiencedway'), ('unknowingly', 'unknowinglyway'),
('undesirable', 'undesirableway'), ('ascertained', 'ascertainedway'),
('enumeration', 'enumerationway'), ('alleviation', 'alleviationway'),
('instruction', 'instructionway'), ('undeserving', 'undeservingway'),
('earnestness', 'earnestnessway'), ('insincerity', 'insincerityway'),
('accompanied', 'accompaniedway'), ('employments', 'employmentsway'),
('inferiority', 'inferiorityway'), ('inexcusable', 'inexcusableway'),
('introducing', 'introducingway'), ('extravagant', 'extravagantway'),
('impatiently', 'impatientlyway'), ('attentively', 'attentivelyway'),
('eligibility', 'eligibilityway'), ('uncertainty', 'uncertaintyway')], (7, 10):
[('anxiety', 'anxietyway'), ('attacks', 'attacksway'), ('ailment',
'ailmentway'), ('awkward', 'awkwardway'), ('effects', 'effectsway'), ('induced',
'inducedway'), ('uncivil', 'uncivilway'), ('against', 'againstway'), ('apology',
'apologyway'), ('endless', 'endlessway'), ('alluded', 'alludedway'), ('ability',
'abilityway'), ('address', 'addressway'), ('engaged', 'engagedway'), ('eagerly',
'eagerlyway'), ('expects', 'expectsway'), ('elliott', 'elliottway'), ('oneself',
'oneselfway'), ('anybody', 'anybodyway'), ('augment', 'augmentway'), ('utility',
'utilityway'), ('unquiet', 'unquietway'), ('infancy', 'infancyway'), ('altered',
'alteredway'), ('unequal', 'unequalway'), ('adapted', 'adaptedway'), ('useless',
'uselessway'), ('opposed', 'opposedway'), ('artless', 'artlessway'), ('imagery',
'imageryway'), ('amiably', 'amiablyway'), ('uncouth', 'uncouthway'), ('affixed',
'affixedway'), ('impulse', 'impulseway'), ('opinion', 'opinionway'), ('imitate',
'imitateway'), ('admires', 'admiresway'), ('affront', 'affrontway'), ('amusing',
'amusingway'), ('accents', 'accentsway'), ('implied', 'impliedway'), ('ecstasy',
'ecstasyway'), ('illness', 'illnessway'), ('entreat', 'entreatway'), ('another',
'anotherway'), ('opening', 'openingway'), ('england', 'englandway'), ('ensured',
'ensuredway'), ('imputed', 'imputedway'), ('enabled', 'enabledway'), ('offence',
'offenceway'), ('insured', 'insuredway'), ('urgency', 'urgencyway'), ('amazing',
'amazingway'), ('uniform', 'uniformway'), ('avarice', 'avariceway'), ('ellison',
'ellisonway'), ('endowed', 'endowedway'), ('ashamed', 'ashamedway'), ('eloping',
```

'elopingway'), ('inhabit', 'inhabitway'), ('advised', 'advisedway'), ('attract', 'attractway'), ('already', 'alreadyway'), ('enquire', 'enquireway'), ('affects', 'affectsway'), ('invalid', 'invalidway')], (9, 11): [('genuinely', 'enuinelygay'), ('columella', 'olumellacay'), ('vouchsafe', 'ouchsafevay'), ('forwarded', 'orwardedfay'), ('scattered', 'atteredscay'), ('christmas', 'istmaschray'), ('crimsoned', 'imsonedcray'), ('contrives', 'ontrivescay'), ('miserable', 'iserablemay'), ('preceding', 'ecedingpray'), ('healthful', 'ealthfulhay'), ('confiding', 'onfidingcay'), ('perfectly', 'erfectlypay'), ('prospects', 'ospectspray'), ('withdrawn', 'ithdrawnway'), ('lingering', 'ingeringlay'), ('contained', 'ontainedcay'), ('depressed', 'epressedday'), ('disgusted', 'isgustedday'), ('promoting', 'omotingpray'), ('supported', 'upportedsay'), ('gradually', 'aduallygray'), ('disturbed', 'isturbedday'), ('perpetual', 'erpetualpay'), ('pronounce', 'onouncepray'), ('sincerely', 'incerelysay'), ('seventeen', 'eventeensay'), ('presented', 'esentedpray'), ('medicines', 'edicinesmay'), ('syllables', 'yllablessay'), ('behaviour', 'ehaviourbay'), ('niggardly', 'iggardlynay'), ('commonest', 'ommonestcay'), ('necessary', 'ecessarynay'), ('prettiest', 'ettiestpray'), ('condemned', 'ondemnedcay'), ('conquests', 'onquestscay'), ('fashioned', 'ashionedfay'), ('hysterics', 'ystericshay'), ('smallness', 'allnesssmay'), ('professed', 'ofessedpray'), ('conceited', 'onceitedcay'), ('compliant', 'ompliantcay'), ('frankness', 'anknessfray'), ('merriment', 'errimentmay'), ('resembled', 'esembledray'), ('leisurely', 'eisurelylay'), ('dispersed', 'ispersedday'), ('departure', 'epartureday'), ('pecuniary', 'ecuniarypay'), ('frivolous', 'ivolousfray'), ('situation', 'ituationsay'), ('discourse', 'iscourseday'), ('resettled', 'esettledray'), ('beholding', 'eholdingbay'), ('gentleman', 'entlemangay'), ('chagrined', 'agrinedchay'), ('struggled', 'uggledstray'), ('preparing', 'eparingpray'), ('respected', 'espectedray'), ('newspaper', 'ewspapernay'), ('justified', 'ustifiedjay'), ('sacrifice', 'acrificesay'), ('stammered', 'ammeredstay'), ('smartness', 'artnesssmay'), ('continued', 'ontinuedcay'), ('contrived', 'ontrivedcay'), ('reconcile', 'econcileray'), ('modestest', 'odestestmay'), ('cordially', 'ordiallycay'), ('sentiment', 'entimentsay'), ('shameless', 'amelessshay'), ('reflected', 'eflectedray'), ('feelingly', 'eelinglyfay'), ('confident', 'onfidentcay'), ('proposals', 'oposalspray'), ('warehouse', 'arehouseway'), ('deviation', 'eviationday'), ('statement', 'atementstay'), ('submitted', 'ubmittedsay'), ('believing', 'elievingbay'), ('gentility', 'entilitygay'), ('resources', 'esourcesray'), ('composure', 'omposurecay'), ('deserving', 'eservingday'), ('vehemence', 'ehemencevay'), ('disgraced', 'isgracedday'), ('perplexed', 'erplexedpay'), ('prejudice', 'ejudicepray'), ('quickened', 'ickenedquay'), ('blundered', 'underedblay'), ('strongest', 'ongeststray'), ('prevailed', 'evailedpray'), ('languages', 'anguageslay'), ('reanimate', 'eanimateray'), ('stationer', 'ationerstay'), ('reminding', 'emindingray'), ('blameless', 'amelessblay'), ('principle', 'inciplepray'), ('hardening', 'ardeninghay'), ('suggested', 'uggestedsay'), ('desperate', 'esperateday'), ('landscape', 'andscapelay'), ('paragraph', 'aragraphpay'), ('provision', 'ovisionpray'), ('relations', 'elationsray'), ('fruitless', 'uitlessfray'), ('christian', 'istianchray'), ('stipulate', 'ipulatestay'), ('constancy', 'onstancycay'), ('gentlemen', 'entlemengay'), ('venturing', 'enturingvay'), ('northward', 'orthwardnay'),

```
('neighbour', 'eighbournay'), ('weakening', 'eakeningway'), ('persisted',
'ersistedpay'), ('malignant', 'alignantmay'), ('wandering', 'anderingway'),
('witnesses', 'itnessesway'), ('dangerous', 'angerousday'), ('safeguard',
'afeguardsay'), ('treasured', 'easuredtray'), ('cleveland', 'evelandclay'),
('patronage', 'atronagepay'), ('despising', 'espisingday'), ('foolishly',
'oolishlyfay'), ('seriously', 'eriouslysay'), ('restraint', 'estraintray'),
('september', 'eptembersay'), ('distrusts', 'istrustsday'), ('temperate',
'emperatetay'), ('finishing', 'inishingfay'), ('persuaded', 'ersuadedpay'),
('stupified', 'upifiedstay'), ('declaring', 'eclaringday'), ('dejection',
'ejectionday'), ('soothings', 'oothingssay'), ('conveying', 'onveyingcay'),
('possessed', 'ossessedpay'), ('described', 'escribedday'), ('spreading',
'eadingspray'), ('foresight', 'oresightfay'), ('political', 'oliticalpay'),
('concluded', 'oncludedcay'), ('doubtless', 'oubtlessday'), ('confusion',
'onfusioncay'), ('disliking', 'islikingday'), ('carriages', 'arriagescay'),
('favourite', 'avouritefay'), ('brilliant', 'illiantbray'), ('hastening',
'asteninghay'), ('precisely', 'eciselypray'), ('wondering', 'onderingway'),
('shoulders', 'oulderssshay'), ('honorably', 'onorablyhay'), ('privilege',
'ivilegepray'), ('deference', 'eferenceday'), ('vulgarity', 'ulgarityvay'),
('reprimand', 'eprimandray'), ('surveying', 'urveyingsay'), ('handsomer',
'andsomerhay'), ('shivering', 'iveringshay'), ('tolerable', 'olerabletay'),
('requiring', 'equiringray'), ('precision', 'ecisionpray'), ('courtland',
'ourtlandcay'), ('sportsman', 'ortsmanspay'), ('happiness', 'appinesshay'),
('pointedly', 'ointedlypay'), ('benefited', 'enefitedbay'), ('substance',
'ubstancesay'), ('beginning', 'eginningbay'), ('gratitude', 'atitudegray'),
('negligent', 'egligentnay'), ('belonging', 'elongingbay'), ('mortified',
'ortifiedmay'), ('neglected', 'eglectednay'), ('shillings', 'illingsshay'),
('warranted', 'arrantedway'), ('performed', 'erformedpay'), ('necessity',
'ecessitynay'), ('slightest', 'ightestslay'), ('guardians', 'uardiansgay'),
('preferred', 'eferredpray'), ('forgotten', 'orgottenfay'), ('refreshed',
'efreshedray'), ('barouches', 'arouchesbay'), ('grievance', 'ievancegray'),
('seduction', 'eductionsay'), ('comprised', 'omprisedcay'), ('listening',
'isteninglay'), ('detection', 'etectionday'), ('narration', 'arrationnay'),
('deceiving', 'eceivingday'), ('provoking', 'ovokingpray'), ('cessation',
'essationcay'), ('questions', 'estionsquay'), ('continual', 'ontinualcay'),
('faintness', 'aintnessfay'), ('diffident', 'iffidentday'), ('referring',
'eferringray'), ('blessings', 'essingsblay'), ('therefore', 'ereforethay'),
('frugality', 'ugalityfray'), ('reprobate', 'eprobateray')], (11, 13):
[('tranquilize', 'anquilizetray'), ('fashionable', 'ashionablefay'),
('respectable', 'espectableray'), ('publication', 'ublicationpay'),
('superintend', 'uperintendsay'), ('recollected', 'ecollectedray'),
('cheerfuller', 'eerfullerchay'), ('concurrence', 'oncurrencecay'),
('restorative', 'estorativeray'), ('reproaching', 'eproachingray'),
('declaration', 'eclarationday'), ('discharging', 'ischargingday'),
('proclaiming', 'oclaimingpray'), ('discovering', 'iscoveringday'),
('possibility', 'ossibilitypay'), ('contributed', 'ontributedcay'),
('captivating', 'aptivatingcay'), ('misinformed', 'isinformedmay'),
('carefulness', 'arefulnesscay'), ('philosophic', 'ilosophicphay'),
('calculation', 'alculationcay'), ('consumption', 'onsumptioncay'),
```

```
('confinement', 'onfinementcay'), ('certainties', 'ertaintiescay'),
('destination', 'estinationday'), ('separations', 'eparationssay'),
('voluntarily', 'oluntarilyvay'), ('connoisseur', 'onnoisseurcay'),
('refinements', 'efinementsray'), ('subservient', 'ubservientsay'),
('marlborough', 'arlboroughmay'), ('sagaciously', 'agaciouslysay'),
('fascinating', 'ascinatingfay'), ('comfortable', 'omfortablecay'),
('reflections', 'eflectionsray'), ('sensibility', 'ensibilitysay'),
('forbearance', 'orbearancefay'), ('mischievous', 'ischievousmay'),
('disgraceful', 'isgracefulday'), ('temporizing', 'emporizingtay'),
('sacrificing', 'acrificingsay'), ('spontaneous', 'ontaneousspay'),
('flourishing', 'ourishingflay'), ('lamentation', 'amentationlay'),
('meditations', 'editationsmay'), ('quarrelling', 'arrellingquay'),
('discussions', 'iscussionsday'), ('supposition', 'uppositionsay'),
('comfortably', 'omfortablycay'), ('disappeared', 'isappearedday'),
('preparation', 'eparationpray'), ('consistency', 'onsistencycay'),
('compliments', 'omplimentscay'), ('promptitude', 'omptitudepray'),
('determining', 'eterminingday'), ('countenance', 'ountenancecay'),
('maintenance', 'aintenancemay'), ('contraction', 'ontractioncay'),
('dissipation', 'issipationday'), ('comparative', 'omparativecay'),
('desperation', 'esperationday'), ('displeasure', 'ispleasureday'),
('despatching', 'espatchingday'), ('playfellows', 'ayfellowsplay'),
('plantations', 'antationsplay'), ('vindication', 'indicationvay'),
('significant', 'ignificantsay'), ('discontents', 'iscontentsday'),
('housekeeper', 'ousekeeperhay'), ('playfulness', 'ayfulnessplay'),
('threatening', 'eateningthray'), ('meritorious', 'eritoriousmay'),
('tediousness', 'ediousnesstay'), ('convenience', 'onveniencecay'),
('possessions', 'ossessionspay'), ('consolation', 'onsolationcay'),
('composition', 'ompositioncay'), ('proposition', 'opositionpray'),
('correctness', 'orrectnesscay'), ('consequence', 'onsequencecay'),
('barbarously', 'arbarouslybay'), ('booksellers', 'ooksellersbay'),
('conjectures', 'onjecturescay'), ('violoncello', 'ioloncellovay'),
('dissembling', 'issemblingday'), ('rapturously', 'apturouslyray'),
('speculative', 'eculativespay'), ('remembering', 'ememberingray'),
('probability', 'obabilitypray'), ('punctuality', 'unctualitypay'),
('forgiveness', 'orgivenessfay'), ('disapproves', 'isapprovesday'),
('troublesome', 'oublesometray')], (3, 5): [('say', 'aysay'), ('for', 'orfay'),
('get', 'etgay'), ('pin', 'inpay'), ('not', 'otnay'), ('fit', 'itfay'), ('rap',
'apray'), ('lip', 'iplay'), ('pry', 'ypray'), ('rob', 'obray'), ('fir',
'irfay'), ('can', 'ancay'), ('two', 'otway'), ('him', 'imhay'), ('son',
'onsay'), ('let', 'etlay'), ('try', 'ytray'), ('hot', 'othay'), ('sum',
'umsay'), ('got', 'otgay'), ('law', 'awlay'), ('how', 'owhay'), ('far',
'arfay'), ('the', 'ethay'), ('sky', 'yskay'), ('sly', 'yslay'), ('cut',
'utcay'), ('hop', 'ophay'), ('had', 'adhay'), ('was', 'asway'), ('bow',
'owbay'), ('hue', 'uehay'), ('now', 'ownay'), ('vex', 'exvay'), ('yet',
'yetay'), ('sit', 'itsay'), ('due', 'ueday'), ('cap', 'apcay'), ('sat',
'atsay'), ('hit', 'ithay'), ('see', 'eesay'), ('pay', 'aypay'), ('nor',
'ornay'), ('fly', 'yflay'), ('dog', 'ogday'), ('too', 'ootay'), ('six',
'ixsay'), ('lap', 'aplay'), ('new', 'ewnay'), ('led', 'edlay'), ('sin',
```

'insay'), ('ten', 'entay'), ('his', 'ishay'), ('day', 'ayday')], (14, 17):
[('insurmountable', 'insurmountableway'), ('inconveniences',
'inconveniencesway'), ('aggrandizement', 'aggrandizementway'),
('irreconcilable', 'irreconcilableway'), ('unacknowledged',
'unacknowledgedway'), ('unpleasantness', 'unpleasantnessway')], (13, 16):
[('understanding', 'understandingway'), ('intentionally', 'intentionallyway'),
('embarrassment', 'embarrassmentway'), ('unembarrassed', 'unembarrassedway'),
('administering', 'administeringway'), ('instantaneous', 'instantaneousway'),
('intrinsically', 'intrinsicallyway'), ('indispensable', 'indispensableway'),
('impoverishing', 'impoverishingway'), ('opportunities', 'opportunitiesway'),
('impertinently', 'impertinentlyway'), ('independently', 'independentlyway'),
('encroachments', 'encroachmentsway'), ('uninteresting', 'uninterestingway'),
('investigation', 'investigationway'), ('unaccountable', 'unaccountableway'),
('irrepressible', 'irrepressibleway'), ('impossibility', 'impossibilityway')],
(10, 14): [('thirty-six', 'irtythay-ixsay')], (8, 11): [('increase',
'increaseway'), ('agitated', 'agitatedway'), ('unkindly', 'unkindlyway'),
('occurred', 'occurredway'), ('arranged', 'arrangedway'), ('appetite',
'appetiteway'), ('inclined', 'inclinedway'), ('indolent', 'indolentway'),
('inquired', 'inquiredway'), ('evenings', 'eveningsway'), ('obliging',
'obligingway'), ('equalled', 'equalledway'), ('accepted', 'acceptedway'),
('extended', 'extendedway'), ('entreaty', 'entreatyway'), ('exigence',
'exigenceway'), ('election', 'electionway'), ('examined', 'examinedway'),
('inviting', 'invitingway'), ('embitter', 'embitterway'), ('idleness',
'idlenessway'), ('admirers', 'admirersway'), ('editions', 'editionsway'),
('extorted', 'extortedway'), ('outweigh', 'outweighway'), ('attended',
'attendedway'), ('addition', 'additionway'), ('entitled', 'entitledway'),
('awakened', 'awakenedway'), ('attacked', 'attackedway'), ('arriving',
'arrivingway'), ('overlook', 'overlookway'), ('eminence', 'eminenceway'),
('adhering', 'adheringway'), ('obtained', 'obtainedway'), ('openness',
'opennessway'), ('allusion', 'allusionway'), ('anything', 'anythingway'),
('overcome', 'overcomeway'), ('announce', 'announceway'), ('enjoying',
'enjoyingway'), ('outlived', 'outlivedway'), ('emphasis', 'emphasisway'),
('insulted', 'insultedway'), ('esteemed', 'esteemedway'), ('alphabet',
'alphabetway'), ('enquired', 'enquiredway'), ('accident', 'accidentway'),
('occupied', 'occupiedway'), ('ellisons', 'ellisonsway'), ('avoiding',
'avoidingway'), ('attained', 'attainedway'), ('unwarily', 'unwarilyway'),
('employed', 'employedway'), ('altering', 'alteringway'), ('intimate',
'intimateway'), ('effected', 'effectedway'), ('offences', 'offencesway'),
('entirely', 'entirelyway'), ('abounded', 'aboundedway'), ('actually',
'actuallyway'), ('identify', 'identifyway'), ('applying', 'applyingway'),
('exterior', 'exteriorway'), ('auditors', 'auditorsway'), ('unfairly',
'unfairlyway'), ('eighteen', 'eighteenway'), ('embraced', 'embracedway'),
('unfolded', 'unfoldedway'), ('injuries', 'injuriesway'), ('elevated',
'elevatedway'), ('alarming', 'alarmingway'), ('adequate', 'adequateway'),
('unabated', 'unabatedway'), ('incurred', 'incurredway'), ('uncommon',
'uncommonway'), ('unawares', 'unawaresway'), ('unsuited', 'unsuitedway'),
('instance', 'instanceway'), ('interval', 'intervalway'), ('audacity',
'audacityway')], (5, 8): [('other', 'otherway'), ('ideas', 'ideasway'),

('agony', 'agonyway'), ('avoid', 'avoidway'), ('awful', 'awfulway'), ('april',
'aprilway'), ('enter', 'enterway'), ('infer', 'inferway'), ('abuse',
'abuseway'), ('ivory', 'ivoryway'), ('adopt', 'adoptway'), ('image',
'imageway'), ('agree', 'agreeway'), ('alike', 'alikeway'), ('earth',
'earthway'), ('ended', 'endedway'), ('owner', 'ownerway'), ('using',
'usingway'), ('ought', 'oughtway'), ('abhor', 'abhorway'), ('altar',
'altarway'), ('often', 'oftenway'), ('atone', 'atoneway'), ('angel',
'angelway'), ('arise', 'ariseway'), ('under', 'underway'), ('abode',
'abodeway'), ('exert', 'exertway'), ('usual', 'usualway'), ('aches',
'achesway'), ('enjoy', 'enjoyway'), ('urged', 'urgedway'), ('admit',
'admitway'), ('aside', 'asideway'), ('equal', 'equalway'), ('alive',
'aliveway'), ('arose', 'aroseway'), ('eight', 'eightway'), ('opera',
'operaway'), ('ennui', 'ennuiway'), ('utter', 'utterway'), ('ankle',
'ankleway')], (2, 5): [('is', 'isway'), ('ay', 'ayway'), ('ah', 'ahway'), ('ii',
'iiway'), ('or', 'orway'), ('am', 'amway'), ('if', 'ifway'), ('an', 'anway')],
(12, 14): [('tranquillity', 'anquillitytray'), ('humiliations',
'umiliationshay'), ('contributing', 'ontributingcay'), ('significancy',
'ignificancysay'), ('wretchedness', 'etchednesswray'), ('prodigiously',
'odigiouslypray'), ('particularly', 'articularlypay'), ('carelessness',
'arelessnesscay'), ('refreshments', 'efreshmentsray'), ('satisfaction',
'atisfactionsay'), ('construction', 'onstructioncay'), ('contribution',
'ontributioncay'), ('gracefulness', 'acefulnessgray'), ('consequences',
'onsequencescay'), ('grandmothers', 'andmothersgray'), ('proportioned',
'oportionedpray'), ('perverseness', 'erversenesspay'), ('conciliation',
'onciliationcay'), ('representing', 'epresentingray'), ('watchfulness',
'atchfulnessway'), ('recollecting', 'ecollectingray'), ('hesitatingly',
'esitatinglyhay'), ('pleasantness', 'easantnessplay'), ('benefactress',
'enefactressbay'), ('distractedly', 'istractedlyday'), ('perseverance',
'erseverancepay'), ('performances', 'erformancespay'), ('disapproving',
'isapprovingday'), ('recollection', 'ecollectionray'), ('comprehended',
'omprehendedcay'), ('relinquished', 'elinquishedray'), ('conjecturing',
'onjecturingcay'), ('reservedness', 'eservednessray'), ('promontories',
'omontoriespray'), ('discontented', 'iscontentedday'), ('persecutions',
'ersecutionspay'), ('relationship', 'elationshipray'), ('triumphantly',
'iumphantlytray'), ('deliberation', 'eliberationday'), ('supplication',
'upplicationsay'), ('dishonorable', 'ishonorableday'), ('propensities',
'opensitiespray'), ('contemptuous', 'ontemptuouscay'), ('transporting',
'ansportingtray')], (13, 15): [('distinguished', 'istinguishedday'),
('resuscitation', 'esuscitationray'), ('particularity', 'articularitypay'),
('superannuated', 'uperannuatedsay'), ('disagreements', 'isagreementsday'),
('gentlemanlike', 'entlemanlikegay'), ('prepossessing', 'epossessingpray'),
('conscientious', 'onscientiouscay'), ('considerately', 'onsideratelycay'),
('felicitations', 'elicitationsfay'), ('prepossession', 'epossessionpray'),
('consternation', 'onsternationcay'), ('disproportion', 'isproportionday'),
('commiseration', 'ommiserationcay'), ('mortification', 'ortificationmay'),
('solicitations', 'olicitationssay'), ('qualification', 'alificationquay'),
('philanthropic', 'ilanthropicphay'), ('disappointing', 'isappointingday'),
('prescriptions', 'escriptionspray'), ('strengthening', 'engtheningstray')],

(13, 17): [('mulberry-tree', 'ulberrymay-eetray'), ('good-humoured', 'oodgay-umouredhay'), ('mean-spirited', 'eanmay-iritedspay'), ('self-reproach', 'elfsay-eproachray'), ('drawing-table', 'awingdray-abletay'), ('well-disposed', 'ellway-isposedday'), ('proud-looking', 'oudpray-ookinglay'), ('flower-garden', 'owerflay-ardengay'), ('mansion-house', 'ansionmay-ousehay'), ('landing-place', 'andinglay-aceplay'), ('music-sellers', 'usicmay-ellerssay')], (4, 7): [('acts', 'actsway'), ('anne', 'anneway'), ('upon', 'uponway'), ('able', 'ableway'), ('inch', 'inchway'), ('ages', 'agesway'), ('envy', 'envyway'), ('idea', 'ideaway'), ('ends', 'endsway'), ('-and', '-andway'), ('eyed', 'eyedway'), ('arms', 'armsway'), ('over', 'overway'), ('easy', 'easyway'), ('ever', 'everway'), ('ours', 'oursway'), ('ears', 'earsway'), ('away', 'awayway')], (16, 20): [('pleasure-grounds', 'easureplay-oundsgray'), ('self-provocation', 'elfsay-ovocationpray')], (11, 15): [('bed-chamber', 'edbay-amberchay'), ('high-minded', 'ighhay-indedmay'), ('before-hand', 'eforebay-andhay'), ('watch-tower', 'atchway-owertay'), ('print-shops', 'intpray-opsshay'), ('card-tables', 'ardcay-ablestay'), ('post-horses', 'ostpay-orseshay'), ('good-nature', 'oodgay-aturenay'), ('dining-room', 'iningday-oomray'), ('high-church', 'ighhay-urchchay'), ('well-wisher', 'ellway-isherway'), ('bond-street', 'ondbay-eetstray'), ('carpet-work', 'arpetcay-orkway')], (2, 4): [('to', 'otay'), ('he', 'ehay'), ('no', 'onay'), ('me', 'emay')], (12, 16): [('good-hearted', 'oodgay-eartedhay'), ('well-judging', 'ellway-udgingjay'), ('drawing-room', 'awingdray-oomray'), ('common-place', 'ommoncay-aceplay'), ('self-control', 'elfsay-ontrolcay'), ('hand-writing', 'andhay-itingwray'), ('well-behaved', 'ellway-ehavedbay'), ('mantel-piece', 'antelmay-iecepay')], (10, 15): [('ill-humour', 'illway-umourhay'), ('ill-judged', 'illway-udgedjay'), ('ill-suited', 'illway-uitedsay')], (1, 3): [('c', 'cay')], (8, 13): [('ill-wind', 'illway-indway'), ('ill-will', 'illway-illway'), ('over-set', 'overway-etsay')], (17, 19): [('disqualifications', 'isqualificationsday')], (9, 16): [('by-and-by', 'ybay-andway-ybay')], (3, 6): [('aye', 'ayeway'), ('use', 'useway'), ('owe', 'oweway'), ('our', 'ourway'), ('old', 'oldway'), ('ear', 'earway'), ('its', 'itsway'), ('ill', 'illway')], (12, 17): [('ill-breeding', 'illway-eedingbray'), ('re-assurance', 'eray-assuranceway'), ('over-powered', 'overway-oweredpay'), ('ill-disposed', 'illway-isposedday')], (11, 16): [('ill-natured', 'illway-aturednay')], (14, 16): [('reconciliation', 'econciliationray'), ('recommendation', 'ecommendationray'), ('reasonableness', 'easonablenessray'), ('disinclination', 'isinclinationday'), ('respectability', 'espectabilityray'), ('circumspection', 'ircumspectioncay'), ('considerations', 'onsiderationscay'), ('qualifications', 'alificationsquay'), ('congratulating', 'ongratulatingcay')], (15, 17): [('dissatisfaction', 'issatisfactionday'), ('disrespectfully', 'isrespectfullyday'), ('conscientiously', 'onscientiouslycay'), ('misconstruction', 'isconstructionmay')], (15, 18): [('instantaneously', 'instantaneouslyway')], (15, 19): [('shooting-jacket', 'ootingshay-acketjay')], (14, 21): [('sisters-in-law', 'isterssay-inway-awlay')], (17, 21): [('self-condemnation', 'elfsay-ondemnationcay')], (13, 20): [('sister-in-law', 'istersay-inway-awlay')], (9, 14): [('ill-timed', 'illway-imedtay')], (13, 18): [('ill-treatment', 'illway-eatmenttray'), ('old-fashioned', 'oldway-ashionedfay'), ('self-interest', 'elfsay-interestway')], (16, 22): [('good-for-nothing', 'oodgay-orfay-othingnay')], (9, 13): [('dove-cote', 'oveday-otecay'),

```
('tea-table', 'eatay-abletay'), ('work-bags', 'orkway-agsbay')], (1, 4): [('i',
'iway')], (16, 19): [('incomprehensible', 'incomprehensibleway')], (8, 12):
[('to-night', 'otay-ightnay'), ('land-tax', 'andlay-axtay')], (11, 18): [('tete-
a-tete', 'etetay-away-etetay')], (5, 9): [('to-do', 'otay-oday')]})
Moved models to GPU!
Epoch:   0 | Flops (T): 0.024908284704, Tokens (M) 0.052678, Train Loss 2.968,
Val Loss 2.966 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   1 | Flops (T): 0.049816569408, Tokens (M) 0.105356, Train Loss 2.896,
Val Loss 2.893 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   2 | Flops (T): 0.074724854112, Tokens (M) 0.158034, Train Loss 2.855,
Val Loss 2.867 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   3 | Flops (T): 0.099633138816, Tokens (M) 0.210712, Train Loss 2.819,
Val Loss 2.846 | Gen: yyyyyyyyyyyyaaaaaaaa
Epoch:   4 | Flops (T): 0.12454142352, Tokens (M) 0.26339, Train Loss 2.777, Val
Loss 2.807 | Gen: yyyyyyyyyyyyaaaaaaaa
Epoch:   5 | Flops (T): 0.149449708224, Tokens (M) 0.316068, Train Loss 2.730,
Val Loss 2.785 | Gen: yyyyyyyyyyyyyyyyyyyy
Epoch:   6 | Flops (T): 0.174357992928, Tokens (M) 0.368746, Train Loss 2.684,
Val Loss 2.723 | Gen: aaaaayyyyyyyyyyyyyaa
Epoch:   7 | Flops (T): 0.199266277632, Tokens (M) 0.421424, Train Loss 2.468,
Val Loss 2.422 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   8 | Flops (T): 0.224174562336, Tokens (M) 0.474102, Train Loss 2.175,
Val Loss 2.204 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   9 | Flops (T): 0.24908284704, Tokens (M) 0.52678, Train Loss 2.045, Val
Loss 2.284 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  10 | Flops (T): 0.273991131744, Tokens (M) 0.579458, Train Loss 1.856,
Val Loss 2.218 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  11 | Flops (T): 0.298899416448, Tokens (M) 0.632136, Train Loss 1.766,
Val Loss 1.938 | Gen: waaaaaaaaaaaaaaaaaaa
Epoch:  12 | Flops (T): 0.323807701152, Tokens (M) 0.684814, Train Loss 1.737,
Val Loss 1.914 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  13 | Flops (T): 0.348715985856, Tokens (M) 0.737492, Train Loss 1.730,
Val Loss 2.304 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  14 | Flops (T): 0.37362427056, Tokens (M) 0.79017, Train Loss 1.803, Val
Loss 2.054 | Gen: oooooooooooooooooooo
Epoch:  15 | Flops (T): 0.398532555264, Tokens (M) 0.842848, Train Loss 1.663,
Val Loss 1.679 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  16 | Flops (T): 0.423440839968, Tokens (M) 0.895526, Train Loss 1.593,
Val Loss 1.639 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  17 | Flops (T): 0.448349124672, Tokens (M) 0.948204, Train Loss 1.560,
Val Loss 1.599 | Gen: oooaaaaaaaaaaaaaaaaa
Epoch:  18 | Flops (T): 0.473257409376, Tokens (M) 1.000882, Train Loss 1.536,
Val Loss 1.581 | Gen: oooooooooooooooooooo
Epoch:  19 | Flops (T): 0.49816569408, Tokens (M) 1.05356, Train Loss 1.514, Val
Loss 1.612 | Gen: oooooooooooooooooooo
Epoch:  20 | Flops (T): 0.523073978784, Tokens (M) 1.106238, Train Loss 1.503,
Val Loss 1.557 | Gen: oooooooooooooooooooo
Epoch:  21 | Flops (T): 0.547982263488, Tokens (M) 1.158916, Train Loss 1.497,
```

```
Val Loss 1.579 | Gen: oooooooooooooooooooo
Epoch:  22 | Flops (T): 0.572890548192, Tokens (M) 1.211594, Train Loss 1.540,
Val Loss 1.543 | Gen: oooooooooooooooooooo
Epoch:  23 | Flops (T): 0.597798832896, Tokens (M) 1.264272, Train Loss 1.479,
Val Loss 1.510 | Gen: oooooooooooooooooooo
Epoch:  24 | Flops (T): 0.6227071176, Tokens (M) 1.31695, Train Loss 1.474, Val
Loss 1.484 | Gen: oooooooooooooooooooo
Epoch:  25 | Flops (T): 0.647615402304, Tokens (M) 1.369628, Train Loss 1.502,
Val Loss 1.936 | Gen: oooooooooaowwwwwwwwww
Epoch:  26 | Flops (T): 0.672523687008, Tokens (M) 1.422306, Train Loss 1.655,
Val Loss 1.630 | Gen: ooaaaaaaaaaaaaaaaaaa
Epoch:  27 | Flops (T): 0.697431971712, Tokens (M) 1.474984, Train Loss 1.547,
Val Loss 1.559 | Gen: oooooooooooooooooooo
Epoch:  28 | Flops (T): 0.722340256416, Tokens (M) 1.527662, Train Loss 1.485,
Val Loss 1.539 | Gen: oooooooooooooooooooa
Epoch:  29 | Flops (T): 0.74724854112, Tokens (M) 1.58034, Train Loss 1.464, Val
Loss 1.505 | Gen: oooooooooooooooooooo
Epoch:  30 | Flops (T): 0.772156825824, Tokens (M) 1.633018, Train Loss 1.442,
Val Loss 1.494 | Gen: oooooooooooooooooooo
Epoch:  31 | Flops (T): 0.797065110528, Tokens (M) 1.685696, Train Loss 1.420,
Val Loss 1.468 | Gen: ooofffffwyyyyyyyyyy
Epoch:  32 | Flops (T): 0.821973395232, Tokens (M) 1.738374, Train Loss 1.408,
Val Loss 1.445 | Gen: oofffffffwyyyyyyyyyy
Epoch:  33 | Flops (T): 0.846881679936, Tokens (M) 1.791052, Train Loss 1.396,
Val Loss 1.430 | Gen: offffffffwyyyyyyyyyy
Epoch:  34 | Flops (T): 0.87178996464, Tokens (M) 1.84373, Train Loss 1.395, Val
Loss 1.436 | Gen: ooooffffafwyyyyyyyyy
Epoch:  35 | Flops (T): 0.896698249344, Tokens (M) 1.896408, Train Loss 1.379,
Val Loss 1.415 | Gen: offffffffwyyyyyyyyyy
Epoch:  36 | Flops (T): 0.921606534048, Tokens (M) 1.949086, Train Loss 1.372,
Val Loss 1.398 | Gen: oofffffffwyyyyyyyyyy
Epoch:  37 | Flops (T): 0.946514818752, Tokens (M) 2.001764, Train Loss 1.368,
Val Loss 1.393 | Gen: offffffffwyyyyyyyyyy
Epoch:  38 | Flops (T): 0.971423103456, Tokens (M) 2.054442, Train Loss 1.351,
Val Loss 1.397 | Gen: offffffffkyyyyyyyyyy
Epoch:  39 | Flops (T): 0.99633138816, Tokens (M) 2.10712, Train Loss 1.344, Val
Loss 1.381 | Gen: offffffffkyyyyyyyyyy
Epoch:  40 | Flops (T): 1.021239672864, Tokens (M) 2.159798, Train Loss 1.336,
Val Loss 1.372 | Gen: ffffffffkyyyyyyyyyy
Epoch:  41 | Flops (T): 1.046147957568, Tokens (M) 2.212476, Train Loss 1.326,
Val Loss 1.364 | Gen: offffffffkyyyyyyyyyy
Epoch:  42 | Flops (T): 1.071056242272, Tokens (M) 2.265154, Train Loss 1.322,
Val Loss 1.357 | Gen: ffffffffkyyyyyyyyyy
Epoch:  43 | Flops (T): 1.095964526976, Tokens (M) 2.317832, Train Loss 1.335,
Val Loss 1.360 | Gen: offffffffkyyyyyyyyyy
Epoch:  44 | Flops (T): 1.12087281168, Tokens (M) 2.37051, Train Loss 1.320, Val
Loss 1.357 | Gen: ffffffffkyyyyyyyyyy
Epoch:  45 | Flops (T): 1.145781096384, Tokens (M) 2.423188, Train Loss 1.308,
```

```
Val Loss 1.349 | Gen: offffffkyyyyyyyyyy
Epoch:  46 | Flops (T): 1.170689381088, Tokens (M) 2.475866, Train Loss 1.310,
Val Loss 1.409 | Gen: offffffkyyyyyyyyyy
Epoch:  47 | Flops (T): 1.195597665792, Tokens (M) 2.528544, Train Loss 1.321,
Val Loss 1.360 | Gen: ffffffkyyyyyyyyyyy
Epoch:  48 | Flops (T): 1.220505950496, Tokens (M) 2.581222, Train Loss 1.319,
Val Loss 1.389 | Gen: ffffffffkyyyyyyyyyy
Epoch:  49 | Flops (T): 1.2454142352, Tokens (M) 2.6339, Train Loss 1.302, Val
Loss 1.335 | Gen: ffffffffkyyyyyyyyyy
Epoch:  50 | Flops (T): 1.270322519904, Tokens (M) 2.686578, Train Loss 1.293,
Val Loss 1.336 | Gen: ffffffffkyyyyyyyyyy
Epoch:  51 | Flops (T): 1.295230804608, Tokens (M) 2.739256, Train Loss 1.282,
Val Loss 1.323 | Gen: ffffffkwyyyyyyyyyy
Epoch:  52 | Flops (T): 1.320139089312, Tokens (M) 2.791934, Train Loss 1.278,
Val Loss 1.330 | Gen: offffffkyyyyyyyyyy
Epoch:  53 | Flops (T): 1.345047374016, Tokens (M) 2.844612, Train Loss 1.269,
Val Loss 1.316 | Gen: offffffkyyyyyyyyyy
Epoch:  54 | Flops (T): 1.36995565872, Tokens (M) 2.89729, Train Loss 1.270, Val
Loss 1.342 | Gen: offffffkyyyyyyyyyy
Epoch:  55 | Flops (T): 1.394863943424, Tokens (M) 2.949968, Train Loss 1.272,
Val Loss 1.330 | Gen: ffffffkwyyyyyyyyyy
Epoch:  56 | Flops (T): 1.419772228128, Tokens (M) 3.002646, Train Loss 1.278,
Val Loss 1.355 | Gen: afffffafwyyyyyyyyy
Epoch:  57 | Flops (T): 1.444680512832, Tokens (M) 3.055324, Train Loss 1.265,
Val Loss 1.321 | Gen: ffffffffkyyyyyyyyyy
Epoch:  58 | Flops (T): 1.469588797536, Tokens (M) 3.108002, Train Loss 1.252,
Val Loss 1.299 | Gen: offffffkyyyyyyyyyy
Epoch:  59 | Flops (T): 1.49449708224, Tokens (M) 3.16068, Train Loss 1.254, Val
Loss 1.296 | Gen: ffffffffw-----------
Epoch:  60 | Flops (T): 1.519405366944, Tokens (M) 3.213358, Train Loss 1.241,
Val Loss 1.404 | Gen: offffffwyyyyyyyyyy
Epoch:  61 | Flops (T): 1.544313651648, Tokens (M) 3.266036, Train Loss 1.255,
Val Loss 1.289 | Gen: ffffffffwyyyyyyyyyy
Epoch:  62 | Flops (T): 1.569221936352, Tokens (M) 3.318714, Train Loss 1.233,
Val Loss 1.293 | Gen: ooffffffwyyyyyyyyy
Epoch:  63 | Flops (T): 1.594130221056, Tokens (M) 3.371392, Train Loss 1.233,
Val Loss 1.276 | Gen: fffffffffwyyyyyyyyy
Epoch:  64 | Flops (T): 1.61903850576, Tokens (M) 3.42407, Train Loss 1.242, Val
Loss 1.335 | Gen: afffaaaaaaaaaaaaaaaa
Epoch:  65 | Flops (T): 1.643946790464, Tokens (M) 3.476748, Train Loss 1.255,
Val Loss 1.303 | Gen: aaffaaaaaaaaaaaaaaaa
Epoch:  66 | Flops (T): 1.668855075168, Tokens (M) 3.529426, Train Loss 1.231,
Val Loss 1.305 | Gen: oaaaaaaaaaaaaaaaaaaa
Epoch:  67 | Flops (T): 1.693763359872, Tokens (M) 3.582104, Train Loss 1.224,
Val Loss 1.277 | Gen: aaffaaaaaaaaaaaaaaaa
Epoch:  68 | Flops (T): 1.718671644576, Tokens (M) 3.634782, Train Loss 1.215,
Val Loss 1.272 | Gen: ooaaaaaaaaaaaaaaaaaa
Epoch:  69 | Flops (T): 1.74357992928, Tokens (M) 3.68746, Train Loss 1.207, Val
```

```
Loss 1.262 | Gen: oaffffffwyyyyyyyyy
Epoch:  70 | Flops (T): 1.768488213984, Tokens (M) 3.740138, Train Loss 1.204,
Val Loss 1.274 | Gen: oaaaaaaaaaaaaaaaaaaa
Epoch:  71 | Flops (T): 1.793396498688, Tokens (M) 3.792816, Train Loss 1.213,
Val Loss 1.270 | Gen: affffffffwyyyyyyyyy
Epoch:  72 | Flops (T): 1.818304783392, Tokens (M) 3.845494, Train Loss 1.209,
Val Loss 1.270 | Gen: oaaaaaaaaaaaaaaaaaaa
Epoch:  73 | Flops (T): 1.843213068096, Tokens (M) 3.898172, Train Loss 1.196,
Val Loss 1.258 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  74 | Flops (T): 1.8681213528, Tokens (M) 3.95085, Train Loss 1.194, Val
Loss 1.262 | Gen: oaaaaaaaaaaaaaaaaaaa
Epoch:  75 | Flops (T): 1.893029637504, Tokens (M) 4.003528, Train Loss 1.190,
Val Loss 1.268 | Gen: oofaaaaaaaaaaaaaaaaa
Epoch:  76 | Flops (T): 1.917937922208, Tokens (M) 4.056206, Train Loss 1.190,
Val Loss 1.281 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  77 | Flops (T): 1.942846206912, Tokens (M) 4.108884, Train Loss 1.202,
Val Loss 1.448 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  78 | Flops (T): 1.967754491616, Tokens (M) 4.161562, Train Loss 1.235,
Val Loss 1.291 | Gen: ooofaaaaaaaaaaaaaaaa
Epoch:  79 | Flops (T): 1.99266277632, Tokens (M) 4.21424, Train Loss 1.200, Val
Loss 1.239 | Gen: offaaaaaaaaaaaaaaaaa
Epoch:  80 | Flops (T): 2.017571061024, Tokens (M) 4.266918, Train Loss 1.209,
Val Loss 1.338 | Gen: oooaaaaaaaaaaaaaaaaa
Epoch:  81 | Flops (T): 2.042479345728, Tokens (M) 4.319596, Train Loss 1.182,
Val Loss 1.282 | Gen: offffaaaaaaaaaaaaaaa
Epoch:  82 | Flops (T): 2.067387630432, Tokens (M) 4.372274, Train Loss 1.186,
Val Loss 1.264 | Gen: ofaaaaaaaaaaaaaaaaaa
Epoch:  83 | Flops (T): 2.092295915136, Tokens (M) 4.424952, Train Loss 1.179,
Val Loss 1.241 | Gen: ooaaaaaaaaaaaaaaaaaa
Epoch:  84 | Flops (T): 2.11720419984, Tokens (M) 4.47763, Train Loss 1.173, Val
Loss 1.225 | Gen: ooffaaaaaaaaaaaaaaaa
Epoch:  85 | Flops (T): 2.142112484544, Tokens (M) 4.530308, Train Loss 1.175,
Val Loss 1.294 | Gen: ooaaaaaaaaaaaaaaaaaa
Epoch:  86 | Flops (T): 2.167020769248, Tokens (M) 4.582986, Train Loss 1.166,
Val Loss 1.242 | Gen: oooaaaaaaaaaaaaaaaaa
Epoch:  87 | Flops (T): 2.191929053952, Tokens (M) 4.635664, Train Loss 1.168,
Val Loss 1.249 | Gen: oaaaaaaaaaaaaaaaaaaa
Epoch:  88 | Flops (T): 2.216837338656, Tokens (M) 4.688342, Train Loss 1.160,
Val Loss 1.245 | Gen: ooaaaaaaaaaaaaaaaaaa
Epoch:  89 | Flops (T): 2.24174562336, Tokens (M) 4.74102, Train Loss 1.167, Val
Loss 1.265 | Gen: laaaaaaaaaaaaaaaaaaa
Epoch:  90 | Flops (T): 2.266653908064, Tokens (M) 4.793698, Train Loss 1.166,
Val Loss 1.287 | Gen: ooooaaaaaaaaaaaaaaaa
Epoch:  91 | Flops (T): 2.291562192768, Tokens (M) 4.846376, Train Loss 1.168,
Val Loss 1.212 | Gen: oaaaaaaaaaaaaaaaaaaa
Epoch:  92 | Flops (T): 2.316470477472, Tokens (M) 4.899054, Train Loss 1.151,
Val Loss 1.211 | Gen: ooaaaaaaaaaaaaaaaaaa
Epoch:  93 | Flops (T): 2.341378762176, Tokens (M) 4.951732, Train Loss 1.146,
```

```
Val Loss 1.211 | Gen: faaaaaaaaaaaaaaaaaa
Epoch:  94 | Flops (T): 2.36628704688, Tokens (M) 5.00441, Train Loss 1.149, Val
Loss 1.224 | Gen: ooaaaaaaaaaaaaaaaaaa
Epoch:  95 | Flops (T): 2.391195331584, Tokens (M) 5.057088, Train Loss 1.138,
Val Loss 1.196 | Gen: oooaaaaaaaaaaaaaaaaa
Epoch:  96 | Flops (T): 2.416103616288, Tokens (M) 5.109766, Train Loss 1.151,
Val Loss 1.329 | Gen: oooaaaaaaaaaaaaaaaaa
Epoch:  97 | Flops (T): 2.441011900992, Tokens (M) 5.162444, Train Loss 1.182,
Val Loss 1.233 | Gen: ooaaaaaaaaaaaaaaaaaa
Epoch:  98 | Flops (T): 2.465920185696, Tokens (M) 5.215122, Train Loss 1.147,
Val Loss 1.193 | Gen: ooaaaaaaaaaaaaaaaaaa
Epoch:  99 | Flops (T): 2.4908284704, Tokens (M) 5.2678, Train Loss 1.137, Val
Loss 1.208 | Gen: looaaaaaaaaaaaaaaaaaa
Obtained lowest validation loss of: 1.1929125238296598
Avg tokens per example (Seq Len): 20.59343236903831, Avg flops per example:
9737406.060985144
```

[ ]:
```python
TEST_SENTENCE = "floor"
trans96_args_s = AttrDict()
args_dict_96 = {
    "data_file_name": "pig_latin_small", #pig_latin_small debug
    "cuda": use_cuda,
    "nepochs": 100,
    "checkpoint_dir": "checkpoints",
    "learning_rate": 5e-4,
    "early_stopping_patience": 100,
    "lr_decay": 0.99,
    "batch_size": 64,
    "hidden_size": 96,
    "num_transformer_layers": 4,
    "loss_seq_span": "target_only",
    "num_layers": 4,
    "vocab_size": 30
}
trans96_args_s.update(args_dict_96)
print_opts(trans96_args_s)
trans96_decoder_s, trans96_losses_s, idx_dict = train(trans96_args_s)
```

```
================================================================================
                                        Opts
--------------------------------------------------------------------------------
                          data_file_name: pig_latin_small
                                    cuda: 1
                                 nepochs: 100
                          checkpoint_dir: checkpoints
                           learning_rate: 0.0005
                  early_stopping_patience: 100
                                 lr_decay: 0.99
```

```
                            batch_size: 64
                           hidden_size: 96
                num_transformer_layers: 4
                          loss_seq_span: target_only
                            num_layers: 4
                            vocab_size: 30
================================================================================
================================================================================
                                   Data Stats
--------------------------------------------------------------------------------
('irresistible', 'irresistibleway')
('repentance', 'epentanceray')
('pretty', 'ettypray')
('parcels', 'arcelspay')
('survey', 'urveysay')
Num unique word pairs: 3198
Vocabulary: dict_keys(['-', 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j',
'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z',
'SOS', 'EOS', 'EOP'])
Vocab size: 30
================================================================================
defaultdict(<class 'list'>, {(12, 15): [('irresistible', 'irresistibleway'),
('advantageous', 'advantageousway'), ('impartiality', 'impartialityway'),
('interruption', 'interruptionway'), ('anticipation', 'anticipationway'),
('impenetrable', 'impenetrableway'), ('excellencies', 'excellenciesway'),
('intermission', 'intermissionway'), ('interspersed', 'interspersedway'),
('unaffectedly', 'unaffectedlyway'), ('insinuations', 'insinuationsway'),
('unpardonable', 'unpardonableway'), ('arrangements', 'arrangementsway'),
('unexpectedly', 'unexpectedlyway'), ('introduction', 'introductionway'),
('occasionally', 'occasionallyway'), ('establishing', 'establishingway'),
('unanswerable', 'unanswerableway'), ('independence', 'independenceway'),
('inclinations', 'inclinationsway'), ('instructions', 'instructionsway'),
('undoubtingly', 'undoubtinglyway'), ('unfathomable', 'unfathomableway'),
('impoverished', 'impoverishedway'), ('outstretched', 'outstretchedway'),
('unrestrained', 'unrestrainedway'), ('ungraciously', 'ungraciouslyway'),
('imperfection', 'imperfectionway'), ('unreasonably', 'unreasonablyway'),
('interference', 'interferenceway'), ('inconvenient', 'inconvenientway'),
('enthusiastic', 'enthusiasticway'), ('extinguished', 'extinguishedway'),
('entanglement', 'entanglementway'), ('affectionate', 'affectionateway')], (10,
12): [('repentance', 'epentanceray'), ('capability', 'apabilitycay'),
('beneficial', 'eneficialbay'), ('retrenched', 'etrenchedray'), ('devonshire',
'evonshireday'), ('palanquins', 'alanquinspay'), ('contribute', 'ontributecay'),
('minuteness', 'inutenessmay'), ('dissimilar', 'issimilarday'), ('flattering',
'atteringflay'), ('confounded', 'onfoundedcay'), ('compromise', 'ompromisecay'),
('contrasted', 'ontrastedcay'), ('precaution', 'ecautionpray'), ('fastidious',
'astidiousfay'), ('blackguard', 'ackguardblay'), ('heroically', 'eroicallyhay'),
('lengthened', 'engthenedlay'), ('pocketbook', 'ocketbookpay'), ('companions',
'ompanionscay'), ('displeased', 'ispleasedday'), ('pianoforte', 'ianofortepay'),
```

('succession', 'uccessionsay'), ('composedly', 'omposedlycay'), ('pronounced', 'onouncedpray'), ('honourable', 'onourablehay'), ('solicitude', 'olicitudesay'), ('justifying', 'ustifyingjay'), ('collecting', 'ollectingcay'), ('headstrong', 'eadstronghay'), ('forbidding', 'orbiddingfay'), ('successful', 'uccessfulsay'), ('confidence', 'onfidencecay'), ('prescribed', 'escribedpray'), ('scrutinies', 'utiniesscray'), ('melancholy', 'elancholymay'), ('handsomest', 'andsomesthay'), ('prescience', 'esciencepray'), ('conclusion', 'onclusioncay'), ('remarkably', 'emarkablyray'), ('gratefully', 'atefullygray'), ('michaelmas', 'ichaelmasmay'), ('containing', 'ontainingcay'), ('discussion', 'iscussionday'), ('resentment', 'esentmentray'), ('restraints', 'estraintsray'), ('tablecloth', 'ableclothtay'), ('projecting', 'ojectingpray'), ('frightened', 'ightenedfray'), ('pardonable', 'ardonablepay'), ('variations', 'ariationsvay'), ('sensations', 'ensationssay'), ('rheumatism', 'eumatismrhay'), ('dispirited', 'ispiritedday'), ('politeness', 'olitenesspay'), ('suggesting', 'uggestingsay'), ('willoughby', 'illoughbyway'), ('reasonable', 'easonableray'), ('discerning', 'iscerningday'), ('previously', 'eviouslypray'), ('passionate', 'assionatepay'), ('prophecies', 'opheciespray'), ('possessing', 'ossessingpay'), ('perfection', 'erfectionpay'), ('heightened', 'eightenedhay'), ('generously', 'enerouslygay'), ('faithfully', 'aithfullyfay'), ('reconciled', 'econciledray'), ('distressed', 'istressedday'), ('respective', 'espectiveray'), ('liberality', 'iberalitylay'), ('disengaged', 'isengagedday'), ('honourably', 'onourablyhay'), ('persuading', 'ersuadingpay'), ('cultivated', 'ultivatedcay'), ('projection', 'ojectionpray'), ('disapprove', 'isapproveday'), ('conciliate', 'onciliatecay'), ('satisfying', 'atisfyingsay'), ('dispatches', 'ispatchesday'), ('repeatedly', 'epeatedlyray'), ('suspicions', 'uspicionssay'), ('perplexity', 'erplexitypay'), ('strengthen', 'engthenstray'), ('dependence', 'ependenceday'), ('dissipated', 'issipatedday'), ('conformity', 'onformitycay'), ('hysterical', 'ystericalhay'), ('throughout', 'oughoutthray'), ('sacrificed', 'acrificedsay'), ('dimensions', 'imensionsday'), ('flatteries', 'atteriesflay'), ('completely', 'ompletelycay'), ('reproaches', 'eproachesray'), ('contriving', 'ontrivingcay'), ('consequent', 'onsequentcay'), ('liveliness', 'ivelinesslay'), ('resistance', 'esistanceray'), ('caricature', 'aricaturecay'), ('stimulated', 'imulatedstay'), ('brightened', 'ightenedbray'), ('prevailing', 'evailingpray'), ('disclosure', 'isclosureday'), ('bequeathed', 'equeathedbay'), ('disappoint', 'isappointday'), ('complexion', 'omplexioncay'), ('dismounted', 'ismountedday'), ('remembered', 'ememberedray'), ('delineated', 'elineatedday'), ('subsequent', 'ubsequentsay'), ('persevered', 'erseveredpay'), ('doubtingly', 'oubtinglyday'), ('characters', 'aracterschay'), ('neighbours', 'eighboursnay'), ('witticisms', 'itticismsway'), ('materially', 'ateriallymay'), ('playthings', 'aythingsplay'), ('suspecting', 'uspectingsay'), ('conditions', 'onditionscay'), ('discovered', 'iscoveredday'), ('deficiency', 'eficiencyday'), ('regularity', 'egularityray'), ('subjection', 'ubjectionsay'), ('brightness', 'ightnessbray'), ('pleasantly', 'easantlyplay'), ('delicacies', 'elicaciesday'), ('preference', 'eferencepray'), ('difference', 'ifferenceday'), ('compassion', 'ompassioncay'), ('competence', 'ompetencecay'), ('generosity', 'enerositygay'), ('retirement', 'etirementray'), ('strikingly', 'ikinglystray'), ('subsisting', 'ubsistingsay'), ('conscience', 'onsciencecay'), ('diabolical', 'iabolicalday'), ('temptation', 'emptationtay'), ('formidable', 'ormidablefay'), ('reconsider', 'econsiderray'), ('prejudiced', 'ejudicedpray'), ('comprehend', 'omprehendcay'), ('compliment', 'omplimentcay'),

('sufferings', 'ufferingssay'), ('concluding', 'oncludingcay'), ('compliance',
'ompliancecay'), ('mosquitoes', 'osquitoesmay'), ('thoughtful', 'oughtfulthay'),
('bitterness', 'itternessbay'), ('mentioning', 'entioningmay'), ('toleration',
'olerationtay'), ('profession', 'ofessionpray'), ('discompose', 'iscomposeday'),
('waistcoats', 'aistcoatsway'), ('reflecting', 'eflectingray'), ('positively',
'ositivelypay'), ('diminished', 'iminishedday')], (6, 8): [('pretty',
'ettypray'), ('survey', 'urveysay'), ('demure', 'emureday'), ('secure',
'ecuresay'), ('sobbed', 'obbedsay'), ('moment', 'omentmay'), ('guilty',
'uiltygay'), ('combat', 'ombatcay'), ('modern', 'odernmay'), ('father',
'atherfay'), ('horses', 'orseshay'), ('closer', 'oserclay'), ('nicety',
'icetynay'), ('shaken', 'akenshay'), ('centre', 'entrecay'), ('powers',
'owerspay'), ('simple', 'implesay'), ('beings', 'eingsbay'), ('recent',
'ecentray'), ('sketch', 'etchskay'), ('loving', 'ovinglay'), ('dagger',
'aggerday'), ('hearts', 'eartshay'), ('copied', 'opiedcay'), ('gazing',
'azinggay'), ('sooner', 'oonersay'), ('disarm', 'isarmday'), ('suited',
'uitedsay'), ('doubts', 'oubtsday'), ('lifted', 'iftedlay'), ('vacant',
'acantvay'), ('palmer', 'almerpay'), ('grudge', 'udgegray'), ('folded',
'oldedfay'), ('remove', 'emoveray'), ('borrow', 'orrowbay'), ('scheme',
'emeschay'), ('solemn', 'olemnsay'), ('denote', 'enoteday'), ('bustle',
'ustlebay'), ('turned', 'urnedtay'), ('mutton', 'uttonmay'), ('tender',
'endertay'), ('hearer', 'earerhay'), ('begins', 'eginsbay'), ('taught',
'aughttay'), ('honour', 'onourhay'), ('robert', 'obertray'), ('gibson',
'ibsongay'), ('virtue', 'irtuevay'), ('vanish', 'anishvay'), ('writes',
'iteswray'), ('silent', 'ilentsay'), ('singly', 'inglysay'), ('jargon',
'argonjay'), ('grieve', 'ievegray'), ('design', 'esignday'), ('linger',
'ingerlay'), ('bathed', 'athedbay'), ('pardon', 'ardonpay'), ('rumour',
'umourray'), ('dances', 'ancesday'), ('remark', 'emarkray'), ('worded',
'ordedway'), ('honest', 'onesthay'), ('chosen', 'osenchay'), ('boyish',
'oyishbay'), ('stress', 'essstray'), ('caught', 'aughtcay'), ('potent',
'otentpay'), ('during', 'uringday'), ('stocks', 'ocksstay'), ('relics',
'elicsray'), ('tongue', 'onguetay'), ('street', 'eetstray'), ('whiled',
'iledwhay'), ('manner', 'annermay'), ('truths', 'uthstray'), ('travel',
'aveltray'), ('couple', 'ouplecay'), ('gaiety', 'aietygay'), ('giving',
'ivinggay'), ('county', 'ountycay'), ('cramps', 'ampscray'), ('wealth',
'ealthway'), ('divine', 'ivineday'), ('behold', 'eholdbay'), ('feared',
'earedfay'), ('sparks', 'arksspay'), ('finger', 'ingerfay'), ('talent',
'alenttay'), ('demand', 'emandday'), ('coffee', 'offeecay'), ('merest',
'erestmay'), ('detail', 'etailday'), ('casino', 'asinocay'), ('choose',
'oosechay'), ('warmed', 'armedway'), ('relate', 'elateray'), ('sashes',
'ashessay'), ('chance', 'ancechay'), ('rushed', 'ushedray'), ('health',
'ealthhay'), ('preyed', 'eyedpray'), ('mutual', 'utualmay'), ('months',
'onthsmay'), ('raises', 'aisesray'), ('things', 'ingsthay'), ('master',
'astermay'), ('thanks', 'anksthay'), ('cheeks', 'eekschay'), ('hazard',
'azardhay'), ('slight', 'ightslay'), ('guests', 'uestsgay'), ('halves',
'alveshay'), ('lessen', 'essenlay'), ('notice', 'oticenay'), ('higher',
'igherhay'), ('truest', 'uesttray'), ('porter', 'orterpay'), ('fickle',
'icklefay'), ('behave', 'ehavebay'), ('credit', 'editcray'), ('number',
'umbernay'), ('mother', 'othermay'), ('tricks', 'ickstray'), ('traced',

```
'acedtray'), ('subdue', 'ubduesay'), ('ruined', 'uinedray'), ('famous',
'amousfay'), ('danced', 'ancedday'), ('clever', 'everclay'), ('drives',
'ivesdray'), ('resist', 'esistray'), ('forget', 'orgetfay'), ('stable',
'ablestay'), ('bigger', 'iggerbay'), ('betray', 'etraybay'), ('senses',
'ensessay'), ('smiled', 'iledsmay'), ('hating', 'atinghay'), ('bolder',
'olderbay'), ('banish', 'anishbay'), ('remain', 'emainray'), ('period',
'eriodpay'), ('gossip', 'ossipgay'), ('basket', 'asketbay'), ('genius',
'eniusgay'), ('seized', 'eizedsay'), ('widely', 'idelyway'), ('pursue',
'ursuepay'), ('favour', 'avourfay'), ('sighed', 'ighedsay'), ('plants',
'antsplay'), ('proved', 'ovedpray'), ('season', 'easonsay'), ('circle',
'irclecay'), ('taking', 'akingtay'), ('screen', 'eenscray'), ('talker',
'alkertay'), ('teazed', 'eazedtay'), ('suffer', 'uffersay'), ('curate',
'uratecay'), ('traits', 'aitstray'), ('defect', 'efectday'), ('called',
'alledcay'), ('dining', 'iningday'), ('direct', 'irectday'), ('dinner',
'innerday'), ('joking', 'okingjay'), ('whiter', 'iterwhay'), ('limits',
'imitslay'), ('papers', 'aperspay'), ('public', 'ublicpay'), ('losing',
'osinglay'), ('handed', 'andedhay'), ('wisest', 'isestway'), ('delays',
'elaysday'), ('leaves', 'eaveslay'), ('points', 'ointspay'), ('hither',
'itherhay'), ('horrid', 'orridhay'), ('middle', 'iddlemay'), ('vastly',
'astlyvay'), ('luxury', 'uxurylay'), ('failed', 'ailedfay'), ('lodges',
'odgeslay'), ('female', 'emalefay'), ('walked', 'alkedway'), ('nipped',
'ippednay'), ('nephew', 'ephewnay'), ('settle', 'ettlesay'), ('kinder',
'inderkay'), ('bowing', 'owingbay'), ('smooth', 'oothsmay'), ('roared',
'oaredray'), ('wiping', 'ipingway'), ('jewels', 'ewelsjay'), ('breath',
'eathbray'), ('steady', 'eadystay'), ('breach', 'eachbray'), ('polite',
'olitepay'), ('colder', 'oldercay'), ('though', 'oughthay'), ('poured',
'ouredpay'), ('paused', 'ausedpay'), ('nought', 'oughtnay'), ('shrewd',
'ewdshray'), ('fourth', 'ourthfay'), ('parish', 'arishpay'), ('traded',
'adedtray'), ('glance', 'anceglay'), ('lesser', 'esserlay'), ('curacy',
'uracycay'), ('labour', 'abourlay'), ('became', 'ecamebay'), ('decide',
'ecideday'), ('kisses', 'isseskay'), ('styled', 'yledstay'), ('sprung',
'ungspray'), ('priory', 'iorypray'), ('derive', 'eriveday'), ('school',
'oolschay'), ('liking', 'ikinglay'), ('dawned', 'awnedday'), ('weaken',
'eakenway'), ('winter', 'interway'), ('rivals', 'ivalsray'), ('sallow',
'allowsay'), ('defend', 'efendday'), ('seated', 'eatedsay'), ('penury',
'enurypay'), ('repair', 'epairray'), ('chaise', 'aisechay'), ('church',
'urchchay'), ('beyond', 'eyondbay'), ('wisdom', 'isdomway'), ('retain',
'etainray'), ('toilet', 'oilettay'), ('talked', 'alkedtay'), ('dearly',
'earlyday'), ('reward', 'ewardray'), ('safely', 'afelysay'), ('making',
'akingmay'), ('debate', 'ebateday'), ('future', 'uturefay'), ('broken',
'okenbray'), ('begged', 'eggedbay'), ('thirty', 'irtythay'), ('muslin',
'uslinmay'), ('marked', 'arkedmay'), ('hinder', 'inderhay'), ('beheld',
'eheldbay'), ('maxims', 'aximsmay'), ('theirs', 'eirsthay'), ('corner',
'ornercay'), ('piqued', 'iquedpay'), ('rested', 'estedray'), ('needed',
'eedednay'), ('fitted', 'ittedfay'), ('candid', 'andidcay'), ('laughs',
'aughslay'), ('window', 'indowway'), ('tastes', 'astestay'), ('spread',
'eadspray'), ('hollow', 'ollowhay'), ('sorrow', 'orrowsay'), ('partly',
'artlypay'), ('served', 'ervedsay'), ('flowed', 'owedflay'), ('remedy',
```

```
'emedyray'), ('trying', 'yingtray'), ('latest', 'atestlay'), ('manage',
'anagemay'), ('gained', 'ainedgay'), ('plenty', 'entyplay'), ('garden',
'ardengay'), ('frosts', 'ostsfray'), ('thirds', 'irdsthay'), ('handle',
'andlehay'), ('verbal', 'erbalvay'), ('sheath', 'eathshay'), ('covert',
'overtcay'), ('darker', 'arkerday'), ('harris', 'arrishay'), ('houses',
'ouseshay'), ('nature', 'aturenay'), ('thomas', 'omasthay'), ('submit',
'ubmitsay'), ('danger', 'angerday'), ('lawyer', 'awyerlay'), ('matter',
'attermay'), ('rugged', 'uggedray'), ('highly', 'ighlyhay'), ('whence',
'encewhay'), ('martha', 'arthamay'), ('hearty', 'eartyhay'), ('recall',
'ecallray'), ('raised', 'aisedray'), ('follow', 'ollowfay'), ('nearer',
'earernay'), ('gentle', 'entlegay'), ('pitied', 'itiedpay'), ('nurses',
'ursesnay'), ('punish', 'unishpay')], (7, 9): [('parcels', 'arcelspay'),
('welcome', 'elcomeway'), ('posture', 'osturepay'), ('mothers', 'othersmay'),
('tempers', 'emperstay'), ('carries', 'arriescay'), ('neglect', 'eglectnay'),
('prevail', 'evailpray'), ('declare', 'eclareday'), ('crowned', 'ownedcray'),
('mindful', 'indfulmay'), ('blessed', 'essedblay'), ('kitchen', 'itchenkay'),
('younger', 'youngeray'), ('hanover', 'anoverhay'), ('lowered', 'oweredlay'),
('summons', 'ummonssay'), ('suffice', 'ufficesay'), ('rousing', 'ousingray'),
('feeling', 'eelingfay'), ('parlors', 'arlorspay'), ('gravely', 'avelygray'),
('connect', 'onnectcay'), ('species', 'eciesspay'), ('revived', 'evivedray'),
('tribute', 'ibutetray'), ('pitying', 'ityingpay'), ('brother', 'otherbray'),
('retired', 'etiredray'), ('suffers', 'ufferssay'), ('spurned', 'urnedspay'),
('dropped', 'oppeddray'), ('galling', 'allinggay'), ('seducer', 'educersay'),
('foibles', 'oiblesfay'), ('tuesday', 'uesdaytay'), ('patient', 'atientpay'),
('worsted', 'orstedway'), ('trifled', 'ifledtray'), ('waiting', 'aitingway'),
('widower', 'idowerway'), ('hardily', 'ardilyhay'), ('glimpse', 'impseglay'),
('whether', 'etherwhay'), ('reverie', 'everieray'), ('resolve', 'esolveray'),
('propose', 'oposepray'), ('tallest', 'allesttay'), ('nominal', 'ominalnay'),
('languid', 'anguidlay'), ('wronged', 'ongedwray'), ('nothing', 'othingnay'),
('promote', 'omotepray'), ('novelty', 'oveltynay'), ('dooming', 'oomingday'),
('display', 'isplayday'), ('correct', 'orrectcay'), ('feigned', 'eignedfay'),
('schemes', 'emesschay'), ('deepest', 'eepestday'), ('flannel', 'annelflay'),
('hauteur', 'auteurhay'), ('sharing', 'aringshay'), ('getting', 'ettinggay'),
('screams', 'eamsscray'), ('variety', 'arietyvay'), ('yielded', 'yieldeday'),
('brought', 'oughtbray'), ('haughty', 'aughtyhay'), ('bristol', 'istolbray'),
('silence', 'ilencesay'), ('staying', 'ayingstay'), ('confess', 'onfesscay'),
('placing', 'acingplay'), ('noisier', 'oisiernay'), ('nursing', 'ursingnay'),
('tuition', 'uitiontay'), ('welfare', 'elfareway'), ('hunters', 'untershay'),
('poplars', 'oplarspay'), ('deprive', 'epriveday'), ('mansion', 'ansionmay'),
('reasons', 'easonsray'), ('conceit', 'onceitcay'), ('burgess', 'urgessbay'),
('receipt', 'eceiptray'), ('befalls', 'efallsbay'), ('fetches', 'etchesfay'),
('ranging', 'angingray'), ('passion', 'assionpay'), ('smirked', 'irkedsmay'),
('honours', 'onourshay'), ('covered', 'overedcay'), ('dancing', 'ancingday'),
('hundred', 'undredhay'), ('forgave', 'orgavefay'), ('fifteen', 'ifteenfay'),
('lasting', 'astinglay'), ('minuter', 'inutermay'), ('tremour', 'emourtray'),
('passing', 'assingpay'), ('streets', 'eetsstray'), ('resists', 'esistsray'),
('cleared', 'earedclay'), ('miracle', 'iraclemay'), ('several', 'everalsay'),
('sighing', 'ighingsay'), ('resumed', 'esumedray'), ('merited', 'eritedmay'),
```

```
('without', 'ithoutway'), ('removed', 'emovedray'), ('behaved', 'ehavedbay'),
('succour', 'uccoursay'), ('bribing', 'ibingbray'), ('charity', 'aritychay'),
('shortly', 'ortlyshay'), ('tempted', 'emptedtay'), ('briefly', 'ieflybray'),
('parties', 'artiespay'), ('touched', 'ouchedtay'), ('content', 'ontentcay'),
('proceed', 'oceedpray'), ('windows', 'indowsway'), ('ribbons', 'ibbonsray'),
('stepped', 'eppedstay'), ('pursuit', 'ursuitpay'), ('quarter', 'arterquay'),
('prudent', 'udentpray'), ('dignity', 'ignityday'), ('payment', 'aymentpay'),
('grounds', 'oundsgray'), ('bequest', 'equestbay'), ('donavan', 'onavanday'),
('renewed', 'enewedray'), ('smaller', 'allersmay'), ('fearful', 'earfulfay'),
('civilly', 'ivillycay'), ('romance', 'omanceray'), ('females', 'emalesfay'),
('grieved', 'ievedgray'), ('created', 'eatedcray'), ('command', 'ommandcay'),
('defined', 'efinedday'), ('bluntly', 'untlyblay'), ('recital', 'ecitalray'),
('sorrows', 'orrowssay'), ('whither', 'itherwhay'), ('spotted', 'ottedspay'),
('justify', 'ustifyjay'), ('conduit', 'onduitcay'), ('revenge', 'evengeray'),
('scenery', 'eneryscay'), ('started', 'artedstay'), ('degrees', 'egreesday'),
('whisper', 'isperwhay'), ('strange', 'angestray'), ('bidding', 'iddingbay'),
('musical', 'usicalmay'), ('happens', 'appenshay'), ('compare', 'omparecay'),
('presume', 'esumepray'), ('sitting', 'ittingsay'), ('regards', 'egardsray'),
('violins', 'iolinsvay'), ('rightly', 'ightlyray'), ('memento', 'ementomay'),
('courted', 'ourtedcay'), ('servant', 'ervantsay'), ('possess', 'ossesspay'),
('herself', 'erselfhay'), ('present', 'esentpray'), ('natural', 'aturalnay'),
('managed', 'anagedmay'), ('hostess', 'ostesshay'), ('leaving', 'eavinglay'),
('mention', 'entionmay'), ('circuit', 'ircuitcay'), ('counsel', 'ounselcay'),
('guineas', 'uineasgay'), ('solidly', 'olidlysay'), ('dashing', 'ashingday'),
('station', 'ationstay'), ('written', 'ittenwray'), ('writing', 'itingwray'),
('country', 'ountrycay'), ('delight', 'elightday'), ('westons', 'estonsway'),
('debated', 'ebatedday'), ('lengths', 'engthslay'), ('torment', 'ormenttay'),
('reverse', 'everseray'), ('bedroom', 'edroombay'), ('forbade', 'orbadefay'),
('comings', 'omingscay'), ('tonight', 'onighttay'), ('similar', 'imilarsay'),
('calling', 'allingcay'), ('cassino', 'assinocay'), ('strains', 'ainsstray'),
('scratch', 'atchscray'), ('gardens', 'ardensgay'), ('freedom', 'eedomfray'),
('compact', 'ompactcay'), ('message', 'essagemay'), ('heavily', 'eavilyhay'),
('manners', 'annersmay'), ('systems', 'ystemssay'), ('forbore', 'orborefay'),
('wrapped', 'appedwray'), ('defects', 'efectsday'), ('villain', 'illainvay'),
('studied', 'udiedstay'), ('playing', 'ayingplay'), ('diction', 'ictionday'),
('tending', 'endingtay'), ('seeking', 'eekingsay'), ('rapture', 'aptureray'),
('dawdled', 'awdledday'), ('wearing', 'earingway'), ('comfort', 'omfortcay'),
('finally', 'inallyfay'), ('circles', 'irclescay'), ('putting', 'uttingpay'),
('twisted', 'istedtway'), ('destroy', 'estroyday'), ('whereas', 'ereaswhay'),
('dismiss', 'ismissday'), ('visibly', 'isiblyvay'), ('talents', 'alentstay'),
('finding', 'indingfay'), ('removes', 'emovesray'), ('replied', 'epliedray'),
('bluster', 'usterblay'), ('raising', 'aisingray'), ('plainly', 'ainlyplay'),
('worried', 'orriedway'), ('meaning', 'eaningmay'), ('blushes', 'ushesblay'),
('relapse', 'elapseray'), ('careful', 'arefulcay'), ('beloved', 'elovedbay'),
('degrade', 'egradeday'), ('foresaw', 'oresawfay'), ('procure', 'ocurepray'),
('kindest', 'indestkay'), ('secures', 'ecuressay'), ('hanging', 'anginghay'),
('persons', 'ersonspay'), ('taverns', 'avernstay'), ('walking', 'alkingway'),
('quicker', 'ickerquay'), ('guessed', 'uessedgay'), ('confirm', 'onfirmcay'),
```

('venture', 'enturevay'), ('project', 'ojectpray'), ('prayers', 'ayerspray'),
('heights', 'eightshay'), ('subject', 'ubjectsay'), ('summits', 'ummitssay'),
('towards', 'owardstay'), ('closing', 'osingclay'), ('journey', 'ourneyjay'),
('pointer', 'ointerpay'), ('chained', 'ainedchay'), ('clothes', 'othesclay'),
('require', 'equireray'), ('chusing', 'usingchay'), ('reproof', 'eproofray'),
('reckons', 'eckonsray'), ('pressed', 'essedpray'), ('talking', 'alkingtay'),
('harbour', 'arbourhay'), ('garrets', 'arretsgay'), ('foresee', 'oreseefay'),
('lustily', 'ustilylay'), ('leagued', 'eaguedlay'), ('judging', 'udgingjay'),
('rejoice', 'ejoiceray'), ('honiton', 'onitonhay'), ('shorten', 'ortenshay'),
('stating', 'atingstay'), ('privacy', 'ivacypray'), ('steeles', 'eelesstay'),
('thickly', 'icklythay'), ('shocked', 'ockedshay'), ('limited', 'imitedlay'),
('blights', 'ightsblay'), ('seclude', 'ecludesay'), ('cordial', 'ordialcay'),
('request', 'equestray'), ('denoted', 'enotedday'), ('darling', 'arlingday'),
('coxcomb', 'oxcombcay'), ('however', 'oweverhay'), ('perform', 'erformpay'),
('society', 'ocietysay'), ('deserts', 'esertsday'), ('palmers', 'almerspay'),
('manager', 'anagermay'), ('rupture', 'uptureray'), ('concise', 'oncisecay'),
('scolded', 'oldedscay'), ('seduced', 'educedsay'), ('doubted', 'oubtedday'),
('forward', 'orwardfay'), ('leisure', 'eisurelay'), ('robbing', 'obbingray'),
('horizon', 'orizonhay'), ('blunder', 'underblay'), ('flushed', 'ushedflay'),
('divided', 'ividedday'), ('charged', 'argedchay'), ('disturb', 'isturbday'),
('smitten', 'ittensmay'), ('seconds', 'econdssay'), ('letters', 'etterslay'),
('provide', 'ovidepray'), ('conduct', 'onductcay'), ('plagued', 'aguedplay'),
('stretch', 'etchstray'), ('hoarded', 'oardedhay'), ('blushed', 'ushedblay'),
('nodding', 'oddingnay'), ('mankind', 'ankindmay'), ('caprice', 'apricecay'),
('quitted', 'ittedquay')], (5, 7): [('china', 'inachay'), ('deter', 'eterday'),
('woman', 'omanway'), ('clock', 'ockclay'), ('borne', 'ornebay'), ('noise',
'oisenay'), ('midst', 'idstmay'), ('dwell', 'elldway'), ('smith', 'ithsmay'),
('gowns', 'ownsgay'), ('named', 'amednay'), ('glass', 'assglay'), ('price',
'icepray'), ('close', 'oseclay'), ('guide', 'uidegay'), ('newly', 'ewlynay'),
('sweep', 'eepsway'), ('swept', 'eptsway'), ('those', 'osethay'), ('lower',
'owerlay'), ('plump', 'umpplay'), ('downs', 'ownsday'), ('purse', 'ursepay'),
('linen', 'inenlay'), ('later', 'aterlay'), ('broad', 'oadbray'), ('miles',
'ilesmay'), ('facts', 'actsfay'), ('sleep', 'eepslay'), ('steps', 'epsstay'),
('wants', 'antsway'), ('forth', 'orthfay'), ('drove', 'ovedray'), ('mouth',
'outhmay'), ('worry', 'orryway'), ('dried', 'ieddray'), ('stood', 'oodstay'),
('brush', 'ushbray'), ('sting', 'ingstay'), ('mirth', 'irthmay'), ('heads',
'eadshay'), ('sight', 'ightsay'), ('whose', 'osewhay'), ('drawn', 'awndray'),
('score', 'orescay'), ('party', 'artypay'), ('tears', 'earstay'), ('heart',
'earthay'), ('words', 'ordsway'), ('hunts', 'untshay'), ('trace', 'acetray'),
('trial', 'ialtray'), ('swoon', 'oonsway'), ('knees', 'eesknay'), ('bring',
'ingbray'), ('solid', 'olidsay'), ('sharp', 'arpshay'), ('basis', 'asisbay'),
('cavil', 'avilcay'), ('rated', 'atedray'), ('grave', 'avegray'), ('serve',
'ervesay'), ('betty', 'ettybay'), ('judge', 'udgejay'), ('boast', 'oastbay'),
('march', 'archmay'), ('built', 'uiltbay'), ('clear', 'earclay'), ('happy',
'appyhay'), ('fruit', 'uitfray'), ('witty', 'ittyway'), ('total', 'otaltay'),
('dared', 'aredday'), ('defer', 'eferday'), ('tried', 'iedtray'), ('vexed',
'exedvay'), ('walls', 'allsway'), ('liked', 'ikedlay'), ('prove', 'ovepray'),
('sneer', 'eersnay'), ('merit', 'eritmay'), ('scott', 'ottscay'), ('madam',

'adammay'), ('truth', 'uthtray'), ('minor', 'inormay'), ('stare', 'arestay'), ('calls', 'allscay'), ('lucky', 'uckylay'), ('hence', 'encehay'), ('press', 'esspray'), ('comes', 'omescay'), ('dined', 'inedday'), ('think', 'inkthay'), ('plate', 'ateplay'), ('grief', 'iefgray'), ('names', 'amesnay'), ('stock', 'ockstay'), ('short', 'ortshay'), ('moved', 'ovedmay'), ('teach', 'eachtay'), ('fears', 'earsfay'), ('tiled', 'iledtay'), ('cried', 'iedcray'), ('mound', 'oundmay'), ('small', 'allsmay'), ('twice', 'icetway'), ('lines', 'ineslay'), ('looks', 'ookslay'), ('rival', 'ivalray'), ('first', 'irstfay'), ('wound', 'oundway'), ('saves', 'avessay'), ('nurse', 'ursenay'), ('treat', 'eattray'), ('could', 'ouldcay'), ('rider', 'iderray'), ('lobby', 'obbylay'), ('table', 'abletay'), ('touch', 'ouchtay'), ('topic', 'opictay'), ('stand', 'andstay'), ('leant', 'eantlay'), ('grass', 'assgray'), ('check', 'eckchay'), ('quite', 'itequay'), ('wives', 'ivesway'), ('walks', 'alksway'), ('labor', 'aborlay'), ('house', 'ousehay'), ('drops', 'opsdray'), ('loose', 'ooselay'), ('tempt', 'empttay'), ('place', 'aceplay'), ('local', 'ocallay'), ('seats', 'eatssay'), ('gates', 'atesgay'), ('dropt', 'optdray'), ('cares', 'arescay'), ('falls', 'allsfay'), ('speak', 'eakspay'), ('shops', 'opsshay'), ('groom', 'oomgray'), ('sugar', 'ugarsay'), ('front', 'ontfray'), ('visit', 'isitvay'), ('raise', 'aiseray'), ('fetch', 'etchfay'), ('silks', 'ilkssay'), ('beset', 'esetbay'), ('rocks', 'ocksray'), ('start', 'artstay'), ('winks', 'inksway'), ('sends', 'endssay'), ('piano', 'ianopay'), ('fully', 'ullyfay'), ('widow', 'idowway'), ('wiser', 'iserway'), ('dirty', 'irtyday'), ('below', 'elowbay'), ('manor', 'anormay'), ('scale', 'alescay'), ('whole', 'olewhay'), ('minds', 'indsmay'), ('glebe', 'ebeglay'), ('fifty', 'iftyfay'), ('years', 'yearsay'), ('story', 'orystay'), ('fresh', 'eshfray'), ('begun', 'egunbay'), ('dance', 'anceday'), ('jokes', 'okesjay'), ('turns', 'urnstay'), ('honor', 'onorhay'), ('spite', 'itespay'), ('green', 'eengray'), ('saint', 'aintsay'), ('hated', 'atedhay'), ('doors', 'oorsday'), ('shown', 'ownshay'), ('power', 'owerpay'), ('today', 'odaytay'), ('floor', 'oorflay'), ('force', 'orcefay'), ('right', 'ightray'), ('trick', 'icktray'), ('dread', 'eaddray'), ('final', 'inalfay'), ('shake', 'akeshay'), ('lamps', 'ampslay'), ('dying', 'yingday'), ('black', 'ackblay'), ('grand', 'andgray'), ('field', 'ieldfay'), ('plead', 'eadplay'), ('loved', 'ovedlay'), ('terms', 'ermstay'), ('spend', 'endspay'), ('mohrs', 'ohrsmay'), ('young', 'youngay'), ('great', 'eatgray'), ('guess', 'uessgay'), ('ninny', 'innynay'), ('hasty', 'astyhay'), ('nerve', 'ervenay'), ('meals', 'ealsmay'), ('three', 'eethray'), ('shade', 'adeshay'), ('store', 'orestay'), ('daily', 'ailyday'), ('taste', 'astetay'), ('gross', 'ossgray'), ('dares', 'aresday'), ('grate', 'ategray'), ('stuff', 'uffstay'), ('sheet', 'eetshay')], (14, 18): [('breakfast-room', 'eakfastbray-oomray'), ('lavender-water', 'avenderlay-aterway'), ('broken-hearted', 'okenbray-eartedhay'), ('spunging-house', 'ungingspay-ousehay')], (9, 12): [('unwelcome', 'unwelcomeway'), ('exhausted', 'exhaustedway'), ('intrusion', 'intrusionway'), ('untouched', 'untouchedway'), ('interview', 'interviewway'), ('effusions', 'effusionsway'), ('impartial', 'impartialway'), ('inventive', 'inventiveway'), ('emergency', 'emergencyway'), ('officious', 'officiousway'), ('everybody', 'everybodyway'), ('observant', 'observantway'), ('emptiness', 'emptinessway'), ('occasions', 'occasionsway'), ('assembled', 'assembledway'), ('admirable', 'admirableway'), ('uselessly', 'uselesslyway'), ('excellent', 'excellentway'), ('associate', 'associateway'),

('affecting', 'affectingway'), ('absurdity', 'absurdityway'), ('allowable', 'allowableway'), ('appointed', 'appointedway'), ('overheard', 'overheardway'), ('improving', 'improvingway'), ('eagerness', 'eagernessway'), ('unluckily', 'unluckilyway'), ('allowance', 'allowanceway'), ('estimable', 'estimableway'), ('obviating', 'obviatingway'), ('entrusted', 'entrustedway'), ('arranging', 'arrangingway'), ('exclaimed', 'exclaimedway'), ('ingenuity', 'ingenuityway'), ('outwardly', 'outwardlyway'), ('expensive', 'expensiveway'), ('entreated', 'entreatedway'), ('unconcern', 'unconcernway'), ('obtaining', 'obtainingway'), ('interests', 'interestsway'), ('assiduous', 'assiduousway'), ('adjusting', 'adjustingway'), ('ungenteel', 'ungenteelway'), ('encourage', 'encourageway'), ('apartment', 'apartmentway'), ('inheritor', 'inheritorway'), ('unceasing', 'unceasingway'), ('insulting', 'insultingway'), ('unchanged', 'unchangedway'), ('animation', 'animationway'), ('enclosure', 'enclosureway'), ('apologize', 'apologizeway'), ('earnestly', 'earnestlyway'), ('exquisite', 'exquisiteway'), ('attending', 'attendingway'), ('invention', 'inventionway'), ('enquiries', 'enquiriesway'), ('appetites', 'appetitesway'), ('additions', 'additionsway'), ('impudence', 'impudenceway'), ('insincere', 'insincereway'), ('abundance', 'abundanceway'), ('exchanged', 'exchangedway'), ('extremely', 'extremelyway'), ('inflicted', 'inflictedway'), ('attracted', 'attractedway'), ('inelegant', 'inelegantway'), ('undoubted', 'undoubtedway'), ('etiquette', 'etiquetteway'), ('unfeigned', 'unfeignedway'), ('agreeable', 'agreeableway'), ('unsubdued', 'unsubduedway'), ('inspiring', 'inspiringway'), ('unguarded', 'unguardedway'), ('advancing', 'advancingway'), ('undressed', 'undressedway'), ('excursive', 'excursiveway'), ('extending', 'extendingway'), ('extolling', 'extollingway'), ('offending', 'offendingway'), ('eloquence', 'eloquenceway'), ('ornaments', 'ornamentsway'), ('intruding', 'intrudingway'), ('incessant', 'incessantway'), ('unwilling', 'unwillingway'), ('annamaria', 'annamariaway'), ('unbounded', 'unboundedway'), ('incurable', 'incurableway'), ('unnoticed', 'unnoticedway'), ('expressly', 'expresslyway'), ('impetuous', 'impetuousway'), ('appearing', 'appearingway'), ('ingenious', 'ingeniousway'), ('amazement', 'amazementway'), ('animating', 'animatingway'), ('interfere', 'interfereway'), ('ascending', 'ascendingway'), ('affection', 'affectionway'), ('undergone', 'undergoneway'), ('unstudied', 'unstudiedway'), ('attempted', 'attemptedway'), ('infirmity', 'infirmityway'), ('apologies', 'apologiesway'), ('elsewhere', 'elsewhereway'), ('expecting', 'expectingway'), ('informing', 'informingway'), ('admission', 'admissionway'), ('extremity', 'extremityway')], (6, 9): [('unread', 'unreadway'), ('owning', 'owningway'), ('odious', 'odiousway'), ('either', 'eitherway'), ('insult', 'insultway'), ('awhile', 'awhileway'), ('assist', 'assistway'), ('injure', 'injureway'), ('eluded', 'eludedway'), ('others', 'othersway'), ('untied', 'untiedway'), ('excess', 'excessway'), ('urgent', 'urgentway'), ('appear', 'appearway'), ('aspect', 'aspectway'), ('issued', 'issuedway'), ('evince', 'evinceway'), ('eating', 'eatingway'), ('easter', 'easterway'), ('arisen', 'arisenway'), ('earned', 'earnedway'), ('esteem', 'esteemway'), ('accuse', 'accuseway'), ('owners', 'ownersway'), ('events', 'eventsway'), ('energy', 'energyway'), ('affair', 'affairway'), ('effect', 'effectway'), ('unlike', 'unlikeway'), ('unison', 'unisonway'), ('enough', 'enoughway'), ('invite', 'inviteway'), ('income', 'incomeway'), ('extent', 'extentway'), ('office', 'officeway'), ('ankles', 'anklesway'), ('admire',

```
'admireway'), ('induce', 'induceway'), ('infant', 'infantway'), ('expand',
'expandway'), ('orders', 'ordersway'), ('occupy', 'occupyway'), ('attend',
'attendway'), ('unable', 'unableway'), ('olives', 'olivesway'), ('awaken',
'awakenway'), ('engage', 'engageway'), ('eldest', 'eldestway'), ('eyeing',
'eyeingway'), ('advice', 'adviceway'), ('always', 'alwaysway'), ('insist',
'insistway'), ('accept', 'acceptway'), ('indies', 'indiesway'), ('excite',
'exciteway'), ('openly', 'openlyway'), ('oblige', 'obligeway'), ('amidst',
'amidstway'), ('across', 'acrossway'), ('afraid', 'afraidway'), ('uproar',
'uproarway')], (8, 10): [('villages', 'illagesvay'), ('doubtful', 'oubtfulday'),
('deceived', 'eceivedday'), ('foreseen', 'oreseenfay'), ('dennison',
'ennisonday'), ('steadily', 'eadilystay'), ('startled', 'artledstay'),
('drawings', 'awingsdray'), ('rejected', 'ejectedray'), ('deserves',
'eservesday'), ('curricle', 'urriclecay'), ('domestic', 'omesticday'),
('believed', 'elievedbay'), ('packages', 'ackagespay'), ('kneeling',
'eelingknay'), ('degraded', 'egradedday'), ('medicine', 'edicinemay'),
('promoted', 'omotedpray'), ('wondered', 'onderedway'), ('reproach',
'eproachray'), ('somerset', 'omersetsay'), ('contents', 'ontentscay'),
('children', 'ildrenchay'), ('politics', 'oliticspay'), ('hastened',
'astenedhay'), ('relative', 'elativeray'), ('henshawe', 'enshawehay'),
('material', 'aterialmay'), ('welcomed', 'elcomedway'), ('weddings',
'eddingsway'), ('diverted', 'ivertedday'), ('patience', 'atiencepay'),
('delivery', 'eliveryday'), ('hazarded', 'azardedhay'), ('happiest',
'appiesthay'), ('trifling', 'iflingtray'), ('casement', 'asementcay'),
('reformed', 'eformedray'), ('demanded', 'emandedday'), ('ceremony',
'eremonycay'), ('vigorous', 'igorousvay'), ('mornings', 'orningsmay'),
('hilarity', 'ilarityhay'), ('daughter', 'aughterday'), ('vexation',
'exationvay'), ('shocking', 'ockingshay'), ('previous', 'eviouspray'),
('pleasant', 'easantplay'), ('candidly', 'andidlycay'), ('cheerful',
'eerfulchay'), ('festival', 'estivalfay'), ('changing', 'angingchay'),
('conveyed', 'onveyedcay'), ('lombardy', 'ombardylay'), ('militate',
'ilitatemay'), ('humanity', 'umanityhay'), ('giggling', 'igglinggay'),
('standard', 'andardstay'), ('returned', 'eturnedray'), ('proclaim',
'oclaimpray'), ('ridicule', 'idiculeray'), ('pointers', 'ointerspay'),
('grounded', 'oundedgray'), ('contrast', 'ontrastcay'), ('feelings',
'eelingsfay'), ('mistakes', 'istakesmay'), ('prepared', 'eparedpray'),
('stranger', 'angerstray'), ('habitual', 'abitualhay'), ('firmness',
'irmnessfay'), ('rational', 'ationalray'), ('regulate', 'egulateray'),
('claiming', 'aimingclay'), ('knocking', 'ockingknay'), ('servants',
'ervantssay'), ('goodness', 'oodnessgay'), ('mutually', 'utuallymay'),
('legacies', 'egacieslay'), ('sisterly', 'isterlysay'), ('punctual',
'unctualpay'), ('deciding', 'ecidingday'), ('suitable', 'uitablesay'),
('guarding', 'uardinggay'), ('converse', 'onversecay'), ('slighter',
'ighterslay'), ('dissuade', 'issuadeday'), ('despatch', 'espatchday'),
('blackest', 'ackestblay'), ('recalled', 'ecalledray'), ('loitered',
'oiteredlay'), ('dreading', 'eadingdray'), ('relieved', 'elievedray'),
('westward', 'estwardway'), ('thistles', 'istlesthay'), ('defended',
'efendedday'), ('contempt', 'ontemptcay'), ('sensibly', 'ensiblysay'),
('romantic', 'omanticray'), ('comforts', 'omfortscay'), ('jealousy',
```

'ealousyjay'), ('repeated', 'epeatedray'), ('banditti', 'andittibay'),
('greatest', 'eatestgray'), ('mistrust', 'istrustmay'), ('forwards',
'orwardsfay'), ('rewarded', 'ewardedray'), ('watching', 'atchingway'),
('negative', 'egativenay'), ('sympathy', 'ympathysay'), ('helpless',
'elplesshay'), ('quickest', 'ickestquay'), ('portrait', 'ortraitpay'),
('consoled', 'onsoledcay'), ('fourteen', 'ourteenfay'), ('retarded',
'etardedray'), ('comments', 'ommentscay'), ('jennings', 'enningsjay'),
('securing', 'ecuringsay'), ('withdrew', 'ithdrewway'), ('youthful',
'youthfulay'), ('bachelor', 'achelorbay'), ('convince', 'onvincecay'),
('measured', 'easuredmay'), ('goodwill', 'oodwillgay'), ('branched',
'anchedbray'), ('purchase', 'urchasepay'), ('covenant', 'ovenantcay'),
('reseated', 'eseatedray'), ('concerns', 'oncernscay'), ('hardship',
'ardshiphay'), ('lessened', 'essenedlay'), ('delaford', 'elafordday'),
('governed', 'overnedgay'), ('listened', 'istenedlay'), ('constant',
'onstantcay'), ('fatigued', 'atiguedfay'), ('persuade', 'ersuadepay'),
('critique', 'itiquecray'), ('remedies', 'emediesray'), ('stepping',
'eppingstay'), ('kindness', 'indnesskay'), ('sociable', 'ociablesay'),
('meetings', 'eetingsmay'), ('deserved', 'eservedday'), ('thoughts',
'oughtsthay'), ('becoming', 'ecomingbay'), ('heartily', 'eartilyhay'),
('promised', 'omisedpray'), ('shutting', 'uttingshay'), ('reproved',
'eprovedray'), ('ladyship', 'adyshiplay'), ('pursuing', 'ursuingpay'),
('residing', 'esidingray'), ('sentence', 'entencesay'), ('revealed',
'evealedray'), ('decrease', 'ecreaseday'), ('speedily', 'eedilyspay'),
('commands', 'ommandscay'), ('mistaken', 'istakenmay'), ('settling',
'ettlingsay'), ('polished', 'olishedpay'), ('stanhill', 'anhillstay'),
('retailed', 'etailedray'), ('pointing', 'ointingpay'), ('strongly',
'onglystray'), ('subjects', 'ubjectssay'), ('calmness', 'almnesscay'),
('doctrine', 'octrineday'), ('handsome', 'andsomehay'), ('profited',
'ofitedpray'), ('planning', 'anningplay'), ('denoting', 'enotingday'),
('commonly', 'ommonlycay'), ('bestowed', 'estowedbay'), ('marriage',
'arriagemay'), ('presents', 'esentspray'), ('complete', 'ompletecay'),
('striking', 'ikingstray'), ('needless', 'eedlessnay'), ('plighted',
'ightedplay'), ('tranquil', 'anquiltray'), ('spacious', 'aciousspay'),
('nonsense', 'onsensenay'), ('likeness', 'ikenesslay'), ('valuable',
'aluablevay'), ('laudable', 'audablelay'), ('delirium', 'eliriumday'),
('believes', 'elievesbay'), ('premises', 'emisespray'), ('presumed',
'esumedpray'), ('coldness', 'oldnesscay'), ('metaphor', 'etaphormay'),
('thankful', 'ankfulthay'), ('suffered', 'ufferedsay'), ('preserve',
'eservepray'), ('resorted', 'esortedray'), ('felicity', 'elicityfay'),
('betrayed', 'etrayedbay'), ('throwing', 'owingthray'), ('powerful',
'owerfulpay'), ('humoured', 'umouredhay'), ('practise', 'actisepray'),
('coloured', 'olouredcay'), ('regiment', 'egimentray'), ('stubborn',
'ubbornstay'), ('huswifes', 'uswifeshay'), ('painting', 'aintingpay'),
('friendly', 'iendlyfray'), ('happened', 'appenedhay'), ('moderate',
'oderatemay'), ('solitary', 'olitarysay'), ('wildness', 'ildnessway'),
('marianne', 'ariannemay'), ('delicacy', 'elicacyday'), ('progress',
'ogresspray'), ('coachman', 'oachmancay'), ('grieving', 'ievinggray'),
('troubles', 'oublestray'), ('coolness', 'oolnesscay'), ('possibly',

'ossiblypay'), ('relating', 'elatingray'), ('patterns', 'atternspay'),
('consists', 'onsistscay'), ('creating', 'eatingcray'), ('confused',
'onfusedcay'), ('critical', 'iticalcray'), ('laboured', 'abouredlay'),
('minutely', 'inutelymay'), ('tendency', 'endencytay'), ('distrust',
'istrustday'), ('distress', 'istressday'), ('conclude', 'oncludecay'),
('sixpence', 'ixpencesay'), ('somewhat', 'omewhatsay'), ('coincide',
'oincidecay'), ('harmless', 'armlesshay'), ('juvenile', 'uvenilejay'),
('pleasing', 'easingplay'), ('precious', 'eciouspray'), ('restrain',
'estrainray'), ('produced', 'oducedpray'), ('renewing', 'enewingray'),
('pinching', 'inchingpay'), ('severely', 'everelysay'), ('blossoms',
'ossomsblay'), ('heighten', 'eightenhay'), ('wandered', 'anderedway'),
('properly', 'operlypray'), ('veracity', 'eracityvay'), ('released',
'eleasedray'), ('transact', 'ansacttray'), ('helpmate', 'elpmatehay'),
('business', 'usinessbay'), ('concerto', 'oncertocay'), ('reserved',
'eservedray'), ('reliance', 'elianceray'), ('headache', 'eadachehay'),
('families', 'amiliesfay'), ('centered', 'enteredcay'), ('mulberry',
'ulberrymay'), ('sensible', 'ensiblesay'), ('steepest', 'eepeststay'),
('compared', 'omparedcay'), ('speaking', 'eakingspay'), ('censured',
'ensuredcay'), ('screened', 'eenedscray'), ('pictures', 'icturespay'),
('reaching', 'eachingray'), ('wonderer', 'ondererway'), ('whenever',
'eneverwhay'), ('detested', 'etestedday'), ('dispatch', 'ispatchday'),
('shutters', 'uttersshay'), ('disgrace', 'isgraceday'), ('dejected',
'ejectedday'), ('conquest', 'onquestcay'), ('foretell', 'oretellfay'),
('weakened', 'eakenedway'), ('brooding', 'oodingbray'), ('shedding',
'eddingshay'), ('poignant', 'oignantpay'), ('trembled', 'embledtray'),
('resolute', 'esoluteray'), ('reverted', 'evertedray'), ('silenced',
'ilencedsay'), ('jointure', 'ointurejay')], (10, 13): [('illaudable',
'illaudableway'), ('incivility', 'incivilityway'), ('oppression',
'oppressionway'), ('acquainted', 'acquaintedway'), ('economical',
'economicalway'), ('ebullition', 'ebullitionway'), ('affronting',
'affrontingway'), ('unforeseen', 'unforeseenway'), ('auspicious',
'auspiciousway'), ('indecorous', 'indecorousway'), ('attractive',
'attractiveway'), ('exercising', 'exercisingway'), ('unsatiable',
'unsatiableway'), ('individual', 'individualway'), ('enthusiasm',
'enthusiasmway'), ('announcing', 'announcingway'), ('indistinct',
'indistinctway'), ('affliction', 'afflictionway'), ('imaginable',
'imaginableway'), ('obstructed', 'obstructedway'), ('improperly',
'improperlyway'), ('illiterate', 'illiterateway'), ('interested',
'interestedway'), ('infamously', 'infamouslyway'), ('amusements',
'amusementsway'), ('endeavours', 'endeavoursway'), ('inhabiting',
'inhabitingway'), ('indisposed', 'indisposedway'), ('acquiesced',
'acquiescedway'), ('anticipate', 'anticipateway'), ('accurately',
'accuratelyway'), ('influenced', 'influencedway'), ('assemblies',
'assembliesway'), ('approached', 'approachedway'), ('accusation',
'accusationway'), ('apparently', 'apparentlyway'), ('intimately',
'intimatelyway'), ('uneasiness', 'uneasinessway'), ('alteration',
'alterationway'), ('artificial', 'artificialway'), ('admiration',
'admirationway'), ('accidental', 'accidentalway'), ('unexpected',

```
'unexpectedway'), ('insensible', 'insensibleway'), ('expressing',
'expressingway'), ('accelerate', 'accelerateway'), ('everything',
'everythingway'), ('increasing', 'increasingway'), ('undervalue',
'undervalueway'), ('invaluable', 'invaluableway'), ('affections',
'affectionsway'), ('infinitely', 'infinitelyway'), ('unsuitable',
'unsuitableway'), ('authorised', 'authorisedway'), ('impudently',
'impudentlyway'), ('unpardoned', 'unpardonedway'), ('inevitable',
'inevitableway'), ('ungenerous', 'ungenerousway'), ('impression',
'impressionway'), ('indelicacy', 'indelicacyway'), ('inducement',
'inducementway'), ('unaffected', 'unaffectedway'), ('instigated',
'instigatedway'), ('inquietude', 'inquietudeway'), ('attempting',
'attemptingway'), ('experience', 'experienceway'), ('impatience',
'impatienceway'), ('indulgence', 'indulgenceway'), ('affability',
'affabilityway'), ('unavailing', 'unavailingway'), ('incommoded',
'incommodedway'), ('altogether', 'altogetherway')], (4, 6): [('done', 'oneday'),
('wife', 'ifeway'), ('plea', 'eaplay'), ('mall', 'allmay'), ('sang', 'angsay'),
('clue', 'ueclay'), ('bell', 'ellbay'), ('bind', 'indbay'), ('join', 'oinjay'),
('went', 'entway'), ('vile', 'ilevay'), ('hens', 'enshay'), ('tore', 'oretay'),
('rage', 'ageray'), ('flow', 'owflay'), ('gate', 'ategay'), ('palm', 'almpay'),
('give', 'ivegay'), ('lips', 'ipslay'), ('sink', 'inksay'), ('rent', 'entray'),
('gave', 'avegay'), ('joke', 'okejay'), ('miss', 'issmay'), ('mere', 'eremay'),
('made', 'ademay'), ('dogs', 'ogsday'), ('bent', 'entbay'), ('duly', 'ulyday'),
('when', 'enwhay'), ('rate', 'ateray'), ('sure', 'uresay'), ('land', 'andlay'),
('pink', 'inkpay'), ('hint', 'inthay'), ('cast', 'astcay'), ('park', 'arkpay'),
('bold', 'oldbay'), ('door', 'oorday'), ('bulk', 'ulkbay'), ('doat', 'oatday'),
('coat', 'oatcay'), ('date', 'ateday'), ('foot', 'ootfay'), ('hand', 'andhay'),
('damp', 'ampday'), ('pall', 'allpay'), ('goes', 'oesgay'), ('rung', 'ungray'),
('none', 'onenay'), ('game', 'amegay'), ('nods', 'odsnay'), ('read', 'eadray'),
('gold', 'oldgay'), ('sing', 'ingsay'), ('puts', 'utspay'), ('cats', 'atscay'),
('many', 'anymay'), ('dare', 'areday'), ('noon', 'oonnay'), ('said', 'aidsay'),
('mark', 'arkmay'), ('case', 'asecay'), ('hate', 'atehay'), ('word', 'ordway'),
('beat', 'eatbay'), ('lose', 'oselay'), ('fish', 'ishfay'), ('laid', 'aidlay'),
('fine', 'inefay'), ('snug', 'ugsnay'), ('moss', 'ossmay'), ('heir', 'eirhay'),
('sigh', 'ighsay'), ('mend', 'endmay'), ('sake', 'akesay'), ('pass', 'asspay'),
('debt', 'ebtday'), ('lane', 'anelay'), ('days', 'aysday'), ('deep', 'eepday'),
('song', 'ongsay'), ('spot', 'otspay'), ('took', 'ooktay'), ('both', 'othbay'),
('year', 'yearay'), ('mama', 'amamay'), ('hill', 'illhay'), ('beds', 'edsbay'),
('less', 'esslay'), ('gown', 'owngay'), ('kill', 'illkay'), ('free', 'eefray'),
('hall', 'allhay'), ('room', 'oomray'), ('gray', 'aygray'), ('beef', 'eefbay'),
('kept', 'eptkay'), ('what', 'atwhay'), ('blue', 'ueblay'), ('fell', 'ellfay'),
('draw', 'awdray'), ('play', 'ayplay'), ('baby', 'abybay'), ('rude', 'uderay'),
('trim', 'imtray'), ('chat', 'atchay'), ('bely', 'elybay'), ('fled', 'edflay'),
('past', 'astpay'), ('wait', 'aitway'), ('meal', 'ealmay'), ('wont', 'ontway'),
('week', 'eekway'), ('best', 'estbay'), ('line', 'inelay'), ('lawn', 'awnlay'),
('tree', 'eetray'), ('tend', 'endtay'), ('find', 'indfay'), ('five', 'ivefay'),
('gone', 'onegay'), ('need', 'eednay'), ('drew', 'ewdray'), ('more', 'oremay'),
('cool', 'oolcay'), ('suit', 'uitsay'), ('from', 'omfray'), ('cure', 'urecay'),
('will', 'illway'), ('slow', 'owslay'), ('they', 'eythay'), ('name', 'amenay'),
```

```
('shot', 'otshay'), ('come', 'omecay'), ('tied', 'iedtay'), ('came', 'amecay'),
('boys', 'oysbay'), ('wish', 'ishway'), ('hour', 'ourhay'), ('kind', 'indkay'),
('mile', 'ilemay'), ('quit', 'itquay'), ('john', 'ohnjay'), ('hazy', 'azyhay'),
('fall', 'allfay'), ('sell', 'ellsay'), ('tale', 'aletay'), ('grew', 'ewgray'),
('seek', 'eeksay'), ('muff', 'uffmay'), ('soft', 'oftsay'), ('same', 'amesay'),
('body', 'odybay'), ('head', 'eadhay'), ('like', 'ikelay'), ('here', 'erehay'),
('help', 'elphay'), ('sail', 'ailsay'), ('pope', 'opepay'), ('deny', 'enyday'),
('bond', 'ondbay'), ('bath', 'athbay'), ('blow', 'owblay'), ('hold', 'oldhay'),
('safe', 'afesay'), ('path', 'athpay'), ('feed', 'eedfay'), ('thus', 'usthay'),
('four', 'ourfay'), ('lest', 'estlay'), ('thin', 'inthay')], (11, 14):
[('approaching', 'approachingway'), ('unequivocal', 'unequivocalway'),
('exceedingly', 'exceedinglyway'), ('application', 'applicationway'),
('observation', 'observationway'), ('unhappiness', 'unhappinessway'),
('interrupted', 'interruptedway'), ('enlightened', 'enlightenedway'),
('appearances', 'appearancesway'), ('unconnected', 'unconnectedway'),
('unconcerned', 'unconcernedway'), ('indignation', 'indignationway'),
('accommodate', 'accommodateway'), ('inattention', 'inattentionway'),
('undoubtedly', 'undoubtedlyway'), ('intercourse', 'intercourseway'),
('imperfectly', 'imperfectlyway'), ('imprudently', 'imprudentlyway'),
('affectation', 'affectationway'), ('instigation', 'instigationway'),
('inclination', 'inclinationway'), ('undervalued', 'undervaluedway'),
('affirmative', 'affirmativeway'), ('inconstancy', 'inconstancyway'),
('universally', 'universallyway'), ('engagements', 'engagementsway'),
('exclamation', 'exclamationway'), ('elucidation', 'elucidationway'),
('accustomary', 'accustomaryway'), ('approbation', 'approbationway'),
('unnecessary', 'unnecessaryway'), ('associating', 'associatingway'),
('attributing', 'attributingway'), ('unobtrusive', 'unobtrusiveway'),
('impropriety', 'improprietyway'), ('examination', 'examinationway'),
('alterations', 'alterationsway'), ('unexpressed', 'unexpressedway'),
('abridgement', 'abridgementway'), ('exhilarated', 'exhilaratedway'),
('unsuspected', 'unsuspectedway'), ('independent', 'independentway'),
('alternately', 'alternatelyway'), ('enforcement', 'enforcementway'),
('undertaking', 'undertakingway'), ('outstripped', 'outstrippedway'),
('appointment', 'appointmentway'), ('explanation', 'explanationway'),
('experienced', 'experiencedway'), ('unknowingly', 'unknowinglyway'),
('undesirable', 'undesirableway'), ('ascertained', 'ascertainedway'),
('enumeration', 'enumerationway'), ('alleviation', 'alleviationway'),
('instruction', 'instructionway'), ('undeserving', 'undeservingway'),
('earnestness', 'earnestnessway'), ('insincerity', 'insincerityway'),
('accompanied', 'accompaniedway'), ('employments', 'employmentsway'),
('inferiority', 'inferiorityway'), ('inexcusable', 'inexcusableway'),
('introducing', 'introducingway'), ('extravagant', 'extravagantway'),
('impatiently', 'impatientlyway'), ('attentively', 'attentivelyway'),
('eligibility', 'eligibilityway'), ('uncertainty', 'uncertaintyway')], (7, 10):
[('anxiety', 'anxietyway'), ('attacks', 'attacksway'), ('ailment',
'ailmentway'), ('awkward', 'awkwardway'), ('effects', 'effectsway'), ('induced',
'inducedway'), ('uncivil', 'uncivilway'), ('against', 'againstway'), ('apology',
'apologyway'), ('endless', 'endlessway'), ('alluded', 'alludedway'), ('ability',
```

'abilityway'), ('address', 'addressway'), ('engaged', 'engagedway'), ('eagerly',
'eagerlyway'), ('expects', 'expectsway'), ('elliott', 'elliottway'), ('oneself',
'oneselfway'), ('anybody', 'anybodyway'), ('augment', 'augmentway'), ('utility',
'utilityway'), ('unquiet', 'unquietway'), ('infancy', 'infancyway'), ('altered',
'alteredway'), ('unequal', 'unequalway'), ('adapted', 'adaptedway'), ('useless',
'uselessway'), ('opposed', 'opposedway'), ('artless', 'artlessway'), ('imagery',
'imageryway'), ('amiably', 'amiablyway'), ('uncouth', 'uncouthway'), ('affixed',
'affixedway'), ('impulse', 'impulseway'), ('opinion', 'opinionway'), ('imitate',
'imitateway'), ('admires', 'admiresway'), ('affront', 'affrontway'), ('amusing',
'amusingway'), ('accents', 'accentsway'), ('implied', 'impliedway'), ('ecstasy',
'ecstasyway'), ('illness', 'illnessway'), ('entreat', 'entreatway'), ('another',
'anotherway'), ('opening', 'openingway'), ('england', 'englandway'), ('ensured',
'ensuredway'), ('imputed', 'imputedway'), ('enabled', 'enabledway'), ('offence',
'offenceway'), ('insured', 'insuredway'), ('urgency', 'urgencyway'), ('amazing',
'amazingway'), ('uniform', 'uniformway'), ('avarice', 'avariceway'), ('ellison',
'ellisonway'), ('endowed', 'endowedway'), ('ashamed', 'ashamedway'), ('eloping',
'elopingway'), ('inhabit', 'inhabitway'), ('advised', 'advisedway'), ('attract',
'attractway'), ('already', 'alreadyway'), ('enquire', 'enquireway'), ('affects',
'affectsway'), ('invalid', 'invalidway')], (9, 11): [('genuinely',
'enuinelygay'), ('columella', 'olumellacay'), ('vouchsafe', 'ouchsafevay'),
('forwarded', 'orwardedfay'), ('scattered', 'atteredscay'), ('christmas',
'istmaschray'), ('crimsoned', 'imsonedcray'), ('contrives', 'ontrivescay'),
('miserable', 'iserablemay'), ('preceding', 'ecedingpray'), ('healthful',
'ealthfulhay'), ('confiding', 'onfidingcay'), ('perfectly', 'erfectlypay'),
('prospects', 'ospectspray'), ('withdrawn', 'ithdrawnway'), ('lingering',
'ingeringlay'), ('contained', 'ontainedcay'), ('depressed', 'epressedday'),
('disgusted', 'isgustedday'), ('promoting', 'omotingpray'), ('supported',
'upportedsay'), ('gradually', 'aduallygray'), ('disturbed', 'isturbedday'),
('perpetual', 'erpetualpay'), ('pronounce', 'onouncepray'), ('sincerely',
'incerelysay'), ('seventeen', 'eventeensay'), ('presented', 'esentedpray'),
('medicines', 'edicinesmay'), ('syllables', 'yllablessay'), ('behaviour',
'ehaviourbay'), ('niggardly', 'iggardlynay'), ('commonest', 'ommonestcay'),
('necessary', 'ecessarynay'), ('prettiest', 'ettiestpray'), ('condemned',
'ondemnedcay'), ('conquests', 'onquestscay'), ('fashioned', 'ashionedfay'),
('hysterics', 'ystericshay'), ('smallness', 'allnesssmay'), ('professed',
'ofessedpray'), ('conceited', 'onceitedcay'), ('compliant', 'ompliantcay'),
('frankness', 'anknessfray'), ('merriment', 'errimentmay'), ('resembled',
'esembledray'), ('leisurely', 'eisurelylay'), ('dispersed', 'ispersedday'),
('departure', 'epartureday'), ('pecuniary', 'ecuniarypay'), ('frivolous',
'ivolousfray'), ('situation', 'ituationsay'), ('discourse', 'iscourseday'),
('resettled', 'esettledray'), ('beholding', 'eholdingbay'), ('gentleman',
'entlemangay'), ('chagrined', 'agrinedchay'), ('struggled', 'uggledstray'),
('preparing', 'eparingpray'), ('respected', 'espectedray'), ('newspaper',
'ewspapernay'), ('justified', 'ustifiedjay'), ('sacrifice', 'acrificesay'),
('stammered', 'ammeredstay'), ('smartness', 'artnesssmay'), ('continued',
'ontinuedcay'), ('contrived', 'ontrivedcay'), ('reconcile', 'econcileray'),
('modestest', 'odestestmay'), ('cordially', 'ordiallycay'), ('sentiment',
'entimentsay'), ('shameless', 'amelessshay'), ('reflected', 'eflectedray'),

('feelingly', 'eelinglyfay'), ('confident', 'onfidentcay'), ('proposals',
'oposalspray'), ('warehouse', 'arehouseway'), ('deviation', 'eviationday'),
('statement', 'atementstay'), ('submitted', 'ubmittedsay'), ('believing',
'elievingbay'), ('gentility', 'entilitygay'), ('resources', 'esourcesray'),
('composure', 'omposurecay'), ('deserving', 'eservingday'), ('vehemence',
'ehemencevay'), ('disgraced', 'isgracedday'), ('perplexed', 'erplexedpay'),
('prejudice', 'ejudicepray'), ('quickened', 'ickenedquay'), ('blundered',
'underedblay'), ('strongest', 'ongeststray'), ('prevailed', 'evailedpray'),
('languages', 'anguageslay'), ('reanimate', 'eanimateray'), ('stationer',
'ationerstay'), ('reminding', 'emindingray'), ('blameless', 'amelessblay'),
('principle', 'inciplepray'), ('hardening', 'ardeninghay'), ('suggested',
'uggestedsay'), ('desperate', 'esperateday'), ('landscape', 'andscapelay'),
('paragraph', 'aragraphpay'), ('provision', 'ovisionpray'), ('relations',
'elationsray'), ('fruitless', 'uitlessfray'), ('christian', 'istianchray'),
('stipulate', 'ipulatestay'), ('constancy', 'onstancycay'), ('gentlemen',
'entlemengay'), ('venturing', 'enturingvay'), ('northward', 'orthwardnay'),
('neighbour', 'eighbournay'), ('weakening', 'eakeningway'), ('persisted',
'ersistedpay'), ('malignant', 'alignantmay'), ('wandering', 'anderingway'),
('witnesses', 'itnessesway'), ('dangerous', 'angerousday'), ('safeguard',
'afeguardsay'), ('treasured', 'easuredtray'), ('cleveland', 'evelandclay'),
('patronage', 'atronagepay'), ('despising', 'espisingday'), ('foolishly',
'oolishlyfay'), ('seriously', 'eriouslysay'), ('restraint', 'estraintray'),
('september', 'eptembersay'), ('distrusts', 'istrustsday'), ('temperate',
'emperatetay'), ('finishing', 'inishingfay'), ('persuaded', 'ersuadedpay'),
('stupified', 'upifiedstay'), ('declaring', 'eclaringday'), ('dejection',
'ejectionday'), ('soothings', 'oothingssay'), ('conveying', 'onveyingcay'),
('possessed', 'ossessedpay'), ('described', 'escribedday'), ('spreading',
'eadingspray'), ('foresight', 'oresightfay'), ('political', 'oliticalpay'),
('concluded', 'oncludedcay'), ('doubtless', 'oubtlessday'), ('confusion',
'onfusioncay'), ('disliking', 'islikingday'), ('carriages', 'arriagescay'),
('favourite', 'avouritefay'), ('brilliant', 'illiantbray'), ('hastening',
'asteninghay'), ('precisely', 'eciselypray'), ('wondering', 'onderingway'),
('shoulders', 'oulderssshay'), ('honorably', 'onorablyhay'), ('privilege',
'ivilegepray'), ('deference', 'eferenceday'), ('vulgarity', 'ulgarityvay'),
('reprimand', 'eprimandray'), ('surveying', 'urveyingsay'), ('handsomer',
'andsomerhay'), ('shivering', 'iveringshay'), ('tolerable', 'olerabletay'),
('requiring', 'equiringray'), ('precision', 'ecisionpray'), ('courtland',
'ourtlandcay'), ('sportsman', 'ortsmanspay'), ('happiness', 'appinesshay'),
('pointedly', 'ointedlypay'), ('benefited', 'enefitedbay'), ('substance',
'ubstancesay'), ('beginning', 'eginningbay'), ('gratitude', 'atitudegray'),
('negligent', 'egligentnay'), ('belonging', 'elongingbay'), ('mortified',
'ortifiedmay'), ('neglected', 'eglectednay'), ('shillings', 'illingsshay'),
('warranted', 'arrantedway'), ('performed', 'erformedpay'), ('necessity',
'ecessitynay'), ('slightest', 'ightestslay'), ('guardians', 'uardiansgay'),
('preferred', 'eferredpray'), ('forgotten', 'orgottenfay'), ('refreshed',
'efreshedray'), ('barouches', 'arouchesbay'), ('grievance', 'ievancegray'),
('seduction', 'eductionsay'), ('comprised', 'omprisedcay'), ('listening',
'isteninglay'), ('detection', 'etectionday'), ('narration', 'arrationnay'),

```
('deceiving', 'eceivingday'), ('provoking', 'ovokingpray'), ('cessation',
'essationcay'), ('questions', 'estionsquay'), ('continual', 'ontinualcay'),
('faintness', 'aintnessfay'), ('diffident', 'iffidentday'), ('referring',
'eferringray'), ('blessings', 'essingsblay'), ('therefore', 'ereforethay'),
('frugality', 'ugalityfray'), ('reprobate', 'eprobateray')], (11, 13):
[('tranquilize', 'anquilizetray'), ('fashionable', 'ashionablefay'),
('respectable', 'espectableray'), ('publication', 'ublicationpay'),
('superintend', 'uperintendsay'), ('recollected', 'ecollectedray'),
('cheerfuller', 'eerfullerchay'), ('concurrence', 'oncurrencecay'),
('restorative', 'estorativeray'), ('reproaching', 'eproachingray'),
('declaration', 'eclarationday'), ('discharging', 'ischargingday'),
('proclaiming', 'oclaimingpray'), ('discovering', 'iscoveringday'),
('possibility', 'ossibilitypay'), ('contributed', 'ontributedcay'),
('captivating', 'aptivatingcay'), ('misinformed', 'isinformedmay'),
('carefulness', 'arefulnesscay'), ('philosophic', 'ilosophicphay'),
('calculation', 'alculationcay'), ('consumption', 'onsumptioncay'),
('confinement', 'onfinementcay'), ('certainties', 'ertaintiescay'),
('destination', 'estinationday'), ('separations', 'eparationssay'),
('voluntarily', 'oluntarilyvay'), ('connoisseur', 'onnoisseurcay'),
('refinements', 'efinementsray'), ('subservient', 'ubservientsay'),
('marlborough', 'arlboroughmay'), ('sagaciously', 'agaciouslysay'),
('fascinating', 'ascinatingfay'), ('comfortable', 'omfortablecay'),
('reflections', 'eflectionsray'), ('sensibility', 'ensibilitysay'),
('forbearance', 'orbearancefay'), ('mischievous', 'ischievousmay'),
('disgraceful', 'isgracefulday'), ('temporizing', 'emporizingtay'),
('sacrificing', 'acrificingsay'), ('spontaneous', 'ontaneousspay'),
('flourishing', 'ourishingflay'), ('lamentation', 'amentationlay'),
('meditations', 'editationsmay'), ('quarrelling', 'arrellingquay'),
('discussions', 'iscussionsday'), ('supposition', 'uppositionsay'),
('comfortably', 'omfortablycay'), ('disappeared', 'isappearedday'),
('preparation', 'eparationpray'), ('consistency', 'onsistencycay'),
('compliments', 'omplimentscay'), ('promptitude', 'omptitudepray'),
('determining', 'eterminingday'), ('countenance', 'ountenancecay'),
('maintenance', 'aintenancemay'), ('contraction', 'ontractioncay'),
('dissipation', 'issipationday'), ('comparative', 'omparativecay'),
('desperation', 'esperationday'), ('displeasure', 'ispleasureday'),
('despatching', 'espatchingday'), ('playfellows', 'ayfellowsplay'),
('plantations', 'antationsplay'), ('vindication', 'indicationvay'),
('significant', 'ignificantsay'), ('discontents', 'iscontentsday'),
('housekeeper', 'ousekeeperhay'), ('playfulness', 'ayfulnessplay'),
('threatening', 'eateningthray'), ('meritorious', 'eritoriousmay'),
('tediousness', 'ediousnesstay'), ('convenience', 'onveniencecay'),
('possessions', 'ossessionspay'), ('consolation', 'onsolationcay'),
('composition', 'ompositioncay'), ('proposition', 'opositionpray'),
('correctness', 'orrectnesscay'), ('consequence', 'onsequencecay'),
('barbarously', 'arbarouslybay'), ('booksellers', 'ooksellersbay'),
('conjectures', 'onjecturescay'), ('violoncello', 'ioloncellovay'),
('dissembling', 'issemblingday'), ('rapturously', 'apturouslyray'),
```

```
('speculative', 'eculativespay'), ('remembering', 'ememberingray'),
('probability', 'obabilitypray'), ('punctuality', 'unctualitypay'),
('forgiveness', 'orgivenessfay'), ('disapproves', 'isapprovesday'),
('troublesome', 'oublesometray')], (3, 5): [('say', 'aysay'), ('for', 'orfay'),
('get', 'etgay'), ('pin', 'inpay'), ('not', 'otnay'), ('fit', 'itfay'), ('rap',
'apray'), ('lip', 'iplay'), ('pry', 'ypray'), ('rob', 'obray'), ('fir',
'irfay'), ('can', 'ancay'), ('two', 'otway'), ('him', 'imhay'), ('son',
'onsay'), ('let', 'etlay'), ('try', 'ytray'), ('hot', 'othay'), ('sum',
'umsay'), ('got', 'otgay'), ('law', 'awlay'), ('how', 'owhay'), ('far',
'arfay'), ('the', 'ethay'), ('sky', 'yskay'), ('sly', 'yslay'), ('cut',
'utcay'), ('hop', 'ophay'), ('had', 'adhay'), ('was', 'asway'), ('bow',
'owbay'), ('hue', 'uehay'), ('now', 'ownay'), ('vex', 'exvay'), ('yet',
'yetay'), ('sit', 'itsay'), ('due', 'ueday'), ('cap', 'apcay'), ('sat',
'atsay'), ('hit', 'ithay'), ('see', 'eesay'), ('pay', 'aypay'), ('nor',
'ornay'), ('fly', 'yflay'), ('dog', 'ogday'), ('too', 'ootay'), ('six',
'ixsay'), ('lap', 'aplay'), ('new', 'ewnay'), ('led', 'edlay'), ('sin',
'insay'), ('ten', 'entay'), ('his', 'ishay'), ('day', 'ayday')], (14, 17):
[('insurmountable', 'insurmountableway'), ('inconveniences',
'inconveniencesway'), ('aggrandizement', 'aggrandizementway'),
('irreconcilable', 'irreconcilableway'), ('unacknowledged',
'unacknowledgedway'), ('unpleasantness', 'unpleasantnessway')], (13, 16):
[('understanding', 'understandingway'), ('intentionally', 'intentionallyway'),
('embarrassment', 'embarrassmentway'), ('unembarrassed', 'unembarrassedway'),
('administering', 'administeringway'), ('instantaneous', 'instantaneousway'),
('intrinsically', 'intrinsicallyway'), ('indispensable', 'indispensableway'),
('impoverishing', 'impoverishingway'), ('opportunities', 'opportunitiesway'),
('impertinently', 'impertinentlyway'), ('independently', 'independentlyway'),
('encroachments', 'encroachmentsway'), ('uninteresting', 'uninterestingway'),
('investigation', 'investigationway'), ('unaccountable', 'unaccountableway'),
('irrepressible', 'irrepressibleway'), ('impossibility', 'impossibilityway')],
(10, 14): [('thirty-six', 'irtythay-ixsay')], (8, 11): [('increase',
'increaseway'), ('agitated', 'agitatedway'), ('unkindly', 'unkindlyway'),
('occurred', 'occurredway'), ('arranged', 'arrangedway'), ('appetite',
'appetiteway'), ('inclined', 'inclinedway'), ('indolent', 'indolentway'),
('inquired', 'inquiredway'), ('evenings', 'eveningsway'), ('obliging',
'obligingway'), ('equalled', 'equalledway'), ('accepted', 'acceptedway'),
('extended', 'extendedway'), ('entreaty', 'entreatyway'), ('exigence',
'exigenceway'), ('election', 'electionway'), ('examined', 'examinedway'),
('inviting', 'invitingway'), ('embitter', 'embitterway'), ('idleness',
'idlenessway'), ('admirers', 'admirersway'), ('editions', 'editionsway'),
('extorted', 'extortedway'), ('outweigh', 'outweighway'), ('attended',
'attendedway'), ('addition', 'additionway'), ('entitled', 'entitledway'),
('awakened', 'awakenedway'), ('attacked', 'attackedway'), ('arriving',
'arrivingway'), ('overlook', 'overlookway'), ('eminence', 'eminenceway'),
('adhering', 'adheringway'), ('obtained', 'obtainedway'), ('openness',
'opennessway'), ('allusion', 'allusionway'), ('anything', 'anythingway'),
('overcome', 'overcomeway'), ('announce', 'announceway'), ('enjoying',
'enjoyingway'), ('outlived', 'outlivedway'), ('emphasis', 'emphasisway'),
```

('insulted', 'insultedway'), ('esteemed', 'esteemedway'), ('alphabet',
'alphabetway'), ('enquired', 'enquiredway'), ('accident', 'accidentway'),
('occupied', 'occupiedway'), ('ellisons', 'ellisonsway'), ('avoiding',
'avoidingway'), ('attained', 'attainedway'), ('unwarily', 'unwarilyway'),
('employed', 'employedway'), ('altering', 'alteringway'), ('intimate',
'intimateway'), ('effected', 'effectedway'), ('offences', 'offencesway'),
('entirely', 'entirelyway'), ('abounded', 'aboundedway'), ('actually',
'actuallyway'), ('identify', 'identifyway'), ('applying', 'applyingway'),
('exterior', 'exteriorway'), ('auditors', 'auditorsway'), ('unfairly',
'unfairlyway'), ('eighteen', 'eighteenway'), ('embraced', 'embracedway'),
('unfolded', 'unfoldedway'), ('injuries', 'injuriesway'), ('elevated',
'elevatedway'), ('alarming', 'alarmingway'), ('adequate', 'adequateway'),
('unabated', 'unabatedway'), ('incurred', 'incurredway'), ('uncommon',
'uncommonway'), ('unawares', 'unawaresway'), ('unsuited', 'unsuitedway'),
('instance', 'instanceway'), ('interval', 'intervalway'), ('audacity',
'audacityway')], (5, 8): [('other', 'otherway'), ('ideas', 'ideasway'),
('agony', 'agonyway'), ('avoid', 'avoidway'), ('awful', 'awfulway'), ('april',
'aprilway'), ('enter', 'enterway'), ('infer', 'inferway'), ('abuse',
'abuseway'), ('ivory', 'ivoryway'), ('adopt', 'adoptway'), ('image',
'imageway'), ('agree', 'agreeway'), ('alike', 'alikeway'), ('earth',
'earthway'), ('ended', 'endedway'), ('owner', 'ownerway'), ('using',
'usingway'), ('ought', 'oughtway'), ('abhor', 'abhorway'), ('altar',
'altarway'), ('often', 'oftenway'), ('atone', 'atoneway'), ('angel',
'angelway'), ('arise', 'ariseway'), ('under', 'underway'), ('abode',
'abodeway'), ('exert', 'exertway'), ('usual', 'usualway'), ('aches',
'achesway'), ('enjoy', 'enjoyway'), ('urged', 'urgedway'), ('admit',
'admitway'), ('aside', 'asideway'), ('equal', 'equalway'), ('alive',
'aliveway'), ('arose', 'aroseway'), ('eight', 'eightway'), ('opera',
'operaway'), ('ennui', 'ennuiway'), ('utter', 'utterway'), ('ankle',
'ankleway')], (2, 5): [('is', 'isway'), ('ay', 'ayway'), ('ah', 'ahway'), ('ii',
'iiway'), ('or', 'orway'), ('am', 'amway'), ('if', 'ifway'), ('an', 'anway')],
(12, 14): [('tranquillity', 'anquillitytray'), ('humiliations',
'umiliationshay'), ('contributing', 'ontributingcay'), ('significancy',
'ignificancysay'), ('wretchedness', 'etchednesswray'), ('prodigiously',
'odigiouslypray'), ('particularly', 'articularlypay'), ('carelessness',
'arelessnesscay'), ('refreshments', 'efreshmentsray'), ('satisfaction',
'atisfactionsay'), ('construction', 'onstructioncay'), ('contribution',
'ontributioncay'), ('gracefulness', 'acefulnessgray'), ('consequences',
'onsequencescay'), ('grandmothers', 'andmothersgray'), ('proportioned',
'oportionedpray'), ('perverseness', 'erversenesspay'), ('conciliation',
'onciliationcay'), ('representing', 'epresentingray'), ('watchfulness',
'atchfulnessway'), ('recollecting', 'ecollectingray'), ('hesitatingly',
'esitatinglyhay'), ('pleasantness', 'easantnessplay'), ('benefactress',
'enefactressbay'), ('distractedly', 'istractedlyday'), ('perseverance',
'erseverancepay'), ('performances', 'erformancespay'), ('disapproving',
'isapprovingday'), ('recollection', 'ecollectionray'), ('comprehended',
'omprehendedcay'), ('relinquished', 'elinquishedray'), ('conjecturing',
'onjecturingcay'), ('reservedness', 'eservednessray'), ('promontories',

'omontoriespray'), ('discontented', 'iscontentedday'), ('persecutions', 'ersecutionspay'), ('relationship', 'elationshipray'), ('triumphantly', 'iumphantlytray'), ('deliberation', 'eliberationday'), ('supplication', 'upplicationsay'), ('dishonorable', 'ishonorableday'), ('propensities', 'opensitiespray'), ('contemptuous', 'ontemptuouscay'), ('transporting', 'ansportingtray')], (13, 15): [('distinguished', 'istinguishedday'), ('resuscitation', 'esuscitationray'), ('particularity', 'articularitypay'), ('superannuated', 'uperannuatedsay'), ('disagreements', 'isagreementsday'), ('gentlemanlike', 'entlemanlikegay'), ('prepossessing', 'epossessingpray'), ('conscientious', 'onscientiouscay'), ('considerately', 'onsideratelycay'), ('felicitations', 'elicitationsfay'), ('prepossession', 'epossessionpray'), ('consternation', 'onsternationcay'), ('disproportion', 'isproportionday'), ('commiseration', 'ommiserationcay'), ('mortification', 'ortificationmay'), ('solicitations', 'olicitationssay'), ('qualification', 'alificationquay'), ('philanthropic', 'ilanthropicphay'), ('disappointing', 'isappointingday'), ('prescriptions', 'escriptionspray'), ('strengthening', 'engtheningstray')], (13, 17): [('mulberry-tree', 'ulberrymay-eetray'), ('good-humoured', 'oodgay-umouredhay'), ('mean-spirited', 'eanmay-iritedspay'), ('self-reproach', 'elfsay-eproachray'), ('drawing-table', 'awingdray-abletay'), ('well-disposed', 'ellway-isposedday'), ('proud-looking', 'oudpray-ookinglay'), ('flower-garden', 'owerflay-ardengay'), ('mansion-house', 'ansionmay-ousehay'), ('landing-place', 'andinglay-aceplay'), ('music-sellers', 'usicmay-ellerssay')], (4, 7): [('acts', 'actsway'), ('anne', 'anneway'), ('upon', 'uponway'), ('able', 'ableway'), ('inch', 'inchway'), ('ages', 'agesway'), ('envy', 'envyway'), ('idea', 'ideaway'), ('ends', 'endsway'), ('-and', '-andway'), ('eyed', 'eyedway'), ('arms', 'armsway'), ('over', 'overway'), ('easy', 'easyway'), ('ever', 'everway'), ('ours', 'oursway'), ('ears', 'earsway'), ('away', 'awayway')], (16, 20): [('pleasure-grounds', 'easureplay-oundsgray'), ('self-provocation', 'elfsay-ovocationpray')], (11, 15): [('bed-chamber', 'edbay-amberchay'), ('high-minded', 'ighhay-indedmay'), ('before-hand', 'eforebay-andhay'), ('watch-tower', 'atchway-owertay'), ('print-shops', 'intpray-opsshay'), ('card-tables', 'ardcay-ablestay'), ('post-horses', 'ostpay-orseshay'), ('good-nature', 'oodgay-aturenay'), ('dining-room', 'iningday-oomray'), ('high-church', 'ighhay-urchchay'), ('well-wisher', 'ellway-isherway'), ('bond-street', 'ondbay-eetstray'), ('carpet-work', 'arpetcay-orkway')], (2, 4): [('to', 'otay'), ('he', 'ehay'), ('no', 'onay'), ('me', 'emay')], (12, 16): [('good-hearted', 'oodgay-eartedhay'), ('well-judging', 'ellway-udgingjay'), ('drawing-room', 'awingdray-oomray'), ('common-place', 'ommoncay-aceplay'), ('self-control', 'elfsay-ontrolcay'), ('hand-writing', 'andhay-itingwray'), ('well-behaved', 'ellway-ehavedbay'), ('mantel-piece', 'antelmay-iecepay')], (10, 15): [('ill-humour', 'illway-umourhay'), ('ill-judged', 'illway-udgedjay'), ('ill-suited', 'illway-uitedsay')], (1, 3): [('c', 'cay')], (8, 13): [('ill-wind', 'illway-indway'), ('ill-will', 'illway-illway'), ('over-set', 'overway-etsay')], (17, 19): [('disqualifications', 'isqualificationsday')], (9, 16): [('by-and-by', 'ybay-andway-ybay')], (3, 6): [('aye', 'ayeway'), ('use', 'useway'), ('owe', 'oweway'), ('our', 'ourway'), ('old', 'oldway'), ('ear', 'earway'), ('its', 'itsway'), ('ill', 'illway')], (12, 17): [('ill-breeding', 'illway-eedingbray'), ('re-assurance', 'eray-assuranceway'), ('over-powered', 'overway-oweredpay'),

```
('ill-disposed', 'illway-isposedday')], (11, 16): [('ill-natured', 'illway-
aturednay')], (14, 16): [('reconciliation', 'econciliationray'),
('recommendation', 'ecommendationray'), ('reasonableness', 'easonablenessray'),
('disinclination', 'isinclinationday'), ('respectability', 'espectabilityray'),
('circumspection', 'ircumspectioncay'), ('considerations', 'onsiderationscay'),
('qualifications', 'alificationsquay'), ('congratulating', 'ongratulatingcay')],
(15, 17): [('dissatisfaction', 'issatisfactionday'), ('disrespectfully',
'isrespectfullyday'), ('conscientiously', 'onscientiouslycay'),
('misconstruction', 'isconstructionmay')], (15, 18): [('instantaneously',
'instantaneouslyway')], (15, 19): [('shooting-jacket', 'ootingshay-acketjay')],
(14, 21): [('sisters-in-law', 'isterssay-inway-awlay')], (17, 21): [('self-
condemnation', 'elfsay-ondemnationcay')], (13, 20): [('sister-in-law',
'istersay-inway-awlay')], (9, 14): [('ill-timed', 'illway-imedtay')], (13, 18):
[('ill-treatment', 'illway-eatmenttray'), ('old-fashioned', 'oldway-
ashionedfay'), ('self-interest', 'elfsay-interestway')], (16, 22): [('good-for-
nothing', 'oodgay-orfay-othingnay')], (9, 13): [('dove-cote', 'oveday-otecay'),
('tea-table', 'eatay-abletay'), ('work-bags', 'orkway-agsbay')], (1, 4): [('i',
'iway')], (16, 19): [('incomprehensible', 'incomprehensibleway')], (8, 12):
[('to-night', 'otay-ightnay'), ('land-tax', 'andlay-axtay')], (11, 18): [('tete-
a-tete', 'etetay-away-etetay')], (5, 9): [('to-do', 'otay-oday')]})
Moved models to GPU!
Epoch:   0 | Flops (T): 0.052877023776, Tokens (M) 0.052678, Train Loss 2.927,
Val Loss 2.900 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   1 | Flops (T): 0.105754047552, Tokens (M) 0.105356, Train Loss 2.828,
Val Loss 2.833 | Gen: oooooooooooooooooooo
Epoch:   2 | Flops (T): 0.158631071328, Tokens (M) 0.158034, Train Loss 2.749,
Val Loss 2.784 | Gen: aoooooooooooooooooo
Epoch:   3 | Flops (T): 0.211508095104, Tokens (M) 0.210712, Train Loss 2.677,
Val Loss 2.732 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   4 | Flops (T): 0.26438511888, Tokens (M) 0.26339, Train Loss 2.644, Val
Loss 2.712 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:   5 | Flops (T): 0.317262142656, Tokens (M) 0.316068, Train Loss 2.582,
Val Loss 2.801 | Gen: faaaaaaaaaaaaaaaaaaa
Epoch:   6 | Flops (T): 0.370139166432, Tokens (M) 0.368746, Train Loss 2.558,
Val Loss 2.482 | Gen: oooooooooooooooooooo
Epoch:   7 | Flops (T): 0.423016190208, Tokens (M) 0.421424, Train Loss 2.403,
Val Loss 2.379 | Gen: fooooooooooooooooooo
Epoch:   8 | Flops (T): 0.475893213984, Tokens (M) 0.474102, Train Loss 2.230,
Val Loss 2.220 | Gen: fooooooooooa
Epoch:   9 | Flops (T): 0.52877023776, Tokens (M) 0.52678, Train Loss 2.140, Val
Loss 2.306 | Gen: ffoooooooooooooa
Epoch:  10 | Flops (T): 0.581647261536, Tokens (M) 0.579458, Train Loss 2.085,
Val Loss 2.417 | Gen: ooooooooooooa
Epoch:  11 | Flops (T): 0.634524285312, Tokens (M) 0.632136, Train Loss 1.941,
Val Loss 1.971 | Gen: fooooooooayyyyyyyyy
Epoch:  12 | Flops (T): 0.687401309088, Tokens (M) 0.684814, Train Loss 1.821,
Val Loss 1.898 | Gen: ffoooooowwwrrrrway
Epoch:  13 | Flops (T): 0.740278332864, Tokens (M) 0.737492, Train Loss 1.752,
```

Val Loss 1.823 | Gen: ffffoooooooooorrrayyy
Epoch:  14 | Flops (T): 0.79315535664, Tokens (M) 0.79017, Train Loss 1.707, Val Loss 1.808 | Gen: ffffoowwwwwwayyyyyyy
Epoch:  15 | Flops (T): 0.846032380416, Tokens (M) 0.842848, Train Loss 1.684, Val Loss 1.806 | Gen: fffoowwwayyyyyyyyyyy
Epoch:  16 | Flops (T): 0.898909404192, Tokens (M) 0.895526, Train Loss 1.642, Val Loss 1.763 | Gen: fffrrrrrayyyyyyyyyyy
Epoch:  17 | Flops (T): 0.951786427968, Tokens (M) 0.948204, Train Loss 1.622, Val Loss 1.788 | Gen: fffffooooooorrrrrrayy
Epoch:  18 | Flops (T): 1.004663451744, Tokens (M) 1.000882, Train Loss 1.630, Val Loss 1.752 | Gen: foooooofwyyyyyyyyyyy
Epoch:  19 | Flops (T): 1.05754047552, Tokens (M) 1.05356, Train Loss 1.580, Val Loss 1.678 | Gen: foooooooorrrrrayyyyyy
Epoch:  20 | Flops (T): 1.110417499296, Tokens (M) 1.106238, Train Loss 1.537, Val Loss 1.620 | Gen: ffoorrrrrrrrrayyyyyy
Epoch:  21 | Flops (T): 1.163294523072, Tokens (M) 1.158916, Train Loss 1.496, Val Loss 1.576 | Gen: foofrrrrrrray
Epoch:  22 | Flops (T): 1.216171546848, Tokens (M) 1.211594, Train Loss 1.440, Val Loss 1.559 | Gen: fooofrrrrrrrayyyyyyy
Epoch:  23 | Flops (T): 1.269048570624, Tokens (M) 1.264272, Train Loss 1.439, Val Loss 1.571 | Gen: ffffrrrrayyyyyyyyyyy
Epoch:  24 | Flops (T): 1.3219255944, Tokens (M) 1.31695, Train Loss 1.450, Val Loss 1.949 | Gen: llllloorrrayyyyyyyyy
Epoch:  25 | Flops (T): 1.374802618176, Tokens (M) 1.369628, Train Loss 1.529, Val Loss 1.614 | Gen: ooooffffrrrrrrayyyyy
Epoch:  26 | Flops (T): 1.427679641952, Tokens (M) 1.422306, Train Loss 1.400, Val Loss 1.684 | Gen: fffffffayyyyyyyyyyy
Epoch:  27 | Flops (T): 1.480556665728, Tokens (M) 1.474984, Train Loss 1.370, Val Loss 1.488 | Gen: ooooofffaay
Epoch:  28 | Flops (T): 1.533433689504, Tokens (M) 1.527662, Train Loss 1.350, Val Loss 1.691 | Gen: ooooffffffffffffffff
Epoch:  29 | Flops (T): 1.58631071328, Tokens (M) 1.58034, Train Loss 1.357, Val Loss 1.453 | Gen: oooffffffay
Epoch:  30 | Flops (T): 1.639187737056, Tokens (M) 1.633018, Train Loss 1.254, Val Loss 1.469 | Gen: oooffaaaaaaaaaaaaaaa
Epoch:  31 | Flops (T): 1.692064760832, Tokens (M) 1.685696, Train Loss 1.358, Val Loss 1.363 | Gen: oooffffaaaaayyyyyyyy
Epoch:  32 | Flops (T): 1.744941784608, Tokens (M) 1.738374, Train Loss 1.234, Val Loss 1.406 | Gen: oooffffayyyyyyyyyyy
Epoch:  33 | Flops (T): 1.797818808384, Tokens (M) 1.791052, Train Loss 1.259, Val Loss 1.326 | Gen: llffffffaaaaaaaaaaa
Epoch:  34 | Flops (T): 1.85069583216, Tokens (M) 1.84373, Train Loss 1.142, Val Loss 1.283 | Gen: fffffffaaaaaaaaaaaaa
Epoch:  35 | Flops (T): 1.903572855936, Tokens (M) 1.896408, Train Loss 1.191, Val Loss 1.253 | Gen: fffffffaaaaaaaaaaaaa
Epoch:  36 | Flops (T): 1.956449879712, Tokens (M) 1.949086, Train Loss 1.120, Val Loss 1.375 | Gen: llofrraaaaaaaaaaaaaa
Epoch:  37 | Flops (T): 2.009326903488, Tokens (M) 2.001764, Train Loss 1.158,

Val Loss 1.286 | Gen: fffffffffaaaaaaaaaa
Epoch: 38 | Flops (T): 2.062203927264, Tokens (M) 2.054442, Train Loss 1.110, Val Loss 1.121 | Gen: looofffaaaaaaaaaaaa
Epoch: 39 | Flops (T): 2.11508095104, Tokens (M) 2.10712, Train Loss 1.021, Val Loss 1.235 | Gen: lloofffffaaaaaaaaaaa
Epoch: 40 | Flops (T): 2.167957974816, Tokens (M) 2.159798, Train Loss 1.011, Val Loss 1.156 | Gen: ffffffaaaaaaaaaaaaa
Epoch: 41 | Flops (T): 2.220834998592, Tokens (M) 2.212476, Train Loss 0.988, Val Loss 1.136 | Gen: ooooffffaaaaaaaaaaaa
Epoch: 42 | Flops (T): 2.273712022368, Tokens (M) 2.265154, Train Loss 0.935, Val Loss 1.032 | Gen: llofffffaaaaaaaaaaa
Epoch: 43 | Flops (T): 2.326589046144, Tokens (M) 2.317832, Train Loss 0.913, Val Loss 1.158 | Gen: lllffffffaaaaaaaaaaa
Epoch: 44 | Flops (T): 2.37946606992, Tokens (M) 2.37051, Train Loss 0.957, Val Loss 1.114 | Gen: oooffffffaaaaaaaaaaa
Epoch: 45 | Flops (T): 2.432343093696, Tokens (M) 2.423188, Train Loss 1.002, Val Loss 1.390 | Gen: llofffffrfrrrrayyyy
Epoch: 46 | Flops (T): 2.485220117472, Tokens (M) 2.475866, Train Loss 1.015, Val Loss 1.177 | Gen: oooffffrray
Epoch: 47 | Flops (T): 2.538097141248, Tokens (M) 2.528544, Train Loss 0.931, Val Loss 1.040 | Gen: ooooffffffffay
Epoch: 48 | Flops (T): 2.590974165024, Tokens (M) 2.581222, Train Loss 0.842, Val Loss 1.079 | Gen: looofffffffrrrayyyyyy
Epoch: 49 | Flops (T): 2.6438511888, Tokens (M) 2.6339, Train Loss 0.820, Val Loss 0.991 | Gen: foofffffffaaaaaaaaaa
Epoch: 50 | Flops (T): 2.696728212576, Tokens (M) 2.686578, Train Loss 0.805, Val Loss 1.017 | Gen: ooooffffffffay
Epoch: 51 | Flops (T): 2.749605236352, Tokens (M) 2.739256, Train Loss 0.846, Val Loss 1.035 | Gen: ooooffffffffrayyyyy
Epoch: 52 | Flops (T): 2.802482260128, Tokens (M) 2.791934, Train Loss 0.822, Val Loss 1.008 | Gen: looofffffayyyyyyyyy
Epoch: 53 | Flops (T): 2.855359283904, Tokens (M) 2.844612, Train Loss 0.790, Val Loss 1.019 | Gen: ffoofffffffffayyyyyy
Epoch: 54 | Flops (T): 2.90823630768, Tokens (M) 2.89729, Train Loss 0.771, Val Loss 0.976 | Gen: foofffffffffrayyyyyy
Epoch: 55 | Flops (T): 2.961113331456, Tokens (M) 2.949968, Train Loss 0.817, Val Loss 1.058 | Gen: oooffffffayyyyyyyyyy
Epoch: 56 | Flops (T): 3.013990355232, Tokens (M) 3.002646, Train Loss 0.758, Val Loss 1.019 | Gen: ooooffffffrayyyyyyyy
Epoch: 57 | Flops (T): 3.066867379008, Tokens (M) 3.055324, Train Loss 1.065, Val Loss 1.324 | Gen: ooooofffffffayyyyyyy
Epoch: 58 | Flops (T): 3.119744402784, Tokens (M) 3.108002, Train Loss 0.922, Val Loss 1.215 | Gen: oooofffffrrayyyyyyyy
Epoch: 59 | Flops (T): 3.17262142656, Tokens (M) 3.16068, Train Loss 0.916, Val Loss 1.152 | Gen: wwwwwwwwwwwwwwwwwwww
Epoch: 60 | Flops (T): 3.225498450336, Tokens (M) 3.213358, Train Loss 0.782, Val Loss 0.970 | Gen: oooffffffaaaaaaaaaaa
Epoch: 61 | Flops (T): 3.278375474112, Tokens (M) 3.266036, Train Loss 0.711,

```
Val Loss 0.943 | Gen: ooooffffffrrayyyyyyyy
Epoch:  62 | Flops (T): 3.331252497888, Tokens (M) 3.318714, Train Loss 0.697,
Val Loss 0.908 | Gen: ooooffffffrhayyyyyyy
Epoch:  63 | Flops (T): 3.384129521664, Tokens (M) 3.371392, Train Loss 0.689,
Val Loss 0.883 | Gen: ooooffffrrayyyyyyyyy
Epoch:  64 | Flops (T): 3.43700654544, Tokens (M) 3.42407, Train Loss 0.735, Val
Loss 0.924 | Gen: ooooffffffrhayyyyyyyy
Epoch:  65 | Flops (T): 3.489883569216, Tokens (M) 3.476748, Train Loss 0.729,
Val Loss 0.932 | Gen: ooooffffrrayyyyyyyyy
Epoch:  66 | Flops (T): 3.542760592992, Tokens (M) 3.529426, Train Loss 0.698,
Val Loss 0.988 | Gen: ooooffffrrrayyyyyyyy
Epoch:  67 | Flops (T): 3.595637616768, Tokens (M) 3.582104, Train Loss 0.660,
Val Loss 0.996 | Gen: oooooffffffrrdyyyyyyy
Epoch:  68 | Flops (T): 3.648514640544, Tokens (M) 3.634782, Train Loss 0.678,
Val Loss 0.867 | Gen: oooooffffrrayyyyyyyy
Epoch:  69 | Flops (T): 3.70139166432, Tokens (M) 3.68746, Train Loss 0.648, Val
Loss 0.938 | Gen: ooooffffffrrdyyyyyyyy
Epoch:  70 | Flops (T): 3.754268688096, Tokens (M) 3.740138, Train Loss 0.626,
Val Loss 0.924 | Gen: oooofffrrayyyyyyyyyy
Epoch:  71 | Flops (T): 3.807145711872, Tokens (M) 3.792816, Train Loss 0.638,
Val Loss 1.066 | Gen: oooooffffffrrrayyyyy
Epoch:  72 | Flops (T): 3.860022735648, Tokens (M) 3.845494, Train Loss 0.642,
Val Loss 0.864 | Gen: oooofffffrdyyyyyyyy
Epoch:  73 | Flops (T): 3.912899759424, Tokens (M) 3.898172, Train Loss 0.603,
Val Loss 0.849 | Gen: oooofffrrrayyyyyyyy
Epoch:  74 | Flops (T): 3.9657767832, Tokens (M) 3.95085, Train Loss 0.595, Val
Loss 0.960 | Gen: ooooorrrrrrrrrrrrrr
Epoch:  75 | Flops (T): 4.018653806976, Tokens (M) 4.003528, Train Loss 0.612,
Val Loss 0.915 | Gen: oooooofffrrrayyyyyyyy
Epoch:  76 | Flops (T): 4.071530830752, Tokens (M) 4.056206, Train Loss 0.586,
Val Loss 0.773 | Gen: ooooffffffrhay
Epoch:  77 | Flops (T): 4.124407854528, Tokens (M) 4.108884, Train Loss 0.573,
Val Loss 0.738 | Gen: oooooofffrayyyyyyyyy
Epoch:  78 | Flops (T): 4.177284878304, Tokens (M) 4.161562, Train Loss 0.551,
Val Loss 0.850 | Gen: oooofffrrrrrrrrrrrrr
Epoch:  79 | Flops (T): 4.23016190208, Tokens (M) 4.21424, Train Loss 0.530, Val
Loss 0.890 | Gen: oooooofffrrayyyyyyyy
Epoch:  80 | Flops (T): 4.283038925856, Tokens (M) 4.266918, Train Loss 0.544,
Val Loss 0.990 | Gen: oooooffffffrayyyyyyyy
Epoch:  81 | Flops (T): 4.335915949632, Tokens (M) 4.319596, Train Loss 0.536,
Val Loss 0.935 | Gen: oooofffrraayyyyyyyyy
Epoch:  82 | Flops (T): 4.388792973408, Tokens (M) 4.372274, Train Loss 0.532,
Val Loss 0.885 | Gen: oooooffffffrdyyyyyyy
Epoch:  83 | Flops (T): 4.441669997184, Tokens (M) 4.424952, Train Loss 0.507,
Val Loss 0.852 | Gen: oooofffffrraayyyyyyyy
Epoch:  84 | Flops (T): 4.49454702096, Tokens (M) 4.47763, Train Loss 0.492, Val
Loss 0.949 | Gen: oooooofffrrrayyyyyyyy
Epoch:  85 | Flops (T): 4.547424044736, Tokens (M) 4.530308, Train Loss 0.497,
```

```
Val Loss 0.929 | Gen: ooooofffrrayyyyyyyy
Epoch:  86 | Flops (T): 4.600301068512, Tokens (M) 4.582986, Train Loss 0.497,
Val Loss 0.820 | Gen: ooooofffraayyyyyyyy
Epoch:  87 | Flops (T): 4.653178092288, Tokens (M) 4.635664, Train Loss 0.557,
Val Loss 1.147 | Gen: ooooofffffrayyyyyyyy
Epoch:  88 | Flops (T): 4.706055116064, Tokens (M) 4.688342, Train Loss 0.580,
Val Loss 1.083 | Gen: ooooofffrrdddddddddd
Epoch:  89 | Flops (T): 4.75893213984, Tokens (M) 4.74102, Train Loss 0.624, Val
Loss 0.770 | Gen: ooooofffraayyyyyyyy
Epoch:  90 | Flops (T): 4.811809163616, Tokens (M) 4.793698, Train Loss 0.476,
Val Loss 0.634 | Gen: ooooofffraayyyyyyyy
Epoch:  91 | Flops (T): 4.864686187392, Tokens (M) 4.846376, Train Loss 0.435,
Val Loss 0.639 | Gen: ooooofrraayyyyyyyy
Epoch:  92 | Flops (T): 4.917563211168, Tokens (M) 4.899054, Train Loss 0.411,
Val Loss 0.622 | Gen: oooofffrraayyyyyyyy
Epoch:  93 | Flops (T): 4.970440234944, Tokens (M) 4.951732, Train Loss 0.403,
Val Loss 0.686 | Gen: oooofffrraaayyyyyyyy
Epoch:  94 | Flops (T): 5.02331725872, Tokens (M) 5.00441, Train Loss 0.405, Val
Loss 0.710 | Gen: ooooofffraayyyyyyyy
Epoch:  95 | Flops (T): 5.076194282496, Tokens (M) 5.057088, Train Loss 0.404,
Val Loss 0.958 | Gen: ooooofrrraayyyyyyyy
Epoch:  96 | Flops (T): 5.129071306272, Tokens (M) 5.109766, Train Loss 0.474,
Val Loss 0.677 | Gen: oooofffrraaaaaaaaa
Epoch:  97 | Flops (T): 5.181948330048, Tokens (M) 5.162444, Train Loss 0.410,
Val Loss 0.659 | Gen: oooorffraaayyyyyyyy
Epoch:  98 | Flops (T): 5.234825353824, Tokens (M) 5.215122, Train Loss 0.377,
Val Loss 0.641 | Gen: oooffrraaayyyyyyyy
Epoch:  99 | Flops (T): 5.2877023776, Tokens (M) 5.2678, Train Loss 0.369, Val
Loss 0.815 | Gen: oooffrraayyyyyyyyy
Obtained lowest validation loss of: 0.6218264688586079
Avg tokens per example (Seq Len): 20.59343236903831, Avg flops per example:
20671236.816262703
```

[ ]:
```
TEST_SENTENCE = "floor"
trans128_args_s = AttrDict()
args_dict_128 = {
    "data_file_name": "pig_latin_small", #pig_latin_small debug
    "cuda": use_cuda,
    "nepochs": 100,
    "checkpoint_dir": "checkpoints",
    "learning_rate": 5e-4,
    "early_stopping_patience": 100,
    "lr_decay": 0.99,
    "batch_size": 64,
    "hidden_size": 128,
    "num_transformer_layers": 4,
    "loss_seq_span": "target_only",
```

```
    "num_layers": 4,
    "vocab_size": 30
}
trans128_args_s.update(args_dict_128)
print_opts(trans128_args_s)
trans128_decoder_s, trans128_losses_s, idx_dict = train(trans128_args_s)
```

```
================================================================================
                                       Opts
--------------------------------------------------------------------------------
                         data_file_name: pig_latin_small
                                   cuda: 1
                               nepochs: 100
                         checkpoint_dir: checkpoints
                          learning_rate: 0.0005
                early_stopping_patience: 100
                                lr_decay: 0.99
                             batch_size: 64
                            hidden_size: 128
                 num_transformer_layers: 4
                          loss_seq_span: target_only
                             num_layers: 4
                             vocab_size: 30
================================================================================
================================================================================
                                     Data Stats
--------------------------------------------------------------------------------
('irresistible', 'irresistibleway')
('repentance', 'epentanceray')
('pretty', 'ettypray')
('parcels', 'arcelspay')
('survey', 'urveysay')
Num unique word pairs: 3198
Vocabulary: dict_keys(['-', 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j',
'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z',
'SOS', 'EOS', 'EOP'])
Vocab size: 30
================================================================================
defaultdict(<class 'list'>, {(12, 15): [('irresistible', 'irresistibleway'),
('advantageous', 'advantageousway'), ('impartiality', 'impartialityway'),
('interruption', 'interruptionway'), ('anticipation', 'anticipationway'),
('impenetrable', 'impenetrableway'), ('excellencies', 'excellenciesway'),
('intermission', 'intermissionway'), ('interspersed', 'interspersedway'),
('unaffectedly', 'unaffectedlyway'), ('insinuations', 'insinuationsway'),
('unpardonable', 'unpardonableway'), ('arrangements', 'arrangementsway'),
('unexpectedly', 'unexpectedlyway'), ('introduction', 'introductionway'),
('occasionally', 'occasionallyway'), ('establishing', 'establishingway'),
```

```
('unanswerable', 'unanswerableway'), ('independence', 'independenceway'),
('inclinations', 'inclinationsway'), ('instructions', 'instructionsway'),
('undoubtingly', 'undoubtinglyway'), ('unfathomable', 'unfathomableway'),
('impoverished', 'impoverishedway'), ('outstretched', 'outstretchedway'),
('unrestrained', 'unrestrainedway'), ('ungraciously', 'ungraciouslyway'),
('imperfection', 'imperfectionway'), ('unreasonably', 'unreasonablyway'),
('interference', 'interferenceway'), ('inconvenient', 'inconvenientway'),
('enthusiastic', 'enthusiasticway'), ('extinguished', 'extinguishedway'),
('entanglement', 'entanglementway'), ('affectionate', 'affectionateway')], (10,
12): [('repentance', 'epentanceray'), ('capability', 'apabilitycay'),
('beneficial', 'eneficialbay'), ('retrenched', 'etrenchedray'), ('devonshire',
'evonshireday'), ('palanquins', 'alanquinspay'), ('contribute', 'ontributecay'),
('minuteness', 'inutenessmay'), ('dissimilar', 'issimilarday'), ('flattering',
'atteringflay'), ('confounded', 'onfoundedcay'), ('compromise', 'ompromisecay'),
('contrasted', 'ontrastedcay'), ('precaution', 'ecautionpray'), ('fastidious',
'astidiousfay'), ('blackguard', 'ackguardblay'), ('heroically', 'eroicallyhay'),
('lengthened', 'engthenedlay'), ('pocketbook', 'ocketbookpay'), ('companions',
'ompanionscay'), ('displeased', 'ispleasedday'), ('pianoforte', 'ianofortepay'),
('succession', 'uccessionsay'), ('composedly', 'omposedlycay'), ('pronounced',
'onouncedpray'), ('honourable', 'onourablehay'), ('solicitude', 'olicitudesay'),
('justifying', 'ustifyingjay'), ('collecting', 'ollectingcay'), ('headstrong',
'eadstronghay'), ('forbidding', 'orbiddingfay'), ('successful', 'uccessfulsay'),
('confidence', 'onfidencecay'), ('prescribed', 'escribedpray'), ('scrutinies',
'utiniesscray'), ('melancholy', 'elancholymay'), ('handsomest', 'andsomesthay'),
('prescience', 'esciencepray'), ('conclusion', 'onclusioncay'), ('remarkably',
'emarkablyray'), ('gratefully', 'atefullygray'), ('michaelmas', 'ichaelmasmay'),
('containing', 'ontainingcay'), ('discussion', 'iscussionday'), ('resentment',
'esentmentray'), ('restraints', 'estraintsray'), ('tablecloth', 'ableclothtay'),
('projecting', 'ojectingpray'), ('frightened', 'ightenedfray'), ('pardonable',
'ardonablepay'), ('variations', 'ariationsvay'), ('sensations', 'ensationssay'),
('rheumatism', 'eumatismrhay'), ('dispirited', 'ispiritedday'), ('politeness',
'olitenesspay'), ('suggesting', 'uggestingsay'), ('willoughby', 'illoughbyway'),
('reasonable', 'easonableray'), ('discerning', 'iscerningday'), ('previously',
'eviouslypray'), ('passionate', 'assionatepay'), ('prophecies', 'opheciespray'),
('possessing', 'ossessingpay'), ('perfection', 'erfectionpay'), ('heightened',
'eightenedhay'), ('generously', 'enerouslygay'), ('faithfully', 'aithfullyfay'),
('reconciled', 'econciledray'), ('distressed', 'istressedday'), ('respective',
'espectiveray'), ('liberality', 'iberalitylay'), ('disengaged', 'isengagedday'),
('honourably', 'onourablyhay'), ('persuading', 'ersuadingpay'), ('cultivated',
'ultivatedcay'), ('projection', 'ojectionpray'), ('disapprove', 'isapproveday'),
('conciliate', 'onciliatecay'), ('satisfying', 'atisfyingsay'), ('dispatches',
'ispatchesday'), ('repeatedly', 'epeatedlyray'), ('suspicions', 'uspicionssay'),
('perplexity', 'erplexitypay'), ('strengthen', 'engthenstray'), ('dependence',
'ependenceday'), ('dissipated', 'issipatedday'), ('conformity', 'onformitycay'),
('hysterical', 'ystericalhay'), ('throughout', 'oughoutthray'), ('sacrificed',
'acrificedsay'), ('dimensions', 'imensionsday'), ('flatteries', 'atteriesflay'),
('completely', 'ompletelycay'), ('reproaches', 'eproachesray'), ('contriving',
'ontrivingcay'), ('consequent', 'onsequentcay'), ('liveliness', 'ivelinesslay'),
```

('resistance', 'esistanceray'), ('caricature', 'aricaturecay'), ('stimulated',
'imulatedstay'), ('brightened', 'ightenedbray'), ('prevailing', 'evailingpray'),
('disclosure', 'isclosureday'), ('bequeathed', 'equeathedbay'), ('disappoint',
'isappointday'), ('complexion', 'omplexioncay'), ('dismounted', 'ismountedday'),
('remembered', 'ememberedray'), ('delineated', 'elineatedday'), ('subsequent',
'ubsequentsay'), ('persevered', 'erseveredpay'), ('doubtingly', 'oubtinglyday'),
('characters', 'aracterschay'), ('neighbours', 'eighboursnay'), ('witticisms',
'itticismsway'), ('materially', 'ateriallymay'), ('playthings', 'aythingsplay'),
('suspecting', 'uspectingsay'), ('conditions', 'onditionscay'), ('discovered',
'iscoveredday'), ('deficiency', 'eficiencyday'), ('regularity', 'egularityray'),
('subjection', 'ubjectionsay'), ('brightness', 'ightnessbray'), ('pleasantly',
'easantlyplay'), ('delicacies', 'elicaciesday'), ('preference', 'eferencepray'),
('difference', 'ifferenceday'), ('compassion', 'ompassioncay'), ('competence',
'ompetencecay'), ('generosity', 'enerositygay'), ('retirement', 'etirementray'),
('strikingly', 'ikinglystray'), ('subsisting', 'ubsistingsay'), ('conscience',
'onsciencecay'), ('diabolical', 'iabolicalday'), ('temptation', 'emptationtay'),
('formidable', 'ormidablefay'), ('reconsider', 'econsiderray'), ('prejudiced',
'ejudicedpray'), ('comprehend', 'omprehendcay'), ('compliment', 'omplimentcay'),
('sufferings', 'ufferingssay'), ('concluding', 'oncludingcay'), ('compliance',
'ompliancecay'), ('mosquitoes', 'osquitoesmay'), ('thoughtful', 'oughtfulthay'),
('bitterness', 'itternessbay'), ('mentioning', 'entioningmay'), ('toleration',
'olerationtay'), ('profession', 'ofessionpray'), ('discompose', 'iscomposeday'),
('waistcoats', 'aistcoatsway'), ('reflecting', 'eflectingray'), ('positively',
'ositivelypay'), ('diminished', 'iminishedday')], (6, 8): [('pretty',
'ettypray'), ('survey', 'urveysay'), ('demure', 'emureday'), ('secure',
'ecuresay'), ('sobbed', 'obbedsay'), ('moment', 'omentmay'), ('guilty',
'uiltygay'), ('combat', 'ombatcay'), ('modern', 'odernmay'), ('father',
'atherfay'), ('horses', 'orseshay'), ('closer', 'oserclay'), ('nicety',
'icetynay'), ('shaken', 'akenshay'), ('centre', 'entrecay'), ('powers',
'owerspay'), ('simple', 'implesay'), ('beings', 'eingsbay'), ('recent',
'ecentray'), ('sketch', 'etchskay'), ('loving', 'ovinglay'), ('dagger',
'aggerday'), ('hearts', 'eartshay'), ('copied', 'opiedcay'), ('gazing',
'azinggay'), ('sooner', 'oonersay'), ('disarm', 'isarmday'), ('suited',
'uitedsay'), ('doubts', 'oubtsday'), ('lifted', 'iftedlay'), ('vacant',
'acantvay'), ('palmer', 'almerpay'), ('grudge', 'udgegray'), ('folded',
'oldedfay'), ('remove', 'emoveray'), ('borrow', 'orrowbay'), ('scheme',
'emeschay'), ('solemn', 'olemnsay'), ('denote', 'enoteday'), ('bustle',
'ustlebay'), ('turned', 'urnedtay'), ('mutton', 'uttonmay'), ('tender',
'endertay'), ('hearer', 'earerhay'), ('begins', 'eginsbay'), ('taught',
'aughttay'), ('honour', 'onourhay'), ('robert', 'obertray'), ('gibson',
'ibsongay'), ('virtue', 'irtuevay'), ('vanish', 'anishvay'), ('writes',
'iteswray'), ('silent', 'ilentsay'), ('singly', 'inglysay'), ('jargon',
'argonjay'), ('grieve', 'ievegray'), ('design', 'esignday'), ('linger',
'ingerlay'), ('bathed', 'athedbay'), ('pardon', 'ardonpay'), ('rumour',
'umourray'), ('dances', 'ancesday'), ('remark', 'emarkray'), ('worded',
'ordedway'), ('honest', 'onesthay'), ('chosen', 'osenchay'), ('boyish',
'oyishbay'), ('stress', 'essstray'), ('caught', 'aughtcay'), ('potent',
'otentpay'), ('during', 'uringday'), ('stocks', 'ocksstay'), ('relics',

'elicsray'), ('tongue', 'onguetay'), ('street', 'eetstray'), ('whiled',
'iledwhay'), ('manner', 'annermay'), ('truths', 'uthstray'), ('travel',
'aveltray'), ('couple', 'ouplecay'), ('gaiety', 'aietygay'), ('giving',
'ivinggay'), ('county', 'ountycay'), ('cramps', 'ampscray'), ('wealth',
'ealthway'), ('divine', 'ivineday'), ('behold', 'eholdbay'), ('feared',
'earedfay'), ('sparks', 'arksspay'), ('finger', 'ingerfay'), ('talent',
'alenttay'), ('demand', 'emandday'), ('coffee', 'offeecay'), ('merest',
'erestmay'), ('detail', 'etailday'), ('casino', 'asinocay'), ('choose',
'oosechay'), ('warmed', 'armedway'), ('relate', 'elateray'), ('sashes',
'ashessay'), ('chance', 'ancechay'), ('rushed', 'ushedray'), ('health',
'ealthhay'), ('preyed', 'eyedpray'), ('mutual', 'utualmay'), ('months',
'onthsmay'), ('raises', 'aisesray'), ('things', 'ingsthay'), ('master',
'astermay'), ('thanks', 'anksthay'), ('cheeks', 'eekschay'), ('hazard',
'azardhay'), ('slight', 'ightslay'), ('guests', 'uestsgay'), ('halves',
'alveshay'), ('lessen', 'essenlay'), ('notice', 'oticenay'), ('higher',
'igherhay'), ('truest', 'uesttray'), ('porter', 'orterpay'), ('fickle',
'icklefay'), ('behave', 'ehavebay'), ('credit', 'editcray'), ('number',
'umbernay'), ('mother', 'othermay'), ('tricks', 'ickstray'), ('traced',
'acedtray'), ('subdue', 'ubduesay'), ('ruined', 'uinedray'), ('famous',
'amousfay'), ('danced', 'ancedday'), ('clever', 'everclay'), ('drives',
'ivesdray'), ('resist', 'esistray'), ('forget', 'orgetfay'), ('stable',
'ablestay'), ('bigger', 'iggerbay'), ('betray', 'etraybay'), ('senses',
'ensessay'), ('smiled', 'iledsmay'), ('hating', 'atinghay'), ('bolder',
'olderbay'), ('banish', 'anishbay'), ('remain', 'emainray'), ('period',
'eriodpay'), ('gossip', 'ossipgay'), ('basket', 'asketbay'), ('genius',
'eniusgay'), ('seized', 'eizedsay'), ('widely', 'idelyway'), ('pursue',
'ursuepay'), ('favour', 'avourfay'), ('sighed', 'ighedsay'), ('plants',
'antsplay'), ('proved', 'ovedpray'), ('season', 'easonsay'), ('circle',
'irclecay'), ('taking', 'akingtay'), ('screen', 'eenscray'), ('talker',
'alkertay'), ('teazed', 'eazedtay'), ('suffer', 'uffersay'), ('curate',
'uratecay'), ('traits', 'aitstray'), ('defect', 'efectday'), ('called',
'alledcay'), ('dining', 'iningday'), ('direct', 'irectday'), ('dinner',
'innerday'), ('joking', 'okingjay'), ('whiter', 'iterwhay'), ('limits',
'imitslay'), ('papers', 'aperspay'), ('public', 'ublicpay'), ('losing',
'osinglay'), ('handed', 'andedhay'), ('wisest', 'isestway'), ('delays',
'elaysday'), ('leaves', 'eaveslay'), ('points', 'ointspay'), ('hither',
'itherhay'), ('horrid', 'orridhay'), ('middle', 'iddlemay'), ('vastly',
'astlyvay'), ('luxury', 'uxurylay'), ('failed', 'ailedfay'), ('lodges',
'odgeslay'), ('female', 'emalefay'), ('walked', 'alkedway'), ('nipped',
'ippednay'), ('nephew', 'ephewnay'), ('settle', 'ettlesay'), ('kinder',
'inderkay'), ('bowing', 'owingbay'), ('smooth', 'oothsmay'), ('roared',
'oaredray'), ('wiping', 'ipingway'), ('jewels', 'ewelsjay'), ('breath',
'eathbray'), ('steady', 'eadystay'), ('breach', 'eachbray'), ('polite',
'olitepay'), ('colder', 'oldercay'), ('though', 'oughthay'), ('poured',
'ouredpay'), ('paused', 'ausedpay'), ('nought', 'oughtnay'), ('shrewd',
'ewdshray'), ('fourth', 'ourthfay'), ('parish', 'arishpay'), ('traded',
'adedtray'), ('glance', 'anceglay'), ('lesser', 'esserlay'), ('curacy',
'uracycay'), ('labour', 'abourlay'), ('became', 'ecamebay'), ('decide',

'ecideday'), ('kisses', 'isseskay'), ('styled', 'yledstay'), ('sprung',
'ungspray'), ('priory', 'iorypray'), ('derive', 'eriveday'), ('school',
'oolschay'), ('liking', 'ikinglay'), ('dawned', 'awnedday'), ('weaken',
'eakenway'), ('winter', 'interway'), ('rivals', 'ivalsray'), ('sallow',
'allowsay'), ('defend', 'efendday'), ('seated', 'eatedsay'), ('penury',
'enurypay'), ('repair', 'epairray'), ('chaise', 'aisechay'), ('church',
'urchchay'), ('beyond', 'eyondbay'), ('wisdom', 'isdomway'), ('retain',
'etainray'), ('toilet', 'oilettay'), ('talked', 'alkedtay'), ('dearly',
'earlyday'), ('reward', 'ewardray'), ('safely', 'afelysay'), ('making',
'akingmay'), ('debate', 'ebateday'), ('future', 'uturefay'), ('broken',
'okenbray'), ('begged', 'eggedbay'), ('thirty', 'irtythay'), ('muslin',
'uslinmay'), ('marked', 'arkedmay'), ('hinder', 'inderhay'), ('beheld',
'eheldbay'), ('maxims', 'aximsmay'), ('theirs', 'eirsthay'), ('corner',
'ornercay'), ('piqued', 'iquedpay'), ('rested', 'estedray'), ('needed',
'eedednay'), ('fitted', 'ittedfay'), ('candid', 'andidcay'), ('laughs',
'aughslay'), ('window', 'indowway'), ('tastes', 'astestay'), ('spread',
'eadspray'), ('hollow', 'ollowhay'), ('sorrow', 'orrowsay'), ('partly',
'artlypay'), ('served', 'ervedsay'), ('flowed', 'owedflay'), ('remedy',
'emedyray'), ('trying', 'yingtray'), ('latest', 'atestlay'), ('manage',
'anagemay'), ('gained', 'ainedgay'), ('plenty', 'entyplay'), ('garden',
'ardengay'), ('frosts', 'ostsfray'), ('thirds', 'irdsthay'), ('handle',
'andlehay'), ('verbal', 'erbalvay'), ('sheath', 'eathshay'), ('covert',
'overtcay'), ('darker', 'arkerday'), ('harris', 'arrishay'), ('houses',
'ouseshay'), ('nature', 'aturenay'), ('thomas', 'omasthay'), ('submit',
'ubmitsay'), ('danger', 'angerday'), ('lawyer', 'awyerlay'), ('matter',
'attermay'), ('rugged', 'uggedray'), ('highly', 'ighlyhay'), ('whence',
'encewhay'), ('martha', 'arthamay'), ('hearty', 'eartyhay'), ('recall',
'ecallray'), ('raised', 'aisedray'), ('follow', 'ollowfay'), ('nearer',
'earernay'), ('gentle', 'entlegay'), ('pitied', 'itiedpay'), ('nurses',
'ursesnay'), ('punish', 'unishpay')], (7, 9): [('parcels', 'arcelspay'),
('welcome', 'elcomeway'), ('posture', 'osturepay'), ('mothers', 'othersmay'),
('tempers', 'emperstay'), ('carries', 'arriescay'), ('neglect', 'eglectnay'),
('prevail', 'evailpray'), ('declare', 'eclareday'), ('crowned', 'ownedcray'),
('mindful', 'indfulmay'), ('blessed', 'essedblay'), ('kitchen', 'itchenkay'),
('younger', 'youngeray'), ('hanover', 'anoverhay'), ('lowered', 'oweredlay'),
('summons', 'ummonssay'), ('suffice', 'ufficesay'), ('rousing', 'ousingray'),
('feeling', 'eelingfay'), ('parlors', 'arlorspay'), ('gravely', 'avelygray'),
('connect', 'onnectcay'), ('species', 'eciesspay'), ('revived', 'evivedray'),
('tribute', 'ibutetray'), ('pitying', 'ityingpay'), ('brother', 'otherbray'),
('retired', 'etiredray'), ('suffers', 'ufferssay'), ('spurned', 'urnedspay'),
('dropped', 'oppeddray'), ('galling', 'allinggay'), ('seducer', 'educersay'),
('foibles', 'oiblesfay'), ('tuesday', 'uesdaytay'), ('patient', 'atientpay'),
('worsted', 'orstedway'), ('trifled', 'ifledtray'), ('waiting', 'aitingway'),
('widower', 'idowerway'), ('hardily', 'ardilyhay'), ('glimpse', 'impseglay'),
('whether', 'etherwhay'), ('reverie', 'everieray'), ('resolve', 'esolveray'),
('propose', 'oposepray'), ('tallest', 'allesttay'), ('nominal', 'ominalnay'),
('languid', 'anguidlay'), ('wronged', 'ongedwray'), ('nothing', 'othingnay'),
('promote', 'omotepray'), ('novelty', 'oveltynay'), ('dooming', 'oomingday'),

```
('display', 'isplayday'), ('correct', 'orrectcay'), ('feigned', 'eignedfay'),
('schemes', 'emesschay'), ('deepest', 'eepestday'), ('flannel', 'annelflay'),
('hauteur', 'auteurhay'), ('sharing', 'aringshay'), ('getting', 'ettinggay'),
('screams', 'eamsscray'), ('variety', 'arietyvay'), ('yielded', 'yieldeday'),
('brought', 'oughtbray'), ('haughty', 'aughtyhay'), ('bristol', 'istolbray'),
('silence', 'ilencesay'), ('staying', 'ayingstay'), ('confess', 'onfesscay'),
('placing', 'acingplay'), ('noisier', 'oisiernay'), ('nursing', 'ursingnay'),
('tuition', 'uitiontay'), ('welfare', 'elfareway'), ('hunters', 'untershay'),
('poplars', 'oplarspay'), ('deprive', 'epriveday'), ('mansion', 'ansionmay'),
('reasons', 'easonsray'), ('conceit', 'onceitcay'), ('burgess', 'urgessbay'),
('receipt', 'eceiptray'), ('befalls', 'efallsbay'), ('fetches', 'etchesfay'),
('ranging', 'angingray'), ('passion', 'assionpay'), ('smirked', 'irkedsmay'),
('honours', 'onourshay'), ('covered', 'overedcay'), ('dancing', 'ancingday'),
('hundred', 'undredhay'), ('forgave', 'orgavefay'), ('fifteen', 'ifteenfay'),
('lasting', 'astinglay'), ('minuter', 'inutermay'), ('tremour', 'emourtray'),
('passing', 'assingpay'), ('streets', 'eetsstray'), ('resists', 'esistsray'),
('cleared', 'earedclay'), ('miracle', 'iraclemay'), ('several', 'everalsay'),
('sighing', 'ighingsay'), ('resumed', 'esumedray'), ('merited', 'eritedmay'),
('without', 'ithoutway'), ('removed', 'emovedray'), ('behaved', 'ehavedbay'),
('succour', 'uccoursay'), ('bribing', 'ibingbray'), ('charity', 'aritychay'),
('shortly', 'ortlyshay'), ('tempted', 'emptedtay'), ('briefly', 'ieflybray'),
('parties', 'artiespay'), ('touched', 'ouchedtay'), ('content', 'ontentcay'),
('proceed', 'oceedpray'), ('windows', 'indowsway'), ('ribbons', 'ibbonsray'),
('stepped', 'eppedstay'), ('pursuit', 'ursuitpay'), ('quarter', 'arterquay'),
('prudent', 'udentpray'), ('dignity', 'ignityday'), ('payment', 'aymentpay'),
('grounds', 'oundsgray'), ('bequest', 'equestbay'), ('donavan', 'onavanday'),
('renewed', 'enewedray'), ('smaller', 'allersmay'), ('fearful', 'earfulfay'),
('civilly', 'ivillycay'), ('romance', 'omanceray'), ('females', 'emalesfay'),
('grieved', 'ievedgray'), ('created', 'eatedcray'), ('command', 'ommandcay'),
('defined', 'efinedday'), ('bluntly', 'untlyblay'), ('recital', 'ecitalray'),
('sorrows', 'orrowssay'), ('whither', 'itherwhay'), ('spotted', 'ottedspay'),
('justify', 'ustifyjay'), ('conduit', 'onduitcay'), ('revenge', 'evengeray'),
('scenery', 'eneryscay'), ('started', 'artedstay'), ('degrees', 'egreesday'),
('whisper', 'isperwhay'), ('strange', 'angestray'), ('bidding', 'iddingbay'),
('musical', 'usicalmay'), ('happens', 'appenshay'), ('compare', 'omparecay'),
('presume', 'esumepray'), ('sitting', 'ittingsay'), ('regards', 'egardsray'),
('violins', 'iolinsvay'), ('rightly', 'ightlyray'), ('memento', 'ementomay'),
('courted', 'ourtedcay'), ('servant', 'ervantsay'), ('possess', 'ossesspay'),
('herself', 'erselfhay'), ('present', 'esentpray'), ('natural', 'aturalnay'),
('managed', 'anagedmay'), ('hostess', 'ostesshay'), ('leaving', 'eavinglay'),
('mention', 'entionmay'), ('circuit', 'ircuitcay'), ('counsel', 'ounselcay'),
('guineas', 'uineasgay'), ('solidly', 'olidlysay'), ('dashing', 'ashingday'),
('station', 'ationstay'), ('written', 'ittenwray'), ('writing', 'itingwray'),
('country', 'ountrycay'), ('delight', 'elightday'), ('westons', 'estonsway'),
('debated', 'ebatedday'), ('lengths', 'engthslay'), ('torment', 'ormenttay'),
('reverse', 'everseray'), ('bedroom', 'edroombay'), ('forbade', 'orbadefay'),
('comings', 'omingscay'), ('tonight', 'onighttay'), ('similar', 'imilarsay'),
('calling', 'allingcay'), ('cassino', 'assinocay'), ('strains', 'ainsstray'),
```

('scratch', 'atchscray'), ('gardens', 'ardensgay'), ('freedom', 'eedomfray'),
('compact', 'ompactcay'), ('message', 'essagemay'), ('heavily', 'eavilyhay'),
('manners', 'annersmay'), ('systems', 'ystemssay'), ('forbore', 'orborefay'),
('wrapped', 'appedwray'), ('defects', 'efectsday'), ('villain', 'illainvay'),
('studied', 'udiedstay'), ('playing', 'ayingplay'), ('diction', 'ictionday'),
('tending', 'endingtay'), ('seeking', 'eekingsay'), ('rapture', 'aptureray'),
('dawdled', 'awdledday'), ('wearing', 'earingway'), ('comfort', 'omfortcay'),
('finally', 'inallyfay'), ('circles', 'irclescay'), ('putting', 'uttingpay'),
('twisted', 'istedtway'), ('destroy', 'estroyday'), ('whereas', 'ereaswhay'),
('dismiss', 'ismissday'), ('visibly', 'isiblyvay'), ('talents', 'alentstay'),
('finding', 'indingfay'), ('removes', 'emovesray'), ('replied', 'epliedray'),
('bluster', 'usterblay'), ('raising', 'aisingray'), ('plainly', 'ainlyplay'),
('worried', 'orriedway'), ('meaning', 'eaningmay'), ('blushes', 'ushesblay'),
('relapse', 'elapseray'), ('careful', 'arefulcay'), ('beloved', 'elovedbay'),
('degrade', 'egradeday'), ('foresaw', 'oresawfay'), ('procure', 'ocurepray'),
('kindest', 'indestkay'), ('secures', 'ecuressay'), ('hanging', 'anginghay'),
('persons', 'ersonspay'), ('taverns', 'avernstay'), ('walking', 'alkingway'),
('quicker', 'ickerquay'), ('guessed', 'uessedgay'), ('confirm', 'onfirmcay'),
('venture', 'enturevay'), ('project', 'ojectpray'), ('prayers', 'ayerspray'),
('heights', 'eightshay'), ('subject', 'ubjectsay'), ('summits', 'ummitssay'),
('towards', 'owardstay'), ('closing', 'osingclay'), ('journey', 'ourneyjay'),
('pointer', 'ointerpay'), ('chained', 'ainedchay'), ('clothes', 'othesclay'),
('require', 'equireray'), ('chusing', 'usingchay'), ('reproof', 'eproofray'),
('reckons', 'eckonsray'), ('pressed', 'essedpray'), ('talking', 'alkingtay'),
('harbour', 'arbourhay'), ('garrets', 'arretsgay'), ('foresee', 'oreseefay'),
('lustily', 'ustilylay'), ('leagued', 'eaguedlay'), ('judging', 'udgingjay'),
('rejoice', 'ejoiceray'), ('honiton', 'onitonhay'), ('shorten', 'ortenshay'),
('stating', 'atingstay'), ('privacy', 'ivacypray'), ('steeles', 'eelesstay'),
('thickly', 'icklythay'), ('shocked', 'ockedshay'), ('limited', 'imitedlay'),
('blights', 'ightsblay'), ('seclude', 'ecludesay'), ('cordial', 'ordialcay'),
('request', 'equestray'), ('denoted', 'enotedday'), ('darling', 'arlingday'),
('coxcomb', 'oxcombcay'), ('however', 'oweverhay'), ('perform', 'erformpay'),
('society', 'ocietysay'), ('deserts', 'esertsday'), ('palmers', 'almerspay'),
('manager', 'anagermay'), ('rupture', 'uptureray'), ('concise', 'oncisecay'),
('scolded', 'oldedscay'), ('seduced', 'educedsay'), ('doubted', 'oubtedday'),
('forward', 'orwardfay'), ('leisure', 'eisurelay'), ('robbing', 'obbingray'),
('horizon', 'orizonhay'), ('blunder', 'underblay'), ('flushed', 'ushedflay'),
('divided', 'ividedday'), ('charged', 'argedchay'), ('disturb', 'isturbday'),
('smitten', 'ittensmay'), ('seconds', 'econdssay'), ('letters', 'etterslay'),
('provide', 'ovidepray'), ('conduct', 'onductcay'), ('plagued', 'aguedplay'),
('stretch', 'etchstray'), ('hoarded', 'oardedhay'), ('blushed', 'ushedblay'),
('nodding', 'oddingnay'), ('mankind', 'ankindmay'), ('caprice', 'apricecay'),
('quitted', 'ittedquay')], (5, 7): [('china', 'inachay'), ('deter', 'eterday'),
('woman', 'omanway'), ('clock', 'ockclay'), ('borne', 'ornebay'), ('noise',
'oisenay'), ('midst', 'idstmay'), ('dwell', 'elldway'), ('smith', 'ithsmay'),
('gowns', 'ownsgay'), ('named', 'amednay'), ('glass', 'assglay'), ('price',
'icepray'), ('close', 'oseclay'), ('guide', 'uidegay'), ('newly', 'ewlynay'),
('sweep', 'eepsway'), ('swept', 'eptsway'), ('those', 'osethay'), ('lower',

'owerlay'), ('plump', 'umpplay'), ('downs', 'ownsday'), ('purse', 'ursepay'),
('linen', 'inenlay'), ('later', 'aterlay'), ('broad', 'oadbray'), ('miles',
'ilesmay'), ('facts', 'actsfay'), ('sleep', 'eepslay'), ('steps', 'epsstay'),
('wants', 'antsway'), ('forth', 'orthfay'), ('drove', 'ovedray'), ('mouth',
'outhmay'), ('worry', 'orryway'), ('dried', 'ieddray'), ('stood', 'oodstay'),
('brush', 'ushbray'), ('sting', 'ingstay'), ('mirth', 'irthmay'), ('heads',
'eadshay'), ('sight', 'ightsay'), ('whose', 'osewhay'), ('drawn', 'awndray'),
('score', 'orescay'), ('party', 'artypay'), ('tears', 'earstay'), ('heart',
'earthay'), ('words', 'ordsway'), ('hunts', 'untshay'), ('trace', 'acetray'),
('trial', 'ialtray'), ('swoon', 'oonsway'), ('knees', 'eesknay'), ('bring',
'ingbray'), ('solid', 'olidsay'), ('sharp', 'arpshay'), ('basis', 'asisbay'),
('cavil', 'avilcay'), ('rated', 'atedray'), ('grave', 'avegray'), ('serve',
'ervesay'), ('betty', 'ettybay'), ('judge', 'udgejay'), ('boast', 'oastbay'),
('march', 'archmay'), ('built', 'uiltbay'), ('clear', 'earclay'), ('happy',
'appyhay'), ('fruit', 'uitfray'), ('witty', 'ittyway'), ('total', 'otaltay'),
('dared', 'aredday'), ('defer', 'eferday'), ('tried', 'iedtray'), ('vexed',
'exedvay'), ('walls', 'allsway'), ('liked', 'ikedlay'), ('prove', 'ovepray'),
('sneer', 'eersnay'), ('merit', 'eritmay'), ('scott', 'ottscay'), ('madam',
'adammay'), ('truth', 'uthtray'), ('minor', 'inormay'), ('stare', 'arestay'),
('calls', 'allscay'), ('lucky', 'uckylay'), ('hence', 'encehay'), ('press',
'esspray'), ('comes', 'omescay'), ('dined', 'inedday'), ('think', 'inkthay'),
('plate', 'ateplay'), ('grief', 'iefgray'), ('names', 'amesnay'), ('stock',
'ockstay'), ('short', 'ortshay'), ('moved', 'ovedmay'), ('teach', 'eachtay'),
('fears', 'earsfay'), ('tiled', 'iledtay'), ('cried', 'iedcray'), ('mound',
'oundmay'), ('small', 'allsmay'), ('twice', 'icetway'), ('lines', 'ineslay'),
('looks', 'ookslay'), ('rival', 'ivalray'), ('first', 'irstfay'), ('wound',
'oundway'), ('saves', 'avessay'), ('nurse', 'ursenay'), ('treat', 'eattray'),
('could', 'ouldcay'), ('rider', 'iderray'), ('lobby', 'obbylay'), ('table',
'abletay'), ('touch', 'ouchtay'), ('topic', 'opictay'), ('stand', 'andstay'),
('leant', 'eantlay'), ('grass', 'assgray'), ('check', 'eckchay'), ('quite',
'itequay'), ('wives', 'ivesway'), ('walks', 'alksway'), ('labor', 'aborlay'),
('house', 'ousehay'), ('drops', 'opsdray'), ('loose', 'ooselay'), ('tempt',
'empttay'), ('place', 'aceplay'), ('local', 'ocallay'), ('seats', 'eatssay'),
('gates', 'atesgay'), ('dropt', 'optdray'), ('cares', 'arescay'), ('falls',
'allsfay'), ('speak', 'eakspay'), ('shops', 'opsshay'), ('groom', 'oomgray'),
('sugar', 'ugarsay'), ('front', 'ontfray'), ('visit', 'isitvay'), ('raise',
'aiseray'), ('fetch', 'etchfay'), ('silks', 'ilkssay'), ('beset', 'esetbay'),
('rocks', 'ocksray'), ('start', 'artstay'), ('winks', 'inksway'), ('sends',
'endssay'), ('piano', 'ianopay'), ('fully', 'ullyfay'), ('widow', 'idowway'),
('wiser', 'iserway'), ('dirty', 'irtyday'), ('below', 'elowbay'), ('manor',
'anormay'), ('scale', 'alescay'), ('whole', 'olewhay'), ('minds', 'indsmay'),
('glebe', 'ebeglay'), ('fifty', 'iftyfay'), ('years', 'yearsay'), ('story',
'orystay'), ('fresh', 'eshfray'), ('begun', 'egunbay'), ('dance', 'anceday'),
('jokes', 'okesjay'), ('turns', 'urnstay'), ('honor', 'onorhay'), ('spite',
'itespay'), ('green', 'eengray'), ('saint', 'aintsay'), ('hated', 'atedhay'),
('doors', 'oorsday'), ('shown', 'ownshay'), ('power', 'owerpay'), ('today',
'odaytay'), ('floor', 'oorflay'), ('force', 'orcefay'), ('right', 'ightray'),
('trick', 'icktray'), ('dread', 'eaddray'), ('final', 'inalfay'), ('shake',

'akeshay'), ('lamps', 'ampslay'), ('dying', 'yingday'), ('black', 'ackblay'), ('grand', 'andgray'), ('field', 'ieldfay'), ('plead', 'eadplay'), ('loved', 'ovedlay'), ('terms', 'ermstay'), ('spend', 'endspay'), ('mohrs', 'ohrsmay'), ('young', 'youngay'), ('great', 'eatgray'), ('guess', 'uessgay'), ('ninny', 'innynay'), ('hasty', 'astyhay'), ('nerve', 'ervenay'), ('meals', 'ealsmay'), ('three', 'eethray'), ('shade', 'adeshay'), ('store', 'orestay'), ('daily', 'ailyday'), ('taste', 'astetay'), ('gross', 'ossgray'), ('dares', 'aresday'), ('grate', 'ategray'), ('stuff', 'uffstay'), ('sheet', 'eetshay')], (14, 18): [('breakfast-room', 'eakfastbray-oomray'), ('lavender-water', 'avenderlay-aterway'), ('broken-hearted', 'okenbray-eartedhay'), ('spunging-house', 'ungingspay-ousehay')], (9, 12): [('unwelcome', 'unwelcomeway'), ('exhausted', 'exhaustedway'), ('intrusion', 'intrusionway'), ('untouched', 'untouchedway'), ('interview', 'interviewway'), ('effusions', 'effusionsway'), ('impartial', 'impartialway'), ('inventive', 'inventiveway'), ('emergency', 'emergencyway'), ('officious', 'officiousway'), ('everybody', 'everybodyway'), ('observant', 'observantway'), ('emptiness', 'emptinessway'), ('occasions', 'occasionsway'), ('assembled', 'assembledway'), ('admirable', 'admirableway'), ('uselessly', 'uselesslyway'), ('excellent', 'excellentway'), ('associate', 'associateway'), ('affecting', 'affectingway'), ('absurdity', 'absurdityway'), ('allowable', 'allowableway'), ('appointed', 'appointedway'), ('overheard', 'overheardway'), ('improving', 'improvingway'), ('eagerness', 'eagernessway'), ('unluckily', 'unluckilyway'), ('allowance', 'allowanceway'), ('estimable', 'estimableway'), ('obviating', 'obviatingway'), ('entrusted', 'entrustedway'), ('arranging', 'arrangingway'), ('exclaimed', 'exclaimedway'), ('ingenuity', 'ingenuityway'), ('outwardly', 'outwardlyway'), ('expensive', 'expensiveway'), ('entreated', 'entreatedway'), ('unconcern', 'unconcernway'), ('obtaining', 'obtainingway'), ('interests', 'interestsway'), ('assiduous', 'assiduousway'), ('adjusting', 'adjustingway'), ('ungenteel', 'ungenteelway'), ('encourage', 'encourageway'), ('apartment', 'apartmentway'), ('inheritor', 'inheritorway'), ('unceasing', 'unceasingway'), ('insulting', 'insultingway'), ('unchanged', 'unchangedway'), ('animation', 'animationway'), ('enclosure', 'enclosureway'), ('apologize', 'apologizeway'), ('earnestly', 'earnestlyway'), ('exquisite', 'exquisiteway'), ('attending', 'attendingway'), ('invention', 'inventionway'), ('enquiries', 'enquiriesway'), ('appetites', 'appetitesway'), ('additions', 'additionsway'), ('impudence', 'impudenceway'), ('insincere', 'insincereway'), ('abundance', 'abundanceway'), ('exchanged', 'exchangedway'), ('extremely', 'extremelyway'), ('inflicted', 'inflictedway'), ('attracted', 'attractedway'), ('inelegant', 'inelegantway'), ('undoubted', 'undoubtedway'), ('etiquette', 'etiquetteway'), ('unfeigned', 'unfeignedway'), ('agreeable', 'agreeableway'), ('unsubdued', 'unsubduedway'), ('inspiring', 'inspiringway'), ('unguarded', 'unguardedway'), ('advancing', 'advancingway'), ('undressed', 'undressedway'), ('excursive', 'excursiveway'), ('extending', 'extendingway'), ('extolling', 'extollingway'), ('offending', 'offendingway'), ('eloquence', 'eloquenceway'), ('ornaments', 'ornamentsway'), ('intruding', 'intrudingway'), ('incessant', 'incessantway'), ('unwilling', 'unwillingway'), ('annamaria', 'annamariaway'), ('unbounded', 'unboundedway'), ('incurable', 'incurableway'), ('unnoticed', 'unnoticedway'), ('expressly', 'expresslyway'), ('impetuous', 'impetuousway'), ('appearing', 'appearingway'), ('ingenious', 'ingeniousway'), ('amazement', 'amazementway'),

('animating', 'animatingway'), ('interfere', 'interfereway'), ('ascending', 'ascendingway'), ('affection', 'affectionway'), ('undergone', 'undergoneway'), ('unstudied', 'unstudiedway'), ('attempted', 'attemptedway'), ('infirmity', 'infirmityway'), ('apologies', 'apologiesway'), ('elsewhere', 'elsewhereway'), ('expecting', 'expectingway'), ('informing', 'informingway'), ('admission', 'admissionway'), ('extremity', 'extremityway')], (6, 9): [('unread', 'unreadway'), ('owning', 'owningway'), ('odious', 'odiousway'), ('either', 'eitherway'), ('insult', 'insultway'), ('awhile', 'awhileway'), ('assist', 'assistway'), ('injure', 'injureway'), ('eluded', 'eludedway'), ('others', 'othersway'), ('untied', 'untiedway'), ('excess', 'excessway'), ('urgent', 'urgentway'), ('appear', 'appearway'), ('aspect', 'aspectway'), ('issued', 'issuedway'), ('evince', 'evinceway'), ('eating', 'eatingway'), ('easter', 'easterway'), ('arisen', 'arisenway'), ('earned', 'earnedway'), ('esteem', 'esteemway'), ('accuse', 'accuseway'), ('owners', 'ownersway'), ('events', 'eventsway'), ('energy', 'energyway'), ('affair', 'affairway'), ('effect', 'effectway'), ('unlike', 'unlikeway'), ('unison', 'unisonway'), ('enough', 'enoughway'), ('invite', 'inviteway'), ('income', 'incomeway'), ('extent', 'extentway'), ('office', 'officeway'), ('ankles', 'anklesway'), ('admire', 'admireway'), ('induce', 'induceway'), ('infant', 'infantway'), ('expand', 'expandway'), ('orders', 'ordersway'), ('occupy', 'occupyway'), ('attend', 'attendway'), ('unable', 'unableway'), ('olives', 'olivesway'), ('awaken', 'awakenway'), ('engage', 'engageway'), ('eldest', 'eldestway'), ('eyeing', 'eyeingway'), ('advice', 'adviceway'), ('always', 'alwaysway'), ('insist', 'insistway'), ('accept', 'acceptway'), ('indies', 'indiesway'), ('excite', 'exciteway'), ('openly', 'openlyway'), ('oblige', 'obligeway'), ('amidst', 'amidstway'), ('across', 'acrossway'), ('afraid', 'afraidway'), ('uproar', 'uproarway')], (8, 10): [('villages', 'illagesvay'), ('doubtful', 'oubtfulday'), ('deceived', 'eceivedday'), ('foreseen', 'oreseenfay'), ('dennison', 'ennisonday'), ('steadily', 'eadilystay'), ('startled', 'artledstay'), ('drawings', 'awingsdray'), ('rejected', 'ejectedray'), ('deserves', 'eservesday'), ('curricle', 'urriclecay'), ('domestic', 'omesticday'), ('believed', 'elievedbay'), ('packages', 'ackagespay'), ('kneeling', 'eelingknay'), ('degraded', 'egradedday'), ('medicine', 'edicinemay'), ('promoted', 'omotedpray'), ('wondered', 'onderedway'), ('reproach', 'eproachray'), ('somerset', 'omersetsay'), ('contents', 'ontentscay'), ('children', 'ildrenchay'), ('politics', 'oliticspay'), ('hastened', 'astenedhay'), ('relative', 'elativeray'), ('henshawe', 'enshawehay'), ('material', 'aterialmay'), ('welcomed', 'elcomedway'), ('weddings', 'eddingsway'), ('diverted', 'ivertedday'), ('patience', 'atiencepay'), ('delivery', 'eliveryday'), ('hazarded', 'azardedhay'), ('happiest', 'appiesthay'), ('trifling', 'iflingtray'), ('casement', 'asementcay'), ('reformed', 'eformedray'), ('demanded', 'emandedday'), ('ceremony', 'eremonycay'), ('vigorous', 'igorousvay'), ('mornings', 'orningsmay'), ('hilarity', 'ilarityhay'), ('daughter', 'aughterday'), ('vexation', 'exationvay'), ('shocking', 'ockingshay'), ('previous', 'eviouspray'), ('pleasant', 'easantplay'), ('candidly', 'andidlycay'), ('cheerful', 'eerfulchay'), ('festival', 'estivalfay'), ('changing', 'angingchay'), ('conveyed', 'onveyedcay'), ('lombardy', 'ombardylay'), ('militate',

'ilitatemay'), ('humanity', 'umanityhay'), ('giggling', 'igglinggay'),
('standard', 'andardstay'), ('returned', 'eturnedray'), ('proclaim',
'oclaimpray'), ('ridicule', 'idiculeray'), ('pointers', 'ointerspay'),
('grounded', 'oundedgray'), ('contrast', 'ontrastcay'), ('feelings',
'eelingsfay'), ('mistakes', 'istakesmay'), ('prepared', 'eparedpray'),
('stranger', 'angerstray'), ('habitual', 'abitualhay'), ('firmness',
'irmnessfay'), ('rational', 'ationalray'), ('regulate', 'egulateray'),
('claiming', 'aimingclay'), ('knocking', 'ockingknay'), ('servants',
'ervantssay'), ('goodness', 'oodnessgay'), ('mutually', 'utuallymay'),
('legacies', 'egacieslay'), ('sisterly', 'isterlysay'), ('punctual',
'unctualpay'), ('deciding', 'ecidingday'), ('suitable', 'uitablesay'),
('guarding', 'uardinggay'), ('converse', 'onversecay'), ('slighter',
'ighterslay'), ('dissuade', 'issuadeday'), ('despatch', 'espatchday'),
('blackest', 'ackestblay'), ('recalled', 'ecalledray'), ('loitered',
'oiteredlay'), ('dreading', 'eadingdray'), ('relieved', 'elievedray'),
('westward', 'estwardway'), ('thistles', 'istlesthay'), ('defended',
'efendedday'), ('contempt', 'ontemptcay'), ('sensibly', 'ensiblysay'),
('romantic', 'omanticray'), ('comforts', 'omfortscay'), ('jealousy',
'ealousyjay'), ('repeated', 'epeatedray'), ('banditti', 'andittibay'),
('greatest', 'eatestgray'), ('mistrust', 'istrustmay'), ('forwards',
'orwardsfay'), ('rewarded', 'ewardedray'), ('watching', 'atchingway'),
('negative', 'egativenay'), ('sympathy', 'ympathysay'), ('helpless',
'elplesshay'), ('quickest', 'ickestquay'), ('portrait', 'ortraitpay'),
('consoled', 'onsoledcay'), ('fourteen', 'ourteenfay'), ('retarded',
'etardedray'), ('comments', 'ommentscay'), ('jennings', 'enningsjay'),
('securing', 'ecuringsay'), ('withdrew', 'ithdrewway'), ('youthful',
'youthfulay'), ('bachelor', 'achelorbay'), ('convince', 'onvincecay'),
('measured', 'easuredmay'), ('goodwill', 'oodwillgay'), ('branched',
'anchedbray'), ('purchase', 'urchasepay'), ('covenant', 'ovenantcay'),
('reseated', 'eseatedray'), ('concerns', 'oncernscay'), ('hardship',
'ardshiphay'), ('lessened', 'essenedlay'), ('delaford', 'elafordday'),
('governed', 'overnedgay'), ('listened', 'istenedlay'), ('constant',
'onstantcay'), ('fatigued', 'atiguedfay'), ('persuade', 'ersuadepay'),
('critique', 'itiquecray'), ('remedies', 'emediesray'), ('stepping',
'eppingstay'), ('kindness', 'indnesskay'), ('sociable', 'ociablesay'),
('meetings', 'eetingsmay'), ('deserved', 'eservedday'), ('thoughts',
'oughtsthay'), ('becoming', 'ecomingbay'), ('heartily', 'eartilyhay'),
('promised', 'omisedpray'), ('shutting', 'uttingshay'), ('reproved',
'eprovedray'), ('ladyship', 'adyshiplay'), ('pursuing', 'ursuingpay'),
('residing', 'esidingray'), ('sentence', 'entencesay'), ('revealed',
'evealedray'), ('decrease', 'ecreaseday'), ('speedily', 'eedilyspay'),
('commands', 'ommandscay'), ('mistaken', 'istakenmay'), ('settling',
'ettlingsay'), ('polished', 'olishedpay'), ('stanhill', 'anhillstay'),
('retailed', 'etailedray'), ('pointing', 'ointingpay'), ('strongly',
'onglystray'), ('subjects', 'ubjectssay'), ('calmness', 'almnesscay'),
('doctrine', 'octrineday'), ('handsome', 'andsomehay'), ('profited',
'ofitedpray'), ('planning', 'anningplay'), ('denoting', 'enotingday'),
('commonly', 'ommonlycay'), ('bestowed', 'estowedbay'), ('marriage',

'arriagemay'), ('presents', 'esentspray'), ('complete', 'ompletecay'),
('striking', 'ikingstray'), ('needless', 'eedlessnay'), ('plighted',
'ightedplay'), ('tranquil', 'anquiltray'), ('spacious', 'aciousspay'),
('nonsense', 'onsensenay'), ('likeness', 'ikenesslay'), ('valuable',
'aluablevay'), ('laudable', 'audablelay'), ('delirium', 'eliriumday'),
('believes', 'elievesbay'), ('premises', 'emisespray'), ('presumed',
'esumedpray'), ('coldness', 'oldnesscay'), ('metaphor', 'etaphormay'),
('thankful', 'ankfulthay'), ('suffered', 'ufferedsay'), ('preserve',
'eservepray'), ('resorted', 'esortedray'), ('felicity', 'elicityfay'),
('betrayed', 'etrayedbay'), ('throwing', 'owingthray'), ('powerful',
'owerfulpay'), ('humoured', 'umouredhay'), ('practise', 'actisepray'),
('coloured', 'olouredcay'), ('regiment', 'egimentray'), ('stubborn',
'ubbornstay'), ('huswifes', 'uswifeshay'), ('painting', 'aintingpay'),
('friendly', 'iendlyfray'), ('happened', 'appenedhay'), ('moderate',
'oderatemay'), ('solitary', 'olitarysay'), ('wildness', 'ildnessway'),
('marianne', 'ariannemay'), ('delicacy', 'elicacyday'), ('progress',
'ogresspray'), ('coachman', 'oachmancay'), ('grieving', 'ievinggray'),
('troubles', 'oublestray'), ('coolness', 'oolnesscay'), ('possibly',
'ossiblypay'), ('relating', 'elatingray'), ('patterns', 'atternspay'),
('consists', 'onsistscay'), ('creating', 'eatingcray'), ('confused',
'onfusedcay'), ('critical', 'iticalcray'), ('laboured', 'abouredlay'),
('minutely', 'inutelymay'), ('tendency', 'endencytay'), ('distrust',
'istrustday'), ('distress', 'istressday'), ('conclude', 'oncludecay'),
('sixpence', 'ixpencesay'), ('somewhat', 'omewhatsay'), ('coincide',
'oincidecay'), ('harmless', 'armlesshay'), ('juvenile', 'uvenilejay'),
('pleasing', 'easingplay'), ('precious', 'eciouspray'), ('restrain',
'estrainray'), ('produced', 'oducedpray'), ('renewing', 'enewingray'),
('pinching', 'inchingpay'), ('severely', 'everelysay'), ('blossoms',
'ossomsblay'), ('heighten', 'eightenhay'), ('wandered', 'anderedway'),
('properly', 'operlypray'), ('veracity', 'eracityvay'), ('released',
'eleasedray'), ('transact', 'ansacttray'), ('helpmate', 'elpmatehay'),
('business', 'usinessbay'), ('concerto', 'oncertocay'), ('reserved',
'eservedray'), ('reliance', 'elianceray'), ('headache', 'eadachehay'),
('families', 'amiliesfay'), ('centered', 'enteredcay'), ('mulberry',
'ulberrymay'), ('sensible', 'ensiblesay'), ('steepest', 'eepeststay'),
('compared', 'omparedcay'), ('speaking', 'eakingspay'), ('censured',
'ensuredcay'), ('screened', 'eenedscray'), ('pictures', 'icturespay'),
('reaching', 'eachingray'), ('wonderer', 'ondererway'), ('whenever',
'eneverwhay'), ('detested', 'etestedday'), ('dispatch', 'ispatchday'),
('shutters', 'uttersshay'), ('disgrace', 'isgraceday'), ('dejected',
'ejectedday'), ('conquest', 'onquestcay'), ('foretell', 'oretellfay'),
('weakened', 'eakenedway'), ('brooding', 'oodingbray'), ('shedding',
'eddingshay'), ('poignant', 'oignantpay'), ('trembled', 'embledtray'),
('resolute', 'esoluteray'), ('reverted', 'evertedray'), ('silenced',
'ilencedsay'), ('jointure', 'ointurejay')], (10, 13): [('illaudable',
'illaudableway'), ('incivility', 'incivilityway'), ('oppression',
'oppressionway'), ('acquainted', 'acquaintedway'), ('economical',
'economicalway'), ('ebullition', 'ebullitionway'), ('affronting',

```
'affrontingway'), ('unforeseen', 'unforeseenway'), ('auspicious',
'auspiciousway'), ('indecorous', 'indecorousway'), ('attractive',
'attractiveway'), ('exercising', 'exercisingway'), ('unsatiable',
'unsatiableway'), ('individual', 'individualway'), ('enthusiasm',
'enthusiasmway'), ('announcing', 'announcingway'), ('indistinct',
'indistinctway'), ('affliction', 'afflictionway'), ('imaginable',
'imaginableway'), ('obstructed', 'obstructedway'), ('improperly',
'improperlyway'), ('illiterate', 'illiterateway'), ('interested',
'interestedway'), ('infamously', 'infamouslyway'), ('amusements',
'amusementsway'), ('endeavours', 'endeavoursway'), ('inhabiting',
'inhabitingway'), ('indisposed', 'indisposedway'), ('acquiesced',
'acquiescedway'), ('anticipate', 'anticipateway'), ('accurately',
'accuratelyway'), ('influenced', 'influencedway'), ('assemblies',
'assembliesway'), ('approached', 'approachedway'), ('accusation',
'accusationway'), ('apparently', 'apparentlyway'), ('intimately',
'intimatelyway'), ('uneasiness', 'uneasinessway'), ('alteration',
'alterationway'), ('artificial', 'artificialway'), ('admiration',
'admirationway'), ('accidental', 'accidentalway'), ('unexpected',
'unexpectedway'), ('insensible', 'insensibleway'), ('expressing',
'expressingway'), ('accelerate', 'accelerateway'), ('everything',
'everythingway'), ('increasing', 'increasingway'), ('undervalue',
'undervalueway'), ('invaluable', 'invaluableway'), ('affections',
'affectionsway'), ('infinitely', 'infinitelyway'), ('unsuitable',
'unsuitableway'), ('authorised', 'authorisedway'), ('impudently',
'impudentlyway'), ('unpardoned', 'unpardonedway'), ('inevitable',
'inevitableway'), ('ungenerous', 'ungenerousway'), ('impression',
'impressionway'), ('indelicacy', 'indelicacyway'), ('inducement',
'inducementway'), ('unaffected', 'unaffectedway'), ('instigated',
'instigatedway'), ('inquietude', 'inquietudeway'), ('attempting',
'attemptingway'), ('experience', 'experienceway'), ('impatience',
'impatienceway'), ('indulgence', 'indulgenceway'), ('affability',
'affabilityway'), ('unavailing', 'unavailingway'), ('incommoded',
'incommodedway'), ('altogether', 'altogetherway')], (4, 6): [('done', 'oneday'),
('wife', 'ifeway'), ('plea', 'eaplay'), ('mall', 'allmay'), ('sang', 'angsay'),
('clue', 'ueclay'), ('bell', 'ellbay'), ('bind', 'indbay'), ('join', 'oinjay'),
('went', 'entway'), ('vile', 'ilevay'), ('hens', 'enshay'), ('tore', 'oretay'),
('rage', 'ageray'), ('flow', 'owflay'), ('gate', 'ategay'), ('palm', 'almpay'),
('give', 'ivegay'), ('lips', 'ipslay'), ('sink', 'inksay'), ('rent', 'entray'),
('gave', 'avegay'), ('joke', 'okejay'), ('miss', 'issmay'), ('mere', 'eremay'),
('made', 'ademay'), ('dogs', 'ogsday'), ('bent', 'entbay'), ('duly', 'ulyday'),
('when', 'enwhay'), ('rate', 'ateray'), ('sure', 'uresay'), ('land', 'andlay'),
('pink', 'inkpay'), ('hint', 'inthay'), ('cast', 'astcay'), ('park', 'arkpay'),
('bold', 'oldbay'), ('door', 'oorday'), ('bulk', 'ulkbay'), ('doat', 'oatday'),
('coat', 'oatcay'), ('date', 'ateday'), ('foot', 'ootfay'), ('hand', 'andhay'),
('damp', 'ampday'), ('pall', 'allpay'), ('goes', 'oesgay'), ('rung', 'ungray'),
('none', 'onenay'), ('game', 'amegay'), ('nods', 'odsnay'), ('read', 'eadray'),
('gold', 'oldgay'), ('sing', 'ingsay'), ('puts', 'utspay'), ('cats', 'atscay'),
('many', 'anymay'), ('dare', 'areday'), ('noon', 'oonnay'), ('said', 'aidsay'),
```

```
('mark', 'arkmay'), ('case', 'asecay'), ('hate', 'atehay'), ('word', 'ordway'),
('beat', 'eatbay'), ('lose', 'oselay'), ('fish', 'ishfay'), ('laid', 'aidlay'),
('fine', 'inefay'), ('snug', 'ugsnay'), ('moss', 'ossmay'), ('heir', 'eirhay'),
('sigh', 'ighsay'), ('mend', 'endmay'), ('sake', 'akesay'), ('pass', 'asspay'),
('debt', 'ebtday'), ('lane', 'anelay'), ('days', 'aysday'), ('deep', 'eepday'),
('song', 'ongsay'), ('spot', 'otspay'), ('took', 'ooktay'), ('both', 'othbay'),
('year', 'yearay'), ('mama', 'amamay'), ('hill', 'illhay'), ('beds', 'edsbay'),
('less', 'esslay'), ('gown', 'owngay'), ('kill', 'illkay'), ('free', 'eefray'),
('hall', 'allhay'), ('room', 'oomray'), ('gray', 'aygray'), ('beef', 'eefbay'),
('kept', 'eptkay'), ('what', 'atwhay'), ('blue', 'ueblay'), ('fell', 'ellfay'),
('draw', 'awdray'), ('play', 'ayplay'), ('baby', 'abybay'), ('rude', 'uderay'),
('trim', 'imtray'), ('chat', 'atchay'), ('bely', 'elybay'), ('fled', 'edflay'),
('past', 'astpay'), ('wait', 'aitway'), ('meal', 'ealmay'), ('wont', 'ontway'),
('week', 'eekway'), ('best', 'estbay'), ('line', 'inelay'), ('lawn', 'awnlay'),
('tree', 'eetray'), ('tend', 'endtay'), ('find', 'indfay'), ('five', 'ivefay'),
('gone', 'onegay'), ('need', 'eednay'), ('drew', 'ewdray'), ('more', 'oremay'),
('cool', 'oolcay'), ('suit', 'uitsay'), ('from', 'omfray'), ('cure', 'urecay'),
('will', 'illway'), ('slow', 'owslay'), ('they', 'eythay'), ('name', 'amenay'),
('shot', 'otshay'), ('come', 'omecay'), ('tied', 'iedtay'), ('came', 'amecay'),
('boys', 'oysbay'), ('wish', 'ishway'), ('hour', 'ourhay'), ('kind', 'indkay'),
('mile', 'ilemay'), ('quit', 'itquay'), ('john', 'ohnjay'), ('hazy', 'azyhay'),
('fall', 'allfay'), ('sell', 'ellsay'), ('tale', 'aletay'), ('grew', 'ewgray'),
('seek', 'eeksay'), ('muff', 'uffmay'), ('soft', 'oftsay'), ('same', 'amesay'),
('body', 'odybay'), ('head', 'eadhay'), ('like', 'ikelay'), ('here', 'erehay'),
('help', 'elphay'), ('sail', 'ailsay'), ('pope', 'opepay'), ('deny', 'enyday'),
('bond', 'ondbay'), ('bath', 'athbay'), ('blow', 'owblay'), ('hold', 'oldhay'),
('safe', 'afesay'), ('path', 'athpay'), ('feed', 'eedfay'), ('thus', 'usthay'),
('four', 'ourfay'), ('lest', 'estlay'), ('thin', 'inthay')], (11, 14):
[('approaching', 'approachingway'), ('unequivocal', 'unequivocalway'),
('exceedingly', 'exceedinglyway'), ('application', 'applicationway'),
('observation', 'observationway'), ('unhappiness', 'unhappinessway'),
('interrupted', 'interruptedway'), ('enlightened', 'enlightenedway'),
('appearances', 'appearancesway'), ('unconnected', 'unconnectedway'),
('unconcerned', 'unconcernedway'), ('indignation', 'indignationway'),
('accommodate', 'accommodateway'), ('inattention', 'inattentionway'),
('undoubtedly', 'undoubtedlyway'), ('intercourse', 'intercourseway'),
('imperfectly', 'imperfectlyway'), ('imprudently', 'imprudentlyway'),
('affectation', 'affectationway'), ('instigation', 'instigationway'),
('inclination', 'inclinationway'), ('undervalued', 'undervaluedway'),
('affirmative', 'affirmativeway'), ('inconstancy', 'inconstancyway'),
('universally', 'universallyway'), ('engagements', 'engagementsway'),
('exclamation', 'exclamationway'), ('elucidation', 'elucidationway'),
('accustomary', 'accustomaryway'), ('approbation', 'approbationway'),
('unnecessary', 'unnecessaryway'), ('associating', 'associatingway'),
('attributing', 'attributingway'), ('unobtrusive', 'unobtrusiveway'),
('impropriety', 'improprietyway'), ('examination', 'examinationway'),
('alterations', 'alterationsway'), ('unexpressed', 'unexpressedway'),
('abridgement', 'abridgementway'), ('exhilarated', 'exhilaratedway'),
```

```
('unsuspected', 'unsuspectedway'), ('independent', 'independentway'),
('alternately', 'alternatelyway'), ('enforcement', 'enforcementway'),
('undertaking', 'undertakingway'), ('outstripped', 'outstrippedway'),
('appointment', 'appointmentway'), ('explanation', 'explanationway'),
('experienced', 'experiencedway'), ('unknowingly', 'unknowinglyway'),
('undesirable', 'undesirableway'), ('ascertained', 'ascertainedway'),
('enumeration', 'enumerationway'), ('alleviation', 'alleviationway'),
('instruction', 'instructionway'), ('undeserving', 'undeservingway'),
('earnestness', 'earnestnessway'), ('insincerity', 'insincerityway'),
('accompanied', 'accompaniedway'), ('employments', 'employmentsway'),
('inferiority', 'inferiorityway'), ('inexcusable', 'inexcusableway'),
('introducing', 'introducingway'), ('extravagant', 'extravagantway'),
('impatiently', 'impatientlyway'), ('attentively', 'attentivelyway'),
('eligibility', 'eligibilityway'), ('uncertainty', 'uncertaintyway')], (7, 10):
[('anxiety', 'anxietyway'), ('attacks', 'attacksway'), ('ailment',
'ailmentway'), ('awkward', 'awkwardway'), ('effects', 'effectsway'), ('induced',
'inducedway'), ('uncivil', 'uncivilway'), ('against', 'againstway'), ('apology',
'apologyway'), ('endless', 'endlessway'), ('alluded', 'alludedway'), ('ability',
'abilityway'), ('address', 'addressway'), ('engaged', 'engagedway'), ('eagerly',
'eagerlyway'), ('expects', 'expectsway'), ('elliott', 'elliottway'), ('oneself',
'oneselfway'), ('anybody', 'anybodyway'), ('augment', 'augmentway'), ('utility',
'utilityway'), ('unquiet', 'unquietway'), ('infancy', 'infancyway'), ('altered',
'alteredway'), ('unequal', 'unequalway'), ('adapted', 'adaptedway'), ('useless',
'uselessway'), ('opposed', 'opposedway'), ('artless', 'artlessway'), ('imagery',
'imageryway'), ('amiably', 'amiablyway'), ('uncouth', 'uncouthway'), ('affixed',
'affixedway'), ('impulse', 'impulseway'), ('opinion', 'opinionway'), ('imitate',
'imitateway'), ('admires', 'admiresway'), ('affront', 'affrontway'), ('amusing',
'amusingway'), ('accents', 'accentsway'), ('implied', 'impliedway'), ('ecstasy',
'ecstasyway'), ('illness', 'illnessway'), ('entreat', 'entreatway'), ('another',
'anotherway'), ('opening', 'openingway'), ('england', 'englandway'), ('ensured',
'ensuredway'), ('imputed', 'imputedway'), ('enabled', 'enabledway'), ('offence',
'offenceway'), ('insured', 'insuredway'), ('urgency', 'urgencyway'), ('amazing',
'amazingway'), ('uniform', 'uniformway'), ('avarice', 'avariceway'), ('ellison',
'ellisonway'), ('endowed', 'endowedway'), ('ashamed', 'ashamedway'), ('eloping',
'elopingway'), ('inhabit', 'inhabitway'), ('advised', 'advisedway'), ('attract',
'attractway'), ('already', 'alreadyway'), ('enquire', 'enquireway'), ('affects',
'affectsway'), ('invalid', 'invalidway')], (9, 11): [('genuinely',
'enuinelygay'), ('columella', 'olumellacay'), ('vouchsafe', 'ouchsafevay'),
('forwarded', 'orwardedfay'), ('scattered', 'atteredscay'), ('christmas',
'istmaschray'), ('crimsoned', 'imsonedcray'), ('contrives', 'ontrivescay'),
('miserable', 'iserablemay'), ('preceding', 'ecedingpray'), ('healthful',
'ealthfulhay'), ('confiding', 'onfidingcay'), ('perfectly', 'erfectlypay'),
('prospects', 'ospectspray'), ('withdrawn', 'ithdrawnway'), ('lingering',
'ingeringlay'), ('contained', 'ontainedcay'), ('depressed', 'epressedday'),
('disgusted', 'isgustedday'), ('promoting', 'omotingpray'), ('supported',
'upportedsay'), ('gradually', 'aduallygray'), ('disturbed', 'isturbedday'),
('perpetual', 'erpetualpay'), ('pronounce', 'onouncepray'), ('sincerely',
'incerelysay'), ('seventeen', 'eventeensay'), ('presented', 'esentedpray'),
```

('medicines', 'edicinesmay'), ('syllables', 'yllablessay'), ('behaviour',
'ehaviourbay'), ('niggardly', 'iggardlynay'), ('commonest', 'ommonestcay'),
('necessary', 'ecessarynay'), ('prettiest', 'ettiestpray'), ('condemned',
'ondemnedcay'), ('conquests', 'onquestscay'), ('fashioned', 'ashionedfay'),
('hysterics', 'ystericshay'), ('smallness', 'allnesssmay'), ('professed',
'ofessedpray'), ('conceited', 'onceitedcay'), ('compliant', 'ompliantcay'),
('frankness', 'anknessfray'), ('merriment', 'errimentmay'), ('resembled',
'esembledray'), ('leisurely', 'eisurelylay'), ('dispersed', 'ispersedday'),
('departure', 'epartureday'), ('pecuniary', 'ecuniarypay'), ('frivolous',
'ivolousfray'), ('situation', 'ituationsay'), ('discourse', 'iscourseday'),
('resettled', 'esettledray'), ('beholding', 'eholdingbay'), ('gentleman',
'entlemangay'), ('chagrined', 'agrinedchay'), ('struggled', 'uggledstray'),
('preparing', 'eparingpray'), ('respected', 'espectedray'), ('newspaper',
'ewspapernay'), ('justified', 'ustifiedjay'), ('sacrifice', 'acrificesay'),
('stammered', 'ammeredstay'), ('smartness', 'artnesssmay'), ('continued',
'ontinuedcay'), ('contrived', 'ontrivedcay'), ('reconcile', 'econcileray'),
('modestest', 'odestestmay'), ('cordially', 'ordiallycay'), ('sentiment',
'entimentsay'), ('shameless', 'amelessshay'), ('reflected', 'eflectedray'),
('feelingly', 'eelinglyfay'), ('confident', 'onfidentcay'), ('proposals',
'oposalspray'), ('warehouse', 'arehouseway'), ('deviation', 'eviationday'),
('statement', 'atementstay'), ('submitted', 'ubmittedsay'), ('believing',
'elievingbay'), ('gentility', 'entilitygay'), ('resources', 'esourcesray'),
('composure', 'omposurecay'), ('deserving', 'eservingday'), ('vehemence',
'ehemencevay'), ('disgraced', 'isgracedday'), ('perplexed', 'erplexedpay'),
('prejudice', 'ejudicepray'), ('quickened', 'ickenedquay'), ('blundered',
'underedblay'), ('strongest', 'ongeststray'), ('prevailed', 'evailedpray'),
('languages', 'anguageslay'), ('reanimate', 'eanimateray'), ('stationer',
'ationerstay'), ('reminding', 'emindingray'), ('blameless', 'amelessblay'),
('principle', 'inciplepray'), ('hardening', 'ardeninghay'), ('suggested',
'uggestedsay'), ('desperate', 'esperateday'), ('landscape', 'andscapelay'),
('paragraph', 'aragraphpay'), ('provision', 'ovisionpray'), ('relations',
'elationsray'), ('fruitless', 'uitlessfray'), ('christian', 'istianchray'),
('stipulate', 'ipulatestay'), ('constancy', 'onstancycay'), ('gentlemen',
'entlemengay'), ('venturing', 'enturingvay'), ('northward', 'orthwardnay'),
('neighbour', 'eighbournay'), ('weakening', 'eakeningway'), ('persisted',
'ersistedpay'), ('malignant', 'alignantmay'), ('wandering', 'anderingway'),
('witnesses', 'itnessesway'), ('dangerous', 'angerousday'), ('safeguard',
'afeguardsay'), ('treasured', 'easuredtray'), ('cleveland', 'evelandclay'),
('patronage', 'atronagepay'), ('despising', 'espisingday'), ('foolishly',
'oolishlyfay'), ('seriously', 'eriouslysay'), ('restraint', 'estraintray'),
('september', 'eptembersay'), ('distrusts', 'istrustsday'), ('temperate',
'emperatetay'), ('finishing', 'inishingfay'), ('persuaded', 'ersuadedpay'),
('stupified', 'upifiedstay'), ('declaring', 'eclaringday'), ('dejection',
'ejectionday'), ('soothings', 'oothingssay'), ('conveying', 'onveyingcay'),
('possessed', 'ossessedpay'), ('described', 'escribedday'), ('spreading',
'eadingspray'), ('foresight', 'oresightfay'), ('political', 'oliticalpay'),
('concluded', 'oncludedcay'), ('doubtless', 'oubtlessday'), ('confusion',
'onfusioncay'), ('disliking', 'islikingday'), ('carriages', 'arriagescay'),

```
('favourite', 'avouritefay'), ('brilliant', 'illiantbray'), ('hastening',
'asteninghay'), ('precisely', 'eciselypray'), ('wondering', 'onderingway'),
('shoulders', 'oulderssshay'), ('honorably', 'onorablyhay'), ('privilege',
'ivilegepray'), ('deference', 'eferenceday'), ('vulgarity', 'ulgarityvay'),
('reprimand', 'eprimandray'), ('surveying', 'urveyingsay'), ('handsomer',
'andsomerhay'), ('shivering', 'iveringshay'), ('tolerable', 'olerabletay'),
('requiring', 'equiringray'), ('precision', 'ecisionpray'), ('courtland',
'ourtlandcay'), ('sportsman', 'ortsmanspay'), ('happiness', 'appinesshay'),
('pointedly', 'ointedlypay'), ('benefited', 'enefitedbay'), ('substance',
'ubstancesay'), ('beginning', 'eginningbay'), ('gratitude', 'atitudegray'),
('negligent', 'egligentnay'), ('belonging', 'elongingbay'), ('mortified',
'ortifiedmay'), ('neglected', 'eglectednay'), ('shillings', 'illingsshay'),
('warranted', 'arrantedway'), ('performed', 'erformedpay'), ('necessity',
'ecessitynay'), ('slightest', 'ightestslay'), ('guardians', 'uardiansgay'),
('preferred', 'eferredpray'), ('forgotten', 'orgottenfay'), ('refreshed',
'efreshedray'), ('barouches', 'arouchesbay'), ('grievance', 'ievancegray'),
('seduction', 'eductionsay'), ('comprised', 'omprisedcay'), ('listening',
'isteninglay'), ('detection', 'etectionday'), ('narration', 'arrationnay'),
('deceiving', 'eceivingday'), ('provoking', 'ovokingpray'), ('cessation',
'essationcay'), ('questions', 'estionsquay'), ('continual', 'ontinualcay'),
('faintness', 'aintnessfay'), ('diffident', 'iffidentday'), ('referring',
'eferringray'), ('blessings', 'essingsblay'), ('therefore', 'ereforethay'),
('frugality', 'ugalityfray'), ('reprobate', 'eprobateray')], (11, 13):
[('tranquilize', 'anquilizetray'), ('fashionable', 'ashionablefay'),
('respectable', 'espectableray'), ('publication', 'ublicationpay'),
('superintend', 'uperintendsay'), ('recollected', 'ecollectedray'),
('cheerfuller', 'eerfullerchay'), ('concurrence', 'oncurrencecay'),
('restorative', 'estorativeray'), ('reproaching', 'eproachingray'),
('declaration', 'eclarationday'), ('discharging', 'ischargingday'),
('proclaiming', 'oclaimingpray'), ('discovering', 'iscoveringday'),
('possibility', 'ossibilitypay'), ('contributed', 'ontributedcay'),
('captivating', 'aptivatingcay'), ('misinformed', 'isinformedmay'),
('carefulness', 'arefulnesscay'), ('philosophic', 'ilosophicphay'),
('calculation', 'alculationcay'), ('consumption', 'onsumptioncay'),
('confinement', 'onfinementcay'), ('certainties', 'ertaintiescay'),
('destination', 'estinationday'), ('separations', 'eparationssay'),
('voluntarily', 'oluntarilyvay'), ('connoisseur', 'onnoisseurcay'),
('refinements', 'efinementsray'), ('subservient', 'ubservientsay'),
('marlborough', 'arlboroughmay'), ('sagaciously', 'agaciouslysay'),
('fascinating', 'ascinatingfay'), ('comfortable', 'omfortablecay'),
('reflections', 'eflectionsray'), ('sensibility', 'ensibilitysay'),
('forbearance', 'orbearancefay'), ('mischievous', 'ischievousmay'),
('disgraceful', 'isgracefulday'), ('temporizing', 'emporizingtay'),
('sacrificing', 'acrificingsay'), ('spontaneous', 'ontaneousspay'),
('flourishing', 'ourishingflay'), ('lamentation', 'amentationlay'),
('meditations', 'editationsmay'), ('quarrelling', 'arrellingquay'),
('discussions', 'iscussionsday'), ('supposition', 'uppositionsay'),
('comfortably', 'omfortablycay'), ('disappeared', 'isappearedday'),
```

```
('preparation', 'eparationpray'), ('consistency', 'onsistencycay'),
('compliments', 'omplimentscay'), ('promptitude', 'omptitudepray'),
('determining', 'eterminingday'), ('countenance', 'ountenancecay'),
('maintenance', 'aintenancemay'), ('contraction', 'ontractioncay'),
('dissipation', 'issipationday'), ('comparative', 'omparativecay'),
('desperation', 'esperationday'), ('displeasure', 'ispleasureday'),
('despatching', 'espatchingday'), ('playfellows', 'ayfellowsplay'),
('plantations', 'antationsplay'), ('vindication', 'indicationvay'),
('significant', 'ignificantsay'), ('discontents', 'iscontentsday'),
('housekeeper', 'ousekeeperhay'), ('playfulness', 'ayfulnessplay'),
('threatening', 'eateningthray'), ('meritorious', 'eritoriousmay'),
('tediousness', 'ediousnesstay'), ('convenience', 'onveniencecay'),
('possessions', 'ossessionspay'), ('consolation', 'onsolationcay'),
('composition', 'ompositioncay'), ('proposition', 'opositionpray'),
('correctness', 'orrectnesscay'), ('consequence', 'onsequencecay'),
('barbarously', 'arbarouslybay'), ('booksellers', 'ooksellersbay'),
('conjectures', 'onjecturescay'), ('violoncello', 'ioloncellovay'),
('dissembling', 'issemblingday'), ('rapturously', 'apturouslyray'),
('speculative', 'eculativespay'), ('remembering', 'ememberingray'),
('probability', 'obabilitypray'), ('punctuality', 'unctualitypay'),
('forgiveness', 'orgivenessfay'), ('disapproves', 'isapprovesday'),
('troublesome', 'oublesometray')], (3, 5): [('say', 'aysay'), ('for', 'orfay'),
('get', 'etgay'), ('pin', 'inpay'), ('not', 'otnay'), ('fit', 'itfay'), ('rap',
'apray'), ('lip', 'iplay'), ('pry', 'ypray'), ('rob', 'obray'), ('fir',
'irfay'), ('can', 'ancay'), ('two', 'otway'), ('him', 'imhay'), ('son',
'onsay'), ('let', 'etlay'), ('try', 'ytray'), ('hot', 'othay'), ('sum',
'umsay'), ('got', 'otgay'), ('law', 'awlay'), ('how', 'owhay'), ('far',
'arfay'), ('the', 'ethay'), ('sky', 'yskay'), ('sly', 'yslay'), ('cut',
'utcay'), ('hop', 'ophay'), ('had', 'adhay'), ('was', 'asway'), ('bow',
'owbay'), ('hue', 'uehay'), ('now', 'ownay'), ('vex', 'exvay'), ('yet',
'yetay'), ('sit', 'itsay'), ('due', 'ueday'), ('cap', 'apcay'), ('sat',
'atsay'), ('hit', 'ithay'), ('see', 'eesay'), ('pay', 'aypay'), ('nor',
'ornay'), ('fly', 'yflay'), ('dog', 'ogday'), ('too', 'ootay'), ('six',
'ixsay'), ('lap', 'aplay'), ('new', 'ewnay'), ('led', 'edlay'), ('sin',
'insay'), ('ten', 'entay'), ('his', 'ishay'), ('day', 'ayday')], (14, 17):
[('insurmountable', 'insurmountableway'), ('inconveniences',
'inconveniencesway'), ('aggrandizement', 'aggrandizementway'),
('irreconcilable', 'irreconcilableway'), ('unacknowledged',
'unacknowledgedway'), ('unpleasantness', 'unpleasantnessway')], (13, 16):
[('understanding', 'understandingway'), ('intentionally', 'intentionallyway'),
('embarrassment', 'embarrassmentway'), ('unembarrassed', 'unembarrassedway'),
('administering', 'administeringway'), ('instantaneous', 'instantaneousway'),
('intrinsically', 'intrinsicallyway'), ('indispensable', 'indispensableway'),
('impoverishing', 'impoverishingway'), ('opportunities', 'opportunitiesway'),
('impertinently', 'impertinentlyway'), ('independently', 'independentlyway'),
('encroachments', 'encroachmentsway'), ('uninteresting', 'uninterestingway'),
('investigation', 'investigationway'), ('unaccountable', 'unaccountableway'),
('irrepressible', 'irrepressibleway'), ('impossibility', 'impossibilityway')],
```

```
(10, 14): [('thirty-six', 'irtythay-ixsay')], (8, 11): [('increase',
'increaseway'), ('agitated', 'agitatedway'), ('unkindly', 'unkindlyway'),
('occurred', 'occurredway'), ('arranged', 'arrangedway'), ('appetite',
'appetiteway'), ('inclined', 'inclinedway'), ('indolent', 'indolentway'),
('inquired', 'inquiredway'), ('evenings', 'eveningsway'), ('obliging',
'obligingway'), ('equalled', 'equalledway'), ('accepted', 'acceptedway'),
('extended', 'extendedway'), ('entreaty', 'entreatyway'), ('exigence',
'exigenceway'), ('election', 'electionway'), ('examined', 'examinedway'),
('inviting', 'invitingway'), ('embitter', 'embitterway'), ('idleness',
'idlenessway'), ('admirers', 'admirersway'), ('editions', 'editionsway'),
('extorted', 'extortedway'), ('outweigh', 'outweighway'), ('attended',
'attendedway'), ('addition', 'additionway'), ('entitled', 'entitledway'),
('awakened', 'awakenedway'), ('attacked', 'attackedway'), ('arriving',
'arrivingway'), ('overlook', 'overlookway'), ('eminence', 'eminenceway'),
('adhering', 'adheringway'), ('obtained', 'obtainedway'), ('openness',
'opennessway'), ('allusion', 'allusionway'), ('anything', 'anythingway'),
('overcome', 'overcomeway'), ('announce', 'announceway'), ('enjoying',
'enjoyingway'), ('outlived', 'outlivedway'), ('emphasis', 'emphasisway'),
('insulted', 'insultedway'), ('esteemed', 'esteemedway'), ('alphabet',
'alphabetway'), ('enquired', 'enquiredway'), ('accident', 'accidentway'),
('occupied', 'occupiedway'), ('ellisons', 'ellisonsway'), ('avoiding',
'avoidingway'), ('attained', 'attainedway'), ('unwarily', 'unwarilyway'),
('employed', 'employedway'), ('altering', 'alteringway'), ('intimate',
'intimateway'), ('effected', 'effectedway'), ('offences', 'offencesway'),
('entirely', 'entirelyway'), ('abounded', 'aboundedway'), ('actually',
'actuallyway'), ('identify', 'identifyway'), ('applying', 'applyingway'),
('exterior', 'exteriorway'), ('auditors', 'auditorsway'), ('unfairly',
'unfairlyway'), ('eighteen', 'eighteenway'), ('embraced', 'embracedway'),
('unfolded', 'unfoldedway'), ('injuries', 'injuriesway'), ('elevated',
'elevatedway'), ('alarming', 'alarmingway'), ('adequate', 'adequateway'),
('unabated', 'unabatedway'), ('incurred', 'incurredway'), ('uncommon',
'uncommonway'), ('unawares', 'unawaresway'), ('unsuited', 'unsuitedway'),
('instance', 'instanceway'), ('interval', 'intervalway'), ('audacity',
'audacityway')], (5, 8): [('other', 'otherway'), ('ideas', 'ideasway'),
('agony', 'agonyway'), ('avoid', 'avoidway'), ('awful', 'awfulway'), ('april',
'aprilway'), ('enter', 'enterway'), ('infer', 'inferway'), ('abuse',
'abuseway'), ('ivory', 'ivoryway'), ('adopt', 'adoptway'), ('image',
'imageway'), ('agree', 'agreeway'), ('alike', 'alikeway'), ('earth',
'earthway'), ('ended', 'endedway'), ('owner', 'ownerway'), ('using',
'usingway'), ('ought', 'oughtway'), ('abhor', 'abhorway'), ('altar',
'altarway'), ('often', 'oftenway'), ('atone', 'atoneway'), ('angel',
'angelway'), ('arise', 'ariseway'), ('under', 'underway'), ('abode',
'abodeway'), ('exert', 'exertway'), ('usual', 'usualway'), ('aches',
'achesway'), ('enjoy', 'enjoyway'), ('urged', 'urgedway'), ('admit',
'admitway'), ('aside', 'asideway'), ('equal', 'equalway'), ('alive',
'aliveway'), ('arose', 'aroseway'), ('eight', 'eightway'), ('opera',
'operaway'), ('ennui', 'ennuiway'), ('utter', 'utterway'), ('ankle',
'ankleway')], (2, 5): [('is', 'isway'), ('ay', 'ayway'), ('ah', 'ahway'), ('ii',
```

'iiway'), ('or', 'orway'), ('am', 'amway'), ('if', 'ifway'), ('an', 'anway')],
(12, 14): [('tranquillity', 'anquillitytray'), ('humiliations',
'umiliationshay'), ('contributing', 'ontributingcay'), ('significancy',
'ignificancysay'), ('wretchedness', 'etchednesswray'), ('prodigiously',
'odigiouslypray'), ('particularly', 'articularlypay'), ('carelessness',
'arelessnesscay'), ('refreshments', 'efreshmentsray'), ('satisfaction',
'atisfactionsay'), ('construction', 'onstructioncay'), ('contribution',
'ontributioncay'), ('gracefulness', 'acefulnessgray'), ('consequences',
'onsequencescay'), ('grandmothers', 'andmothersgray'), ('proportioned',
'oportionedpray'), ('perverseness', 'erversenesspay'), ('conciliation',
'onciliationcay'), ('representing', 'epresentingray'), ('watchfulness',
'atchfulnessway'), ('recollecting', 'ecollectingray'), ('hesitatingly',
'esitatinglyhay'), ('pleasantness', 'easantnessplay'), ('benefactress',
'enefactressbay'), ('distractedly', 'istractedlyday'), ('perseverance',
'erseverancepay'), ('performances', 'erformancespay'), ('disapproving',
'isapprovingday'), ('recollection', 'ecollectionray'), ('comprehended',
'omprehendedcay'), ('relinquished', 'elinquishedray'), ('conjecturing',
'onjecturingcay'), ('reservedness', 'eservednessray'), ('promontories',
'omontoriespray'), ('discontented', 'iscontentedday'), ('persecutions',
'ersecutionspay'), ('relationship', 'elationshipray'), ('triumphantly',
'iumphantlytray'), ('deliberation', 'eliberationday'), ('supplication',
'upplicationsay'), ('dishonorable', 'ishonorableday'), ('propensities',
'opensitiespray'), ('contemptuous', 'ontemptuouscay'), ('transporting',
'ansportingtray')], (13, 15): [('distinguished', 'istinguishedday'),
('resuscitation', 'esuscitationray'), ('particularity', 'articularitypay'),
('superannuated', 'uperannuatedsay'), ('disagreements', 'isagreementsday'),
('gentlemanlike', 'entlemanlikegay'), ('prepossessing', 'epossessingpray'),
('conscientious', 'onscientiouscay'), ('considerately', 'onsideratelycay'),
('felicitations', 'elicitationsfay'), ('prepossession', 'epossessionpray'),
('consternation', 'onsternationcay'), ('disproportion', 'isproportionday'),
('commiseration', 'ommiserationcay'), ('mortification', 'ortificationmay'),
('solicitations', 'olicitationssay'), ('qualification', 'alificationquay'),
('philanthropic', 'ilanthropicphay'), ('disappointing', 'isappointingday'),
('prescriptions', 'escriptionspray'), ('strengthening', 'engtheningstray')],
(13, 17): [('mulberry-tree', 'ulberrymay-eetray'), ('good-humoured', 'oodgay-
umouredhay'), ('mean-spirited', 'eanmay-iritedspay'), ('self-reproach', 'elfsay-
eproachray'), ('drawing-table', 'awingdray-abletay'), ('well-disposed', 'ellway-
isposedday'), ('proud-looking', 'oudpray-ookinglay'), ('flower-garden',
'owerflay-ardengay'), ('mansion-house', 'ansionmay-ousehay'), ('landing-place',
'andinglay-aceplay'), ('music-sellers', 'usicmay-ellerssay')], (4, 7): [('acts',
'actsway'), ('anne', 'anneway'), ('upon', 'uponway'), ('able', 'ableway'),
('inch', 'inchway'), ('ages', 'agesway'), ('envy', 'envyway'), ('idea',
'ideaway'), ('ends', 'endsway'), ('-and', '-andway'), ('eyed', 'eyedway'),
('arms', 'armsway'), ('over', 'overway'), ('easy', 'easyway'), ('ever',
'everway'), ('ours', 'oursway'), ('ears', 'earsway'), ('away', 'awayway')], (16,
20): [('pleasure-grounds', 'easureplay-oundsgray'), ('self-provocation',
'elfsay-ovocationpray')], (11, 15): [('bed-chamber', 'edbay-amberchay'), ('high-
minded', 'ighhay-indedmay'), ('before-hand', 'eforebay-andhay'), ('watch-tower',

'atchway-owertay'), ('print-shops', 'intpray-opsshay'), ('card-tables', 'ardcay-ablestay'), ('post-horses', 'ostpay-orseshay'), ('good-nature', 'oodgay-aturenay'), ('dining-room', 'iningday-oomray'), ('high-church', 'ighhay-urchchay'), ('well-wisher', 'ellway-isherway'), ('bond-street', 'ondbay-eetstray'), ('carpet-work', 'arpetcay-orkway')], (2, 4): [('to', 'otay'), ('he', 'ehay'), ('no', 'onay'), ('me', 'emay')], (12, 16): [('good-hearted', 'oodgay-eartedhay'), ('well-judging', 'ellway-udgingjay'), ('drawing-room', 'awingdray-oomray'), ('common-place', 'ommoncay-aceplay'), ('self-control', 'elfsay-ontrolcay'), ('hand-writing', 'andhay-itingwray'), ('well-behaved', 'ellway-ehavedbay'), ('mantel-piece', 'antelmay-iecepay')], (10, 15): [('ill-humour', 'illway-umourhay'), ('ill-judged', 'illway-udgedjay'), ('ill-suited', 'illway-uitedsay')], (1, 3): [('c', 'cay')], (8, 13): [('ill-wind', 'illway-indway'), ('ill-will', 'illway-illway'), ('over-set', 'overway-etsay')], (17, 19): [('disqualifications', 'isqualificationsday')], (9, 16): [('by-and-by', 'ybay-andway-ybay')], (3, 6): [('aye', 'ayeway'), ('use', 'useway'), ('owe', 'oweway'), ('our', 'ourway'), ('old', 'oldway'), ('ear', 'earway'), ('its', 'itsway'), ('ill', 'illway')], (12, 17): [('ill-breeding', 'illway-eedingbray'), ('re-assurance', 'eray-assuranceway'), ('over-powered', 'overway-oweredpay'), ('ill-disposed', 'illway-isposedday')], (11, 16): [('ill-natured', 'illway-aturednay')], (14, 16): [('reconciliation', 'econciliationray'), ('recommendation', 'ecommendationray'), ('reasonableness', 'easonablenessray'), ('disinclination', 'isinclinationday'), ('respectability', 'espectabilityray'), ('circumspection', 'ircumspectioncay'), ('considerations', 'onsiderationscay'), ('qualifications', 'alificationsquay'), ('congratulating', 'ongratulatingcay')], (15, 17): [('dissatisfaction', 'issatisfactionday'), ('disrespectfully', 'isrespectfullyday'), ('conscientiously', 'onscientiouslycay'), ('misconstruction', 'isconstructionmay')], (15, 18): [('instantaneously', 'instantaneouslyway')], (15, 19): [('shooting-jacket', 'ootingshay-acketjay')], (14, 21): [('sisters-in-law', 'isterssay-inway-awlay')], (17, 21): [('self-condemnation', 'elfsay-ondemnationcay')], (13, 20): [('sister-in-law', 'istersay-inway-awlay')], (9, 14): [('ill-timed', 'illway-imedtay')], (13, 18): [('ill-treatment', 'illway-eatmenttray'), ('old-fashioned', 'oldway-ashionedfay'), ('self-interest', 'elfsay-interestway')], (16, 22): [('good-for-nothing', 'oodgay-orfay-othingnay')], (9, 13): [('dove-cote', 'oveday-otecay'), ('tea-table', 'eatay-abletay'), ('work-bags', 'orkway-agsbay')], (1, 4): [('i', 'iway')], (16, 19): [('incomprehensible', 'incomprehensibleway')], (8, 12): [('to-night', 'otay-ightnay'), ('land-tax', 'andlay-axtay')], (11, 18): [('tete-a-tete', 'etetay-away-etetay')], (5, 9): [('to-do', 'otay-oday')]})
Moved models to GPU!
Epoch:  0 | Flops (T): 0.091202679072, Tokens (M) 0.052678, Train Loss 2.919, Val Loss 2.908 | Gen: aaaaaaaaaaaaaaaaaaaa
Epoch:  1 | Flops (T): 0.182405358144, Tokens (M) 0.105356, Train Loss 2.767, Val Loss 2.693 | Gen: fffffffffffffffffff
Epoch:  2 | Flops (T): 0.273608037216, Tokens (M) 0.158034, Train Loss 2.553, Val Loss 2.599 | Gen: fffffyyyyyyy
Epoch:  3 | Flops (T): 0.364810716288, Tokens (M) 0.210712, Train Loss 2.479, Val Loss 2.602 | Gen: ffaaaaaaaaaaaaayyyyy
Epoch:  4 | Flops (T): 0.45601339536, Tokens (M) 0.26339, Train Loss 2.409, Val

```
Loss 2.547 | Gen: ffffffaayyyyyyy
Epoch:   5 | Flops (T): 0.547216074432, Tokens (M) 0.316068, Train Loss 2.290,
Val Loss 2.526 | Gen: ffffffaayyyyyyyyy
Epoch:   6 | Flops (T): 0.638418753504, Tokens (M) 0.368746, Train Loss 2.297,
Val Loss 2.486 | Gen: ooooooaaaaaaaayyyyyy
Epoch:   7 | Flops (T): 0.729621432576, Tokens (M) 0.421424, Train Loss 2.259,
Val Loss 2.325 | Gen: aayyyy
Epoch:   8 | Flops (T): 0.820824111648, Tokens (M) 0.474102, Train Loss 2.163,
Val Loss 2.275 | Gen: ooooooaaayyyyyyyyyyy
Epoch:   9 | Flops (T): 0.91202679072, Tokens (M) 0.52678, Train Loss 2.156, Val
Loss 2.212 | Gen: -o-o----ooyyyyyyyyyy
Epoch:  10 | Flops (T): 1.003229469792, Tokens (M) 0.579458, Train Loss 2.044,
Val Loss 2.111 | Gen: ol--o---------------
Epoch:  11 | Flops (T): 1.094432148864, Tokens (M) 0.632136, Train Loss 1.938,
Val Loss 2.236 | Gen: ll------------------
Epoch:  12 | Flops (T): 1.185634827936, Tokens (M) 0.684814, Train Loss 1.988,
Val Loss 2.151 | Gen: olf-----------------
Epoch:  13 | Flops (T): 1.276837507008, Tokens (M) 0.737492, Train Loss 1.975,
Val Loss 2.088 | Gen: oofffoayyyyyyyyyyyy
Epoch:  14 | Flops (T): 1.36804018608, Tokens (M) 0.79017, Train Loss 2.007, Val
Loss 2.050 | Gen: ooffffy
Epoch:  15 | Flops (T): 1.459242865152, Tokens (M) 0.842848, Train Loss 1.770,
Val Loss 1.839 | Gen: ooffffa
Epoch:  16 | Flops (T): 1.550445544224, Tokens (M) 0.895526, Train Loss 1.840,
Val Loss 2.076 | Gen: offffffay
Epoch:  17 | Flops (T): 1.641648223296, Tokens (M) 0.948204, Train Loss 1.868,
Val Loss 1.905 | Gen: ooffffay
Epoch:  18 | Flops (T): 1.732850902368, Tokens (M) 1.000882, Train Loss 1.707,
Val Loss 2.211 | Gen: oooffffaa
Epoch:  19 | Flops (T): 1.82405358144, Tokens (M) 1.05356, Train Loss 1.788, Val
Loss 2.008 | Gen: oooofffaay
Epoch:  20 | Flops (T): 1.915256260512, Tokens (M) 1.106238, Train Loss 1.817,
Val Loss 2.016 | Gen: ooooffffa
Epoch:  21 | Flops (T): 2.006458939584, Tokens (M) 1.158916, Train Loss 1.744,
Val Loss 1.779 | Gen: oooffffay
Epoch:  22 | Flops (T): 2.097661618656, Tokens (M) 1.211594, Train Loss 1.625,
Val Loss 1.719 | Gen: ooooroay
Epoch:  23 | Flops (T): 2.188864297728, Tokens (M) 1.264272, Train Loss 1.618,
Val Loss 1.748 | Gen: oooofffa
Epoch:  24 | Flops (T): 2.2800669768, Tokens (M) 1.31695, Train Loss 1.590, Val
Loss 2.180 | Gen: oooffay
Epoch:  25 | Flops (T): 2.371269655872, Tokens (M) 1.369628, Train Loss 1.535,
Val Loss 1.689 | Gen: soofffay
Epoch:  26 | Flops (T): 2.462472334944, Tokens (M) 1.422306, Train Loss 1.495,
Val Loss 1.690 | Gen: ooooffaa
Epoch:  27 | Flops (T): 2.553675014016, Tokens (M) 1.474984, Train Loss 1.487,
Val Loss 2.014 | Gen: ooorfffayyy
Epoch:  28 | Flops (T): 2.644877693088, Tokens (M) 1.527662, Train Loss 1.537,
```

```
Val Loss 1.517 | Gen: sooorraa
Epoch:  29 | Flops (T): 2.73608037216, Tokens (M) 1.58034, Train Loss 1.314, Val
Loss 1.578 | Gen: iooofay
Epoch:  30 | Flops (T): 2.827283051232, Tokens (M) 1.633018, Train Loss 1.341,
Val Loss 1.703 | Gen: aoorfffay
Epoch:  31 | Flops (T): 2.918485730304, Tokens (M) 1.685696, Train Loss 1.370,
Val Loss 1.643 | Gen: koorrfaay
Epoch:  32 | Flops (T): 3.009688409376, Tokens (M) 1.738374, Train Loss 1.422,
Val Loss 1.495 | Gen: aooofay
Epoch:  33 | Flops (T): 3.100891088448, Tokens (M) 1.791052, Train Loss 1.324,
Val Loss 1.765 | Gen: ooorfaay
Epoch:  34 | Flops (T): 3.19209376752, Tokens (M) 1.84373, Train Loss 1.276, Val
Loss 1.662 | Gen: oooorffayy
Epoch:  35 | Flops (T): 3.283296446592, Tokens (M) 1.896408, Train Loss 1.278,
Val Loss 1.468 | Gen: woooofayy
Epoch:  36 | Flops (T): 3.374499125664, Tokens (M) 1.949086, Train Loss 1.283,
Val Loss 1.598 | Gen: goooofaayy
Epoch:  37 | Flops (T): 3.465701804736, Tokens (M) 2.001764, Train Loss 1.322,
Val Loss 1.479 | Gen: iooooaayy
Epoch:  38 | Flops (T): 3.556904483808, Tokens (M) 2.054442, Train Loss 1.305,
Val Loss 1.478 | Gen: oooorwaay
Epoch:  39 | Flops (T): 3.64810716288, Tokens (M) 2.10712, Train Loss 1.290, Val
Loss 1.506 | Gen: ooooofay
Epoch:  40 | Flops (T): 3.739309841952, Tokens (M) 2.159798, Train Loss 1.211,
Val Loss 1.360 | Gen: oooofay
Epoch:  41 | Flops (T): 3.830512521024, Tokens (M) 2.212476, Train Loss 1.250,
Val Loss 1.496 | Gen: ooorffay
Epoch:  42 | Flops (T): 3.921715200096, Tokens (M) 2.265154, Train Loss 1.201,
Val Loss 1.637 | Gen: wooofaay
Epoch:  43 | Flops (T): 4.012917879168, Tokens (M) 2.317832, Train Loss 1.153,
Val Loss 2.258 | Gen: wooofay
Epoch:  44 | Flops (T): 4.10412055824, Tokens (M) 2.37051, Train Loss 1.182, Val
Loss 1.659 | Gen: soorfaay
Epoch:  45 | Flops (T): 4.195323237312, Tokens (M) 2.423188, Train Loss 1.074,
Val Loss 1.310 | Gen: oooofay
Epoch:  46 | Flops (T): 4.286525916384, Tokens (M) 2.475866, Train Loss 1.184,
Val Loss 2.230 | Gen: oooofayy
Epoch:  47 | Flops (T): 4.377728595456, Tokens (M) 2.528544, Train Loss 1.111,
Val Loss 1.364 | Gen: oooofay
Epoch:  48 | Flops (T): 4.468931274528, Tokens (M) 2.581222, Train Loss 1.022,
Val Loss 1.647 | Gen: oooofaay
Epoch:  49 | Flops (T): 4.5601339536, Tokens (M) 2.6339, Train Loss 0.999, Val
Loss 1.585 | Gen: oooofayy
Epoch:  50 | Flops (T): 4.651336632672, Tokens (M) 2.686578, Train Loss 1.008,
Val Loss 2.259 | Gen: wooofay
Epoch:  51 | Flops (T): 4.742539311744, Tokens (M) 2.739256, Train Loss 1.055,
Val Loss 1.509 | Gen: iloofaay
Epoch:  52 | Flops (T): 4.833741990816, Tokens (M) 2.791934, Train Loss 0.898,
```

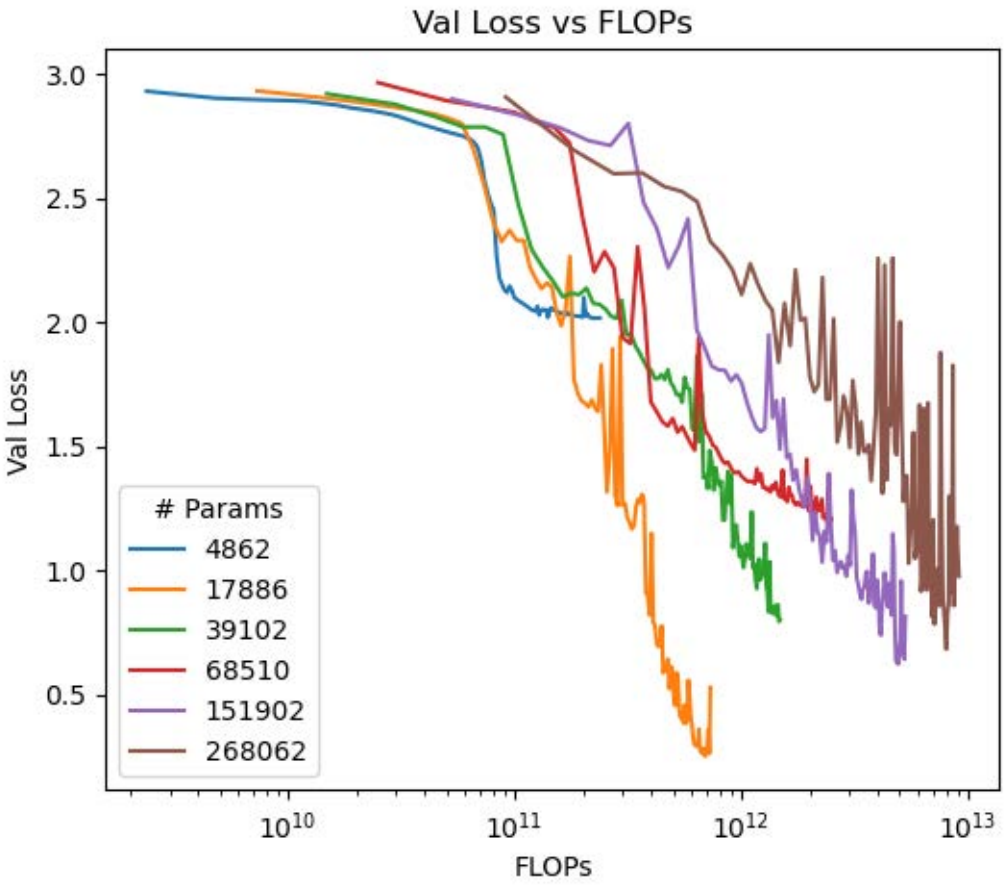
```
Val Loss 1.465 | Gen: iooofaa
Epoch:  53 | Flops (T): 4.924944669888, Tokens (M) 2.844612, Train Loss 0.823,
Val Loss 1.720 | Gen: wooofay
Epoch:  54 | Flops (T): 5.01614734896, Tokens (M) 2.89729, Train Loss 1.132, Val
Loss 2.002 | Gen: oorrfffay
Epoch:  55 | Flops (T): 5.107350028032, Tokens (M) 2.949968, Train Loss 1.224,
Val Loss 1.562 | Gen: fforffayy
Epoch:  56 | Flops (T): 5.198552707104, Tokens (M) 3.002646, Train Loss 1.055,
Val Loss 1.278 | Gen: ooorffay
Epoch:  57 | Flops (T): 5.289755386176, Tokens (M) 3.055324, Train Loss 0.920,
Val Loss 1.381 | Gen: ooorffaay
Epoch:  58 | Flops (T): 5.380958065248, Tokens (M) 3.108002, Train Loss 0.936,
Val Loss 1.380 | Gen: ooorfay
Epoch:  59 | Flops (T): 5.47216074432, Tokens (M) 3.16068, Train Loss 0.807, Val
Loss 1.028 | Gen: ooorffay
Epoch:  60 | Flops (T): 5.563363423392, Tokens (M) 3.213358, Train Loss 0.739,
Val Loss 1.155 | Gen: ooorrfaay
Epoch:  61 | Flops (T): 5.654566102464, Tokens (M) 3.266036, Train Loss 0.751,
Val Loss 1.255 | Gen: ioorffayy
Epoch:  62 | Flops (T): 5.745768781536, Tokens (M) 3.318714, Train Loss 0.765,
Val Loss 1.555 | Gen: aoorfay
Epoch:  63 | Flops (T): 5.836971460608, Tokens (M) 3.371392, Train Loss 0.682,
Val Loss 1.047 | Gen: ooorffay
Epoch:  64 | Flops (T): 5.92817413968, Tokens (M) 3.42407, Train Loss 0.628, Val
Loss 1.096 | Gen: ooorrfayy
Epoch:  65 | Flops (T): 6.019376818752, Tokens (M) 3.476748, Train Loss 0.778,
Val Loss 1.076 | Gen: ooooffay
Epoch:  66 | Flops (T): 6.110579497824, Tokens (M) 3.529426, Train Loss 0.923,
Val Loss 1.667 | Gen: ooorfayy
Epoch:  67 | Flops (T): 6.201782176896, Tokens (M) 3.582104, Train Loss 0.707,
Val Loss 0.915 | Gen: aoorffay
Epoch:  68 | Flops (T): 6.292984855968, Tokens (M) 3.634782, Train Loss 0.589,
Val Loss 0.966 | Gen: ooorfaay
Epoch:  69 | Flops (T): 6.38418753504, Tokens (M) 3.68746, Train Loss 0.697, Val
Loss 1.652 | Gen: ooorfay
Epoch:  70 | Flops (T): 6.475390214112, Tokens (M) 3.740138, Train Loss 0.690,
Val Loss 1.021 | Gen: aoorffayy
Epoch:  71 | Flops (T): 6.566592893184, Tokens (M) 3.792816, Train Loss 0.686,
Val Loss 0.920 | Gen: soorffay
Epoch:  72 | Flops (T): 6.657795572256, Tokens (M) 3.845494, Train Loss 0.891,
Val Loss 1.675 | Gen: aoorffay
Epoch:  73 | Flops (T): 6.748998251328, Tokens (M) 3.898172, Train Loss 0.645,
Val Loss 0.956 | Gen: ioorffay
Epoch:  74 | Flops (T): 6.8402009304, Tokens (M) 3.95085, Train Loss 0.575, Val
Loss 1.081 | Gen: aoorfay
Epoch:  75 | Flops (T): 6.931403609472, Tokens (M) 4.003528, Train Loss 0.594,
Val Loss 0.814 | Gen: aoorffay
Epoch:  76 | Flops (T): 7.022606288544, Tokens (M) 4.056206, Train Loss 0.521,
```

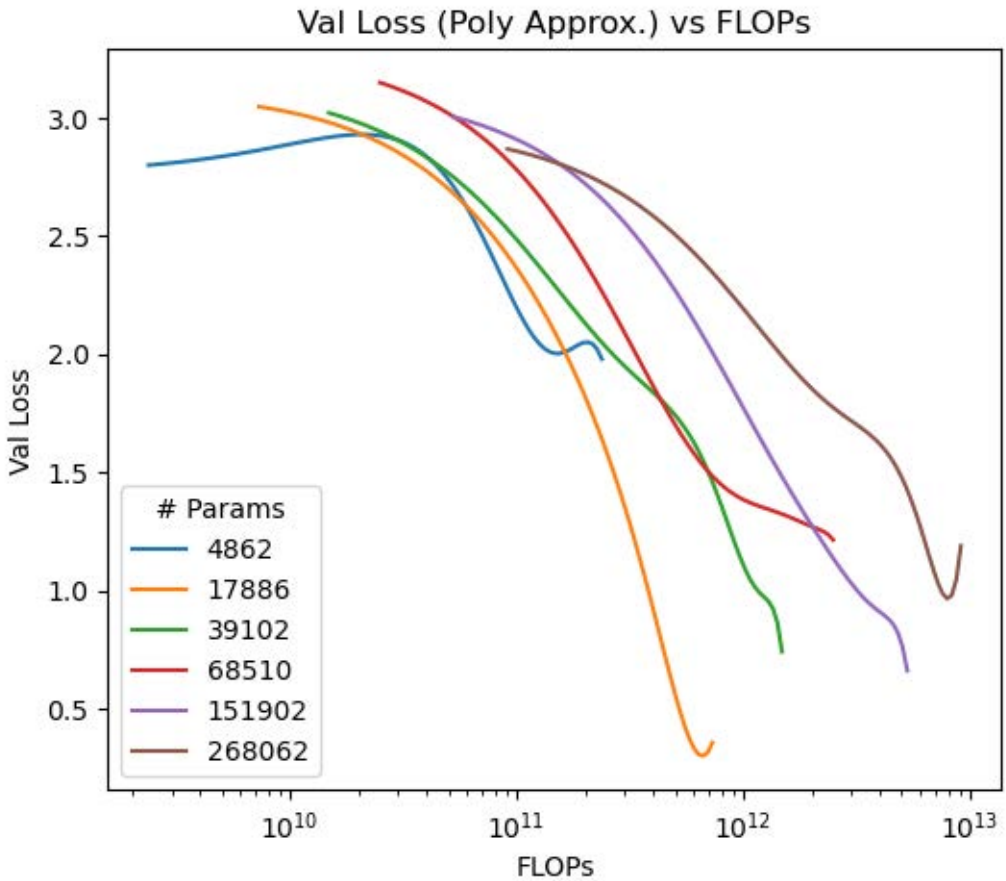
```
Val Loss 1.203 | Gen: aoorffa
Epoch:  77 | Flops (T): 7.113808967616, Tokens (M) 4.108884, Train Loss 0.541,
Val Loss 0.783 | Gen: aoorrfay
Epoch:  78 | Flops (T): 7.205011646688, Tokens (M) 4.161562, Train Loss 0.503,
Val Loss 0.864 | Gen: aoorffay
Epoch:  79 | Flops (T): 7.29621432576, Tokens (M) 4.21424, Train Loss 0.483, Val
Loss 0.867 | Gen: aoorffay
Epoch:  80 | Flops (T): 7.387417004832, Tokens (M) 4.266918, Train Loss 0.502,
Val Loss 1.009 | Gen: aoorfay
Epoch:  81 | Flops (T): 7.478619683904, Tokens (M) 4.319596, Train Loss 0.509,
Val Loss 0.857 | Gen: aoorrfay
Epoch:  82 | Flops (T): 7.569822362976, Tokens (M) 4.372274, Train Loss 0.743,
Val Loss 1.878 | Gen: ooorrfaay
Epoch:  83 | Flops (T): 7.661025042048, Tokens (M) 4.424952, Train Loss 1.102,
Val Loss 1.208 | Gen: aoorffay
Epoch:  84 | Flops (T): 7.75222772112, Tokens (M) 4.47763, Train Loss 0.600, Val
Loss 0.887 | Gen: aoooffay
Epoch:  85 | Flops (T): 7.843430400192, Tokens (M) 4.530308, Train Loss 0.522,
Val Loss 0.827 | Gen: aoorffay
Epoch:  86 | Flops (T): 7.934633079264, Tokens (M) 4.582986, Train Loss 0.449,
Val Loss 0.759 | Gen: aoorffay
Epoch:  87 | Flops (T): 8.025835758336, Tokens (M) 4.635664, Train Loss 0.416,
Val Loss 0.682 | Gen: ioorffay
Epoch:  88 | Flops (T): 8.117038437408, Tokens (M) 4.688342, Train Loss 0.371,
Val Loss 0.830 | Gen: aoorffay
Epoch:  89 | Flops (T): 8.20824111648, Tokens (M) 4.74102, Train Loss 0.459, Val
Loss 0.860 | Gen: aoorffay
Epoch:  90 | Flops (T): 8.299443795552, Tokens (M) 4.793698, Train Loss 0.408,
Val Loss 1.299 | Gen: aoorrfay
Epoch:  91 | Flops (T): 8.390646474624, Tokens (M) 4.846376, Train Loss 0.571,
Val Loss 1.017 | Gen: aoorffay
Epoch:  92 | Flops (T): 8.481849153696, Tokens (M) 4.899054, Train Loss 0.498,
Val Loss 0.910 | Gen: aoorffay
Epoch:  93 | Flops (T): 8.573051832768, Tokens (M) 4.951732, Train Loss 0.765,
Val Loss 1.826 | Gen: moorrfay
Epoch:  94 | Flops (T): 8.66425451184, Tokens (M) 5.00441, Train Loss 0.802, Val
Loss 0.857 | Gen: ooorffay
Epoch:  95 | Flops (T): 8.755457190912, Tokens (M) 5.057088, Train Loss 0.497,
Val Loss 1.008 | Gen: moorrfay
Epoch:  96 | Flops (T): 8.846659869984, Tokens (M) 5.109766, Train Loss 0.536,
Val Loss 1.079 | Gen: moorrfay
Epoch:  97 | Flops (T): 8.937862549056, Tokens (M) 5.162444, Train Loss 0.480,
Val Loss 1.176 | Gen: ioorrfay
Epoch:  98 | Flops (T): 9.029065228128, Tokens (M) 5.215122, Train Loss 0.447,
Val Loss 1.062 | Gen: aoorrfay
Epoch:  99 | Flops (T): 9.1202679072, Tokens (M) 5.2678, Train Loss 0.433, Val
Loss 0.976 | Gen: ioorffay
Obtained lowest validation loss of: 0.6818617564994235
```

```
Avg tokens per example (Seq Len): 20.59343236903831, Avg flops per example:
35653901.12275215
```

[ ]:
```
def getf_loss_flop(data, degree=5):
    x, y = data[3], data[1]
    p = np.polyfit(x, y, degree)
    return p

def plot_loss_flop_o(data_list, num_params, degree=5):
    fig, ax = plt.subplots(figsize=(6, 5))
    for data, params in zip(data_list, num_params):
        x, y = data[3], data[1]
        ax.plot(x, y, label=f'{params}')

    ax.legend(title='# Params', loc = "lower left")
    plt.xscale('log')
    plt.xlabel('FLOPs')
    plt.ylabel('Val Loss')
    plt.title('Val Loss vs FLOPs')

def plot_loss_flop(data_list, num_params, degree=5):
    fig, ax = plt.subplots(figsize=(6, 5))
    for data, params in zip(data_list, num_params):
        x, y = data[3], data[1]
        p = np.polyfit(x, y, degree)
        x_min, x_max = min(x), max(x)
        px = np.logspace(np.log10(x_min), np.log10(x_max), 100)
        py = np.polyval(p, px)
        ax.plot(px, py, label=f'{params}')

    ax.legend(title='# Params', loc = "lower left")
    plt.xscale('log')
    plt.xlabel('FLOPs')
    plt.ylabel('Val Loss')
    plt.title('Val Loss (Poly Approx.) vs FLOPs')
```

[ ]:
```
col_names = ['Train loss', 'Val loss', 'Tokens', 'Flops', 'Number of Examples',␣
 ↪'Number of Parameters']

hidden_list = [16, 32, 48, 64, 96, 128]
num_params = [transformer_params(30, hidden, 4) for hidden in hidden_list]
data_list = [trans16_losses_s, trans32_losses_s, trans48_losses_s,␣
 ↪trans64_losses_s, trans96_losses_s, trans128_losses_s]

plot_loss_flop_o(data_list, num_params)
plot_loss_flop(data_list, num_params)
```


Val Loss vs FLOPs
Val Loss
FLOPs
3.0
2.5
2.0
1.5
1.0
0.5
$10^{10}$
$10^{11}$
$10^{12}$
$10^{13}$
# Params
4862
17886
39102
68510
151902
268062



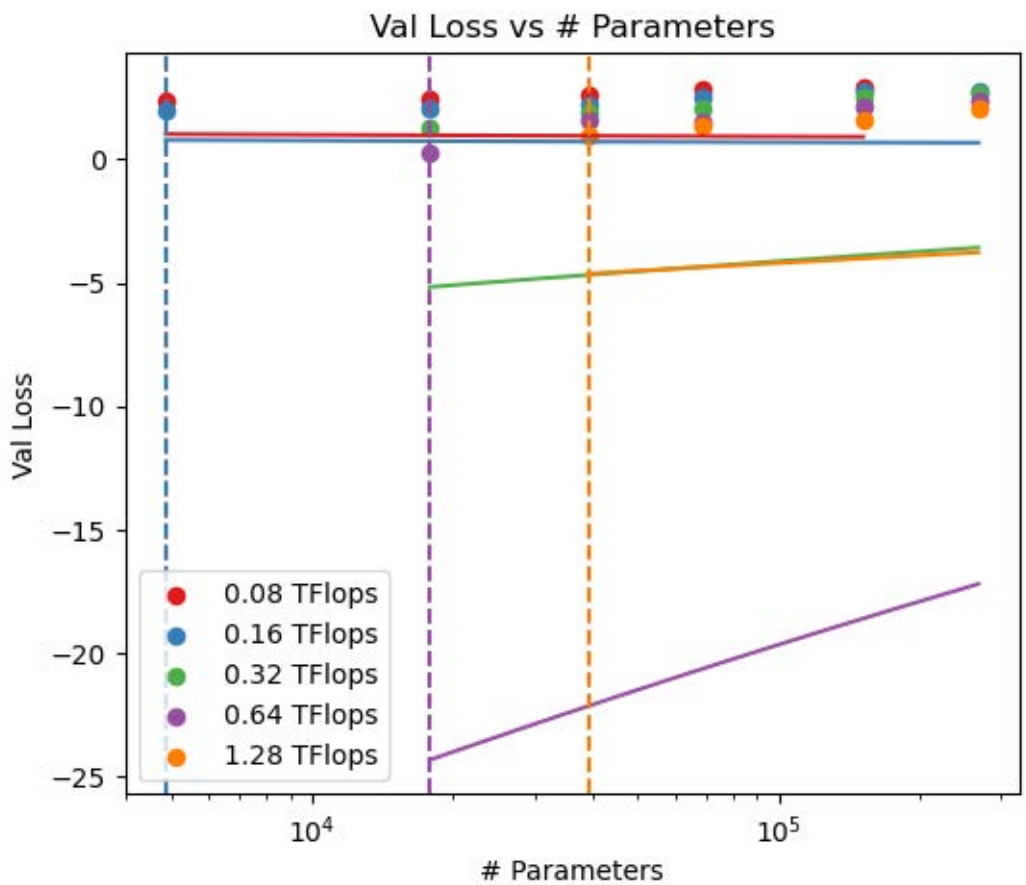

### 6.2 Step 2: IsoFLOP profiles

[ ]:
```
def get_scatter_data(data_list, num_params, f_list, tflops):
    x, y = [], []
    for i, f in enumerate(f_list):
        if tflops <= max(data_list[i][3]) and tflops >= min(data_list[i][3]):
            x.append(num_params[i])
            y.append(np.polyval(f, tflops))
    return x, y


def find_optimal_params(x, y):
    # Fit a quadratic function to the validation loss and number of parameters␣
 ↪in the log space
    p = np.polyfit(np.log(x), np.log(y), 2)

    # Find the optimal number of parameters using the quadratic function
    optimal_params_index = np.argmin(y)
```

```
    optimal_params = x[optimal_params_index]

    return p, optimal_params
```

[ ]:
```
def plot_isoflop(data_list, num_params, degree=5):
    colors = ['#E41A1C',␣
↪'#377EB8','#4DAF4A','#984EA3','#FF7F00','#FFFF33','#A65628']
    target_flops = [8e10, 16e10, 32e10, 64e10, 128e10]
    f_list = [getf_loss_flop(data, degree=5) for data in data_list]

    fig, ax = plt.subplots(figsize=(6, 5))
    for fidx, tflops in enumerate(target_flops):
        x, y = get_scatter_data(data_list, num_params, f_list, tflops)
        p, optimal_params = find_optimal_params(x, y)

        ax.scatter(np.array(x), np.array(y), label=f'{tflops/1e12} TFlops',␣
↪color=colors[fidx])
        px = np.logspace(np.log10(x[0]), np.log10(x[-1]), 100)
        py = np.polyval(p, np.log10(px))
        ax.plot(px, py, color=colors[fidx])
        ax.axvline(x=optimal_params, color=colors[fidx], linestyle='--')

    ax.legend(loc = "lower left")
    plt.xscale('log')
    plt.xlabel('# Parameters')
    plt.ylabel('Val Loss')
    plt.title('Val Loss vs # Parameters')
```

[ ]:
```
plot_isoflop(data_list, num_params)
```




### 6.3 Step 3: Parameters vs FLOPs

[ ]:
```
def fit_linear_log(x, y):
    # Fit a linear line to the target FLOPs and the optimal model parameters in␣
↪the log space
    m, c = np.polyfit(np.log10(x), np.log10(y), 1)
    return m, c
```

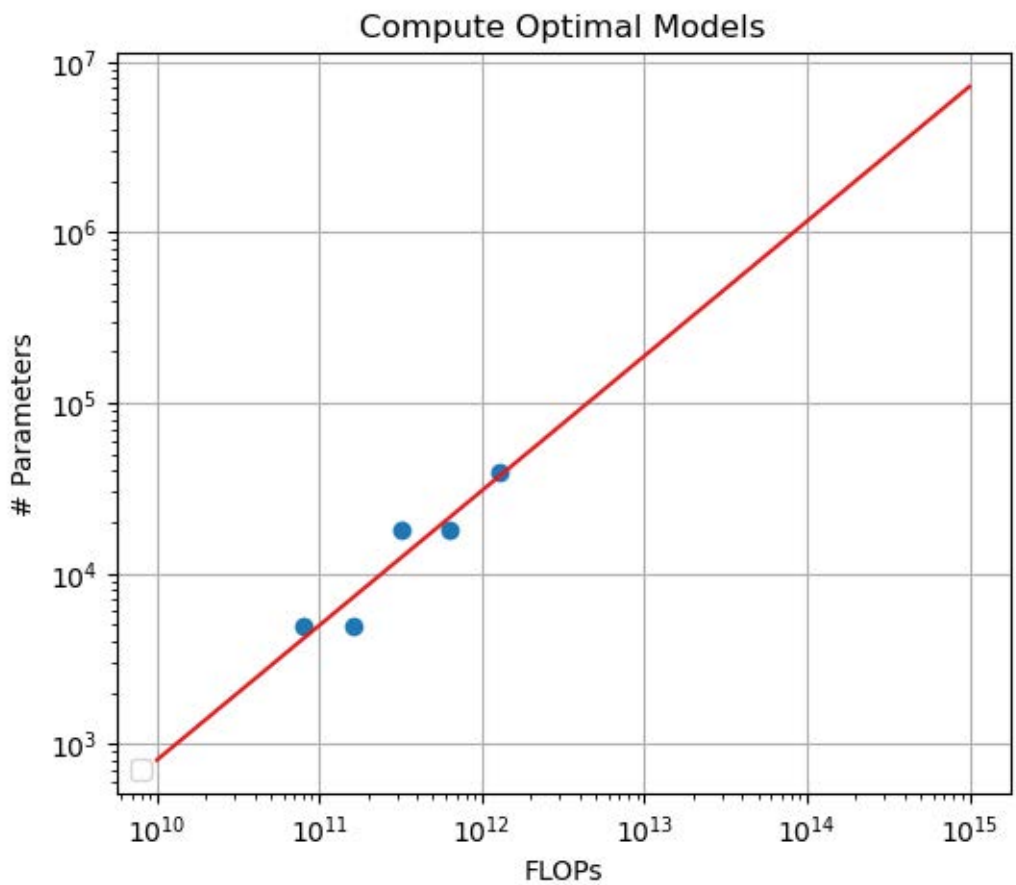
[ ]:
```
def plot_flops_params(data_list, num_params):
    colors = ['#E41A1C',␣
↪'#377EB8','#4DAF4A','#984EA3','#FF7F00','#FFFF33','#A65628']
    target_flops = [8e10, 16e10, 32e10, 64e10, 128e10]
    f_list = [getf_loss_flop(data, degree=5) for data in data_list]

    fig, ax = plt.subplots(figsize=(6, 5))
    params = []
    for fidx, tflops in enumerate(target_flops):
        x, y = get_scatter_data(data_list, num_params, f_list, tflops)
```

```
        p, optimal_params = find_optimal_params(x, y)
        params.append(optimal_params)

    ax.scatter(target_flops, params)
    m, c = fit_linear_log(target_flops, params)
    print(f"y = {m}x + {c}")

    lx = np.logspace(np.log10(1e10), np.log10(1e15), 100)
    ly = 10**(np.log10(lx) * m + c)
    ax.plot(lx, ly, color = colors[0])

    ax.legend(loc = "lower left")
    plt.xscale('log')
    plt.yscale('log')
    plt.xlabel('FLOPs')
    plt.ylabel('# Parameters')
    plt.xlim()
    plt.title('Compute Optimal Models')
    plt.grid()
```

[ ]:
```
plot_flops_params(data_list, num_params)
```

```
No artists with labels found to put in legend.  Note that artists whose label
start with an underscore are ignored when legend() is called with no argument.
```

```
y = 0.7894450187301238x + -4.988512273482354
```




### 6.4 Step 4: Tokens vs FLOPs

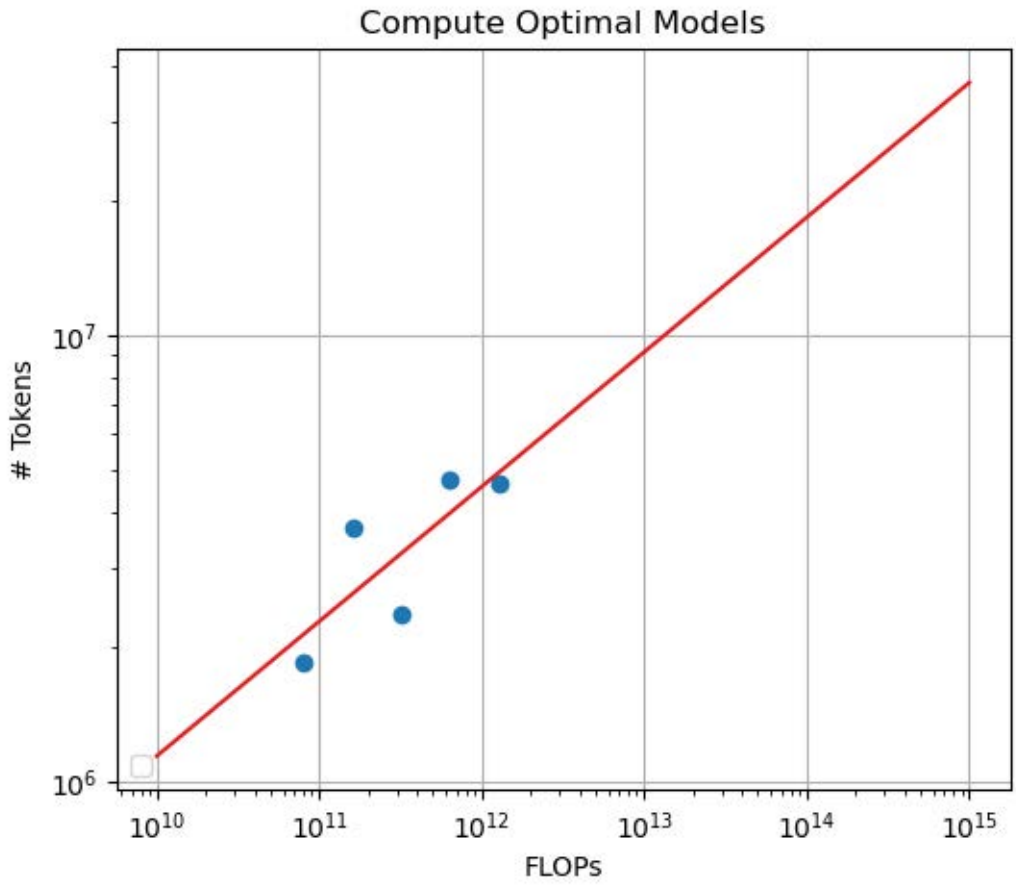
[ ]:
```
from scipy.optimize import bisect

def find_hidden(params, x_min=16, x_max=1e4, tol=1e-4):
    f = lambda h: transformer_params(vocab_size=30, hidden_size=h, num_layers=4)
    # Define the function to find the root of
    def g(x):
        return f(x) - params

    # Find the root of the function using the bisection method
    x_root = bisect(g, x_min, x_max, rtol=tol)
    return x_root

def find_tokens(params, flops, seq_len):
    hidden = find_hidden(params)
    flops_per_example = transformer_flops(seq_len=seq_len, vocab_size=30,
                                          hidden_size=hidden, num_layers=4)
```

```
    return flops/flops_per_example * seq_len

def plot_flops_tokens(data_list, num_params):
    seq_len = 19.55
    colors = ['#E41A1C',␣
 ↪'#377EB8','#4DAF4A','#984EA3','#FF7F00','#FFFF33','#A65628']
    target_flops = [8e10, 16e10, 32e10, 64e10, 128e10]
    f_list = [getf_loss_flop(data, degree=5) for data in data_list]

    fig, ax = plt.subplots(figsize=(6, 5))
    params = []
    tokens = []
    for fidx, tflops in enumerate(target_flops):
        x, y = get_scatter_data(data_list, num_params, f_list, tflops)
        p, optimal_params = find_optimal_params(x, y)
        params.append(optimal_params)
        tokens.append(find_tokens(optimal_params, tflops, seq_len))

    ax.scatter(target_flops, tokens)

    m, c = fit_linear_log(target_flops, tokens)
    print(f"y = {m}x + {c}")

    lx = np.logspace(np.log10(1e10), np.log10(1e15), 100)
    ly = 10**(np.log10(lx) * m + c)
    ax.plot(lx, ly, color = colors[0])

    ax.legend(loc = "lower left")
    plt.xscale('log')
    plt.yscale('log')
    plt.xlabel('FLOPs')
    plt.ylabel('# Tokens')
    plt.title('Compute Optimal Models')
    plt.grid()
```

[ ]:
```
plot_flops_tokens(data_list, num_params)
```

```
No artists with labels found to put in legend.  Note that artists whose label
start with an underscore are ignored when legend() is called with no argument.
```

```
y = 0.300954063639963x + 3.0486445647861524
```


Compute Optimal Models
# Tokens
$10^7$
$10^6$
$10^{10}$
$10^{11}$
$10^{12}$
$10^{13}$
$10^{14}$
$10^{15}$
FLOPs

# bert

March 27, 2023

## 1 Colab FAQ

For some basic overview and features offered in Colab notebooks, check out: Overview of Colaboratory Features

You need to use the colab GPU for this assignment by selecting:

**Runtime** → **Change runtime type** → **Hardware Accelerator: GPU**

## 2 Part 4: Fine-tuning pretrained language models

Acknowledgement: This notebook is based on the code from https://mccormickml.com/2019/07/22/BERT-fine-tuning/. Credit to Chris McCormick and Nick Ryan.

### 2.1 Background

Fine-tuning BERT on our task of interest takes some setup. Although these steps are done for you, please take a moment to look through them and make sure you understand their purpose.

Install the HuggingFace Transformers package that contains the pretrained BERT models.

Set the random seeds for reproducibility.

[ ]:
```
import os
import random

import numpy as np
import torch

SEED = 42

torch.manual_seed(SEED)
torch.cuda.manual_seed_all(SEED)
torch.backends.cudnn.deterministic = True
torch.backends.cudnn.benchmark = False
np.random.seed(SEED)
random.seed(SEED)
os.environ['PYTHONHASHSEED'] = str(SEED)
```

```
/home/yuling/miniconda3/envs/cuda11.6/lib/python3.9/site-
packages/tqdm/auto.py:22: TqdmWarning: IProgress not found. Please update
jupyter and ipywidgets. See
https://ipywidgets.readthedocs.io/en/stable/user_install.html
  from .autonotebook import tqdm as notebook_tqdm
```

Run the following cells to download the verbal arithmetic dataset from the CSC413 webpage and load it into a `DataFrame`

[ ]:
```
import wget
import os

print('Downloading verbal arithmetic dataset')

# The URL for the dataset zip file.
url = 'https://csc413-uoft.github.io/2021/assets/misc/'

# Download the file (if we haven't already)
if not os.path.exists('./PA03_data_20_train.csv'):
  wget.download(url + 'PA03_data_20_train.csv', './PA03_data_20_train.csv')
  print('Done downloading training data')
else:
  print('Already downloaded training data')

if not os.path.exists('./PA03_data_20_test.csv'):
  wget.download(url + 'PA03_data_20_test.csv', './PA03_data_20_test.csv')
  print('Done downloading test data')
else:
  print('Already downloaded test data')
```

```
Downloading verbal arithmetic dataset
Already downloaded training data
Already downloaded test data
```

[ ]:
```
import pandas as pd

df = pd.read_csv("./PA03_data_20_train.csv", header=0, names=["index", "input",␣
↪"label"])

print("Number of data points: ", df.shape[0])
sampled = df.sample(10)
# Display 10 random rows from the data.
df.sample(10)
```

```
Number of data points:  640
```

[ ]:

| | index | input | label |
|---|---|---|---|
| 20 | 264 | thirteen plus four | 2 |

```
180    712      fifteen minus twelve      2
270    327        sixteen plus seven      2
221     58         two plus eighteen      2
542    646          twelve minus six      2
143    626          eleven minus six      2
247    759  seventeen minus nineteen      0
467    691     fourteen minus eleven      2
283    372      eighteen plus twelve      2
259    274    thirteen plus fourteen      2
```

### 2.1.1 Tokenizer

To feed our text to BERT, it must be split into tokens, and then these tokens must be mapped to their index in the tokenizer vocabulary. For this we can use the `AutoTokenizer` from the `transformers` library.

As mentioned in the assignment handout, we will use MathBERT, which uses the same architecture as BERT, but has been pretrained on text from pre-kindergarten, high-school, and college graduate level mathematical content.

[ ]:
```
from transformers import AutoTokenizer

bert_tokenizer = AutoTokenizer.from_pretrained('tbs17/MathBERT',␣
↪do_lower_case=True)
```

[ ]:
```
inputs = df.input.values
labels = df.label.values
print("Train data size ", len(inputs))
print('* Original:  ', inputs[0])
# Print the sentence split into tokens.
print('* Tokenized: ', bert_tokenizer.tokenize(inputs[0]))
# Print the sentence mapped to token ids.
print('* Token IDs: ', bert_tokenizer.convert_tokens_to_ids(bert_tokenizer.
↪tokenize(inputs[0])))
```

```
Train data size  640
* Original:   five minus twelve
* Tokenized:  ['five', 'minus', 'twelve']
* Token IDs:  [2274, 15718, 4376]
```

### 2.1.2 Formatting the inputs

In order to use BERT for fine-tuning, we need to format the inputs in a way that matches the inputs of the pretraining step. In short, we need to:

1. Add special tokens to the start and end of each sentence.
2. Pad & truncate all sentences to a single constant length.
3. Explicitly differentiate real tokens from padding tokens with the "attention mask".

**Special Tokens** `[SEP]`

At the end of every sentence, we need to append the special `[SEP]` token.

This token is an artifact of two-sentence tasks, where BERT is given two separate sentences and asked to determine something (e.g., can the answer to the question in sentence A be found in sentence B?).

`[CLS]`

For classification tasks, we must prepend the special `[CLS]` token to the beginning of every sentence.

This token has special significance. BERT consists of 12 Transformer layers. Each transformer takes in a list of token embeddings, and produces the same number of embeddings on the output.

On the output of the final transformer, *only the first embedding (corresponding to the [CLS] token) is used by the classifier.*

> "The first token of every sequence is always a special classification token (`[CLS]`). The final hidden state corresponding to this token is used as the aggregate sequence representation for classification tasks." (from the BERT paper)

Also, because BERT is trained to only use this [CLS] token for classification, we know that the model has been motivated to encode everything it needs for the classification step into that single 768-value embedding vector.

**Sentence Length & Attention Mask** The sentences in our dataset obviously have varying lengths, so how does BERT handle this?

BERT has two constraints: 1. All sentences must be padded or truncated to a single, fixed length. 2. The maximum sentence length is 512 tokens.

Padding is done with a special `[PAD]` token, which is at index 0 in the BERT vocabulary.

The "Attention Mask" is simply an array of 0s and 1s indicating which tokens are padding and which aren't.

In our dataset, all sentences have three word tokens. However, we set the max length of sentence to 7 in this example to show what paddings will be in real world applications.

[ ]:
```
# Set the maximum sequence length.
MAX_LEN = 7

# Print BERTs special PAD token and its index in the vocabulary
print(f'Padding token: "{bert_tokenizer.pad_token}", ID: {bert_tokenizer.
 ↪pad_token_id}')
```

```
Padding token: "[PAD]", ID: 0
```

Luckily, the `BertTokenizer` object from the transformers library makes it easy to preprocess our input text correctly

[ ]:
```
tokenized_inputs = bert_tokenizer(
    inputs.tolist(),          # Input text
```

```
    add_special_tokens=True,  # add '[CLS]' and '[SEP]'
    padding='max_length',     # pad to a length specified by the max_length
    max_length=MAX_LEN,       # truncate all sentences longer than max_length
    return_tensors='pt',      # return everything we need as PyTorch tensors
)

input_ids = tokenized_inputs['input_ids']
attention_masks = tokenized_inputs['attention_mask']

# Print sentence 0, now as a list of IDs.
print('Original: ', tokenized_inputs['input_ids'][0])
print('* Token IDs:', tokenized_inputs['attention_mask'][0])
print('* Tokenized:', bert_tokenizer.decode(tokenized_inputs['input_ids'][0]))
print('* Attention_mask', tokenized_inputs['attention_mask'][0])
```

```
Original:  tensor([  101,  2274, 15718,  4376,   102,     0,     0])
* Token IDs: tensor([1, 1, 1, 1, 1, 0, 0])
* Tokenized: [CLS] five minus twelve [SEP] [PAD] [PAD]
* Attention_mask tensor([1, 1, 1, 1, 1, 0, 0])
```

### 2.1.3 Training & Validation Split

Let's divide up our data into a train set (70%) and a validation set (30%).

We'also create an iterator for our dataset using the torch `DataLoader` class. This helps save on memory during training because, unlike a for loop, with an iterator the entire dataset does not need to be loaded into memory.

[ ]:
```
from sklearn.model_selection import train_test_split
import torch
from torch.utils.data import TensorDataset, DataLoader, RandomSampler,␣
↪SequentialSampler

def train_valid_split(input_ids, attention_masks, labels, batch_size=32):
    # Use 70% for training and 30% for validation.
    train_inputs, validation_inputs,  train_masks, validation_masks,␣
↪train_labels, validation_labels = train_test_split(
        input_ids, attention_masks, labels, random_state=SEED, test_size=0.3,␣
↪stratify=labels
    )

    print('example train_input:    ', train_inputs[0])
    print('example attention_mask: ', train_masks[0])

    train_labels = torch.tensor(train_labels)
    validation_labels = torch.tensor(validation_labels)

    # Create the DataLoader for our training set.
```

```
    train_data = TensorDataset(train_inputs, train_masks, train_labels)
    train_dataloader = DataLoader(train_data, shuffle=True,␣
↪batch_size=batch_size)

    # Create the DataLoader for our validation set.
    validation_data = TensorDataset(validation_inputs, validation_masks,␣
↪validation_labels)
    validation_dataloader = DataLoader(validation_data, shuffle=False,␣
↪batch_size=batch_size)

    return train_dataloader, validation_dataloader
```

[ ]:
```
bert_train_dataloader, bert_validation_dataloader = train_valid_split(
    input_ids=input_ids,
    attention_masks=attention_masks,
    labels=labels,
    batch_size=32
)
```

```
example train_input:     tensor([  101,  2416, 15718,  7032,   102,     0,
0])
example attention_mask:  tensor([1, 1, 1, 1, 1, 0, 0])
```

## 2.2 Questions

### 2.2.1 Question 1: Add a classifier to BERT [1pts]

Here, we will add a simple classifier to the BertModel provided by the Transformers library.

Your tasks are:

1. In `__init__`, add a linear classifier that will map BERTs `[CLS]` token representation to the unnormalized output probabilities for each class (`logits`).
2. In `forward`, pass BERTs `[CLS]` token representation to this new classifier to produce the `logits`.

In total, you won't have to write more than three new lines of code. See the comments in the code for help!

[ ]:
```
from transformers import BertModel
import torch.nn as nn

class BertForSentenceClassification(BertModel):
    def __init__(self, config):
        super().__init__(config)

        ##### START YOUR CODE HERE #####
        # Add a linear classifier that map BERTs [CLS] token representation to␣
↪the unnormalized
```

```
        # output probabilities for each class (logits).
        # Notes:
        #  * See the documentation for torch.nn.Linear
        #  * You do not need to add a softmax, as this is included in the loss␣
↪function
        #  * The size of BERTs token representation can be accessed at config.
↪hidden_size
        #  * The number of output classes can be accessed at config.num_labels
        self.classifier = nn.Linear(config.hidden_size, config.num_labels)
        ##### END YOUR CODE HERE #####
        self.loss = torch.nn.CrossEntropyLoss()

    def forward(self, labels=None, **kwargs):
        outputs = super().forward(**kwargs)
        ##### START YOUR CODE HERE #####
        # Pass BERTs [CLS] token representation to this new classifier to␣
↪produce the logits.
        # Notes:
        #  * The [CLS] token representation can be accessed at outputs.
↪pooler_output
        cls_token_repr = outputs.pooler_output
        logits = self.classifier(cls_token_repr)
        ##### END YOUR CODE HERE #####
        if labels is not None:
            outputs = (logits, self.loss(logits, labels))
        else:
            outputs = (logits,)
        return outputs
```

### 2.2.2 Question 2: Fine-tune BERT [0pts]

In this section, we will instantiate our pretrained BERT model + the new classifier, and train both on our verbal arithmetic dataset for a few epochs.

As mentioned in the assignment handout, we will use MathBERT, which uses the same architecture as BERT, but has been pretrained on text from pre-kindergarten, high-school, and college graduate level mathematical content.

> Although the code is written for you, please read it first to understand what it is doing. Additionally, running this code and making sure the model can be fine-tuned helps you check your implementation from Question 1. **Note**: This may print a warning: *"Some weights of the model checkpoint at…"* which you can ignore.

[ ]:
```
mathbert = BertForSentenceClassification.from_pretrained(
    "tbs17/MathBERT",  # the name of the pretrained model
    num_labels=3,      # the number of classes in our downstream task
)
```

```
Some weights of the model checkpoint at tbs17/MathBERT were not used when
initializing BertForSentenceClassification: ['cls.seq_relationship.weight',
'cls.predictions.decoder.weight', 'cls.predictions.transform.LayerNorm.weight',
'cls.predictions.transform.LayerNorm.bias',
'cls.predictions.transform.dense.weight', 'cls.predictions.bias',
'cls.predictions.transform.dense.bias', 'cls.seq_relationship.bias',
'cls.predictions.decoder.bias']
- This IS expected if you are initializing BertForSentenceClassification from
the checkpoint of a model trained on another task or with another architecture
(e.g. initializing a BertForSequenceClassification model from a
BertForPreTraining model).
- This IS NOT expected if you are initializing BertForSentenceClassification
from the checkpoint of a model that you expect to be exactly identical
(initializing a BertForSequenceClassification model from a
BertForSequenceClassification model).
Some weights of BertForSentenceClassification were not initialized from the
model checkpoint at tbs17/MathBERT and are newly initialized:
['bert.classifier.weight', 'bert.classifier.bias']
You should probably TRAIN this model on a down-stream task to be able to use it
for predictions and inference.
```

The following cell prints information about the models parameters

[ ]:
```python
# Model parameters visualization
params = list(mathbert.named_parameters())

print('The BERT model has {:} different named parameters.\n'.
 ↪format(len(params)))

print('==== Embedding Layer ====\n')

for p in params[0:5]:
    print("{:<55} {:>12}".format(p[0], str(tuple(p[1].size()))))

print('\n==== First Transformer Layer ====\n')

for p in params[5:21]:
    print("{:<55} {:>12}".format(p[0], str(tuple(p[1].size()))))

print('\n==== Output Layer ====\n')

for p in params[-4:]:
    print("{:<55} {:>12}".format(p[0], str(tuple(p[1].size()))))
```

```
The BERT model has 201 different named parameters.

==== Embedding Layer ====
```

```
embeddings.word_embeddings.weight                       (30522, 768)
embeddings.position_embeddings.weight                     (512, 768)
embeddings.token_type_embeddings.weight                     (2, 768)
embeddings.LayerNorm.weight                                   (768,)
embeddings.LayerNorm.bias                                     (768,)

==== First Transformer Layer ====

encoder.layer.0.attention.self.query.weight               (768, 768)
encoder.layer.0.attention.self.query.bias                     (768,)
encoder.layer.0.attention.self.key.weight                 (768, 768)
encoder.layer.0.attention.self.key.bias                       (768,)
encoder.layer.0.attention.self.value.weight               (768, 768)
encoder.layer.0.attention.self.value.bias                     (768,)
encoder.layer.0.attention.output.dense.weight             (768, 768)
encoder.layer.0.attention.output.dense.bias                   (768,)
encoder.layer.0.attention.output.LayerNorm.weight             (768,)
encoder.layer.0.attention.output.LayerNorm.bias               (768,)
encoder.layer.0.intermediate.dense.weight                (3072, 768)
encoder.layer.0.intermediate.dense.bias                     (3072,)
encoder.layer.0.output.dense.weight                      (768, 3072)
encoder.layer.0.output.dense.bias                             (768,)
encoder.layer.0.output.LayerNorm.weight                       (768,)
encoder.layer.0.output.LayerNorm.bias                         (768,)

==== Output Layer ====

pooler.dense.weight                                       (768, 768)
pooler.dense.bias                                             (768,)
classifier.weight                                           (3, 768)
classifier.bias                                                 (3,)
```

The next cell defines fairly standard train and evaluation loops in PyTorch

[ ]:
```
from torch.optim import AdamW
import time
import datetime
from transformers import get_linear_schedule_with_warmup
from tqdm import tqdm

def flat_accuracy(preds, labels):
    pred_flat = np.argmax(preds, axis=1).flatten()
    labels_flat = labels.flatten()
    return np.sum(pred_flat == labels_flat) / len(labels_flat)

def format_time(elapsed):
    elapsed_rounded = int(round(elapsed, 2))
    return str(datetime.timedelta(seconds=elapsed_rounded))
```

```python
def get_optimizer_and_scheduler(model, total_steps, lr=2e-5, weight_decay=0.01):
    # Apply weight decay to all parameters beside the biases or LayerNorm␣
 ↪weights
    no_decay = ['bias', 'LayerNorm.weight']
    optimizer_grouped_parameters = [
        {
            'params': [p for n, p in model.named_parameters() if not any(nd in␣
 ↪n for nd in no_decay)],
            'weight_decay': weight_decay},
        {
            'params': [p for n, p in model.named_parameters() if any(nd in n␣
 ↪for nd in no_decay)],
            'weight_decay': 0.0
        }
    ]
    optimizer = AdamW(model.parameters(), lr=lr)
    scheduler = get_linear_schedule_with_warmup(
        optimizer,
        # Warmup learning rate for first 10% of training steps
        num_warmup_steps=int(0.10 * total_steps),
        num_training_steps=total_steps,
    )
    return optimizer, scheduler

def train_model(model, epochs, train_dataloader, validation_dataloader):
    # Use GPU, if available
    device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
    model = model.to(device)

    # Setup optimizer and LR scheduler
    total_steps = len(train_dataloader) * epochs
    optimizer, scheduler = get_optimizer_and_scheduler(
        model, total_steps, lr=5e-5, weight_decay=0.01
    )

    loss_values = []
    eval_accs = []

    start = time.time()

    for epoch in range(0, epochs):
        t0 = time.time()

        total_loss = 0
        model.train()
```

```
        with tqdm(train_dataloader, unit="batch") as train_pbar:
            for batch in train_pbar:
                train_pbar.set_description(f"Training (epoch {epoch + 1})")
                b_input_ids = batch[0].to(device)
                b_input_mask = batch[1].to(device)
                b_labels = batch[2].to(device)

                model.zero_grad()

                # Perform a forward pass (evaluate the model on this training␣
↪batch).
                # This will return the loss because we have provided the␣
↪`labels`.
                outputs = model(
                    input_ids=b_input_ids,
                    attention_mask=b_input_mask,
                    labels=b_labels
                )

                # The call to `model` always returns a tuple, so we need to␣
↪pull the
                # loss value out of the tuple.
                _, loss = outputs

                # Accumulate the training loss over all of the batches so that␣
↪we can
                # calculate the average loss at the end. `loss` is a Tensor␣
↪containing a
                # single value; the `.item()` function just returns the Python␣
↪value
                # from the tensor.
                total_loss += loss.item()

                # Perform a backward pass to calculate the gradients.
                loss.backward()

                # Clip the norm of the gradients to 1.0.
                # This is to help prevent the "exploding gradients" problem.
                torch.nn.utils.clip_grad_norm_(model.parameters(), 1.0)

                # Update parameters and take a step using the computed gradient.
                # The optimizer dictates the "update rule"--how the parameters␣
↪are
                # modified based on their gradients, the learning rate, etc.
                optimizer.step()
```

```
            # Update the learning rate.
            scheduler.step()

    # Calculate the average loss over the training data.
    avg_train_loss = total_loss / len(train_dataloader)

    # Store the loss value for plotting the learning curve.
    loss_values.append(avg_train_loss)

    print("  * Average training loss: {0:.2f}".format(avg_train_loss))
    print("  * Training epoch took: {:}".format(format_time(time.time() -␣
↪t0)))

    print("Running Validation...")

    t0 = time.time()
    model.eval()

    eval_loss, eval_accuracy = 0, 0
    nb_eval_steps, nb_eval_examples = 0, 0

    # Evaluate data for one epoch
    for batch in validation_dataloader:
        batch = tuple(t.to(device) for t in batch)
        b_input_ids, b_input_mask, b_labels = batch

        with torch.no_grad():
            # Forward pass, calculate logit predictions.
            # This will return the logits rather than the loss because we␣
↪have
            # not provided labels.
            # token_type_ids is the same as the "segment ids", which
            # differentiates sentence 1 and 2 in 2-sentence tasks.
            outputs = model(
                input_ids=b_input_ids,
                attention_mask=b_input_mask
            )

        # Get the "logits" output by the model. The "logits" are the output
        # values prior to applying an activation function like the softmax.
        logits = outputs[0]
        # Move logits and labels to CPU
        logits = logits.detach().cpu().numpy()
        label_ids = b_labels.to('cpu').numpy()
        # Calculate the accuracy for this batch of test sentences.
        tmp_eval_accuracy = flat_accuracy(logits, label_ids)
        # Accumulate the total accuracy.
```

```
            eval_accuracy += tmp_eval_accuracy
            # Track the number of batches
            nb_eval_steps += 1

        avg_eval_acc = eval_accuracy/nb_eval_steps
        print("  * Accuracy: {0:.2f}".format(avg_eval_acc))
        print("  * Validation took: {:}".format(format_time(time.time() - t0)))
        eval_accs.append(avg_eval_acc)
    print("Training complete in {}".format(format_time(time.time() - start)))
    return loss_values, eval_accs
```

Finally, run the following cell to fine-tune the model

[ ]:
```
# About 2-3 seconds per epoch using GPU
mathbert_loss_vals, mathbert_eval_accs = train_model(
    model=mathbert,
    epochs=3,
    train_dataloader=bert_train_dataloader,
    validation_dataloader=bert_validation_dataloader
)
```

```
Training (epoch 1): 100%|      | 14/14 [00:01<00:00,  9.84batch/s]

  * Average training loss: 0.64
  * Training epoch took: 0:00:01
Running Validation…
  * Accuracy: 0.85
  * Validation took: 0:00:00

Training (epoch 2): 100%|      | 14/14 [00:00<00:00, 17.20batch/s]

  * Average training loss: 0.36
  * Training epoch took: 0:00:00
Running Validation…
  * Accuracy: 0.87
  * Validation took: 0:00:00

Training (epoch 3): 100%|      | 14/14 [00:00<00:00, 17.18batch/s]

  * Average training loss: 0.27
  * Training epoch took: 0:00:00
Running Validation…
  * Accuracy: 0.89
  * Validation took: 0:00:00
Training complete in 0:00:03
```

Once the model is trained, we can plot some performance metrics

[ ]:
```
%matplotlib inline
import matplotlib.pyplot as plt
import seaborn as sns
```

```
def plot_loss_and_acc(loss_vals, eval_accs):
    sns.set(style='darkgrid')
    sns.set(font_scale=1.5)
    plt.rcParams["figure.figsize"] = (12,6)
    fig, ax1 = plt.subplots(1,1)
    ax1.plot(loss_vals, 'b-o', label = 'training loss')
    ax2 = ax1.twinx()
    ax2.plot(eval_accs, 'y-o', label = 'validation accuracy')
    ax2.set_title("Training loss and validation accuracy")
    ax2.set_xlabel("Epoch")
    ax1.set_ylabel("Loss", color='b')
    ax2.set_ylabel("Accuracy", color='y')
    ax1.tick_params(axis='y', rotation=0, labelcolor='b' )
    ax2.tick_params(axis='y', rotation=0, labelcolor='y' )
    plt.show()
```

[ ]:
```
plot_loss_and_acc(mathbert_loss_vals, mathbert_eval_accs)
```

### 2.2.3 Question 3: Freezing the pretrained weights [0.5pts]

Now, lets try training the model again, except this time we will *not* fine-tune BERTs weights (we sometimes say these weights are "frozen"). To do this, we will only compute gradients for the classifiers parameters.

> We can do this in pytorch by setting the `requires_grad` attribute to `False` for all parameters beside the classifiers.

Run the following cells to instantiate the model and train only the classifier. Then answer the follow-up questions in the assignment handout.

**Note**: This may print a warning: *"Some weights of the model checkpoint at…"* which you can ignore.

[ ]:
```
mathbert_frozen = BertForSentenceClassification.from_pretrained(
    "tbs17/MathBERT",  # the name of the pretrained model
    num_labels=3,      # the number of classes in our downstream task
)
```

```
Some weights of the model checkpoint at tbs17/MathBERT were not used when
initializing BertForSentenceClassification: ['cls.seq_relationship.weight',
'cls.predictions.decoder.weight', 'cls.predictions.transform.LayerNorm.weight',
'cls.predictions.transform.LayerNorm.bias',
'cls.predictions.transform.dense.weight', 'cls.predictions.bias',
'cls.predictions.transform.dense.bias', 'cls.seq_relationship.bias',
'cls.predictions.decoder.bias']
- This IS expected if you are initializing BertForSentenceClassification from
the checkpoint of a model trained on another task or with another architecture
(e.g. initializing a BertForSequenceClassification model from a
BertForPreTraining model).
- This IS NOT expected if you are initializing BertForSentenceClassification
from the checkpoint of a model that you expect to be exactly identical
(initializing a BertForSequenceClassification model from a
BertForSequenceClassification model).
Some weights of BertForSentenceClassification were not initialized from the
model checkpoint at tbs17/MathBERT and are newly initialized:
['bert.classifier.weight', 'bert.classifier.bias']
You should probably TRAIN this model on a down-stream task to be able to use it
for predictions and inference.
```

[ ]:
```
for name, param in mathbert_frozen.named_parameters():
        # Only compute gradients for parameters of our
        # newly added classifier. BERT will not be trained.
        if 'classifier' not in name:
                param.requires_grad = False
```

[ ]:
```
# About 1 second per epoch on GPU
mathbert_frozen_loss_vals, mathbert_frozen_eval_accs = train_model(
    model=mathbert_frozen,
    epochs=3,
    train_dataloader=bert_train_dataloader,
    validation_dataloader=bert_validation_dataloader
)
```

```
Training (epoch 1): 100%|      | 14/14 [00:00<00:00, 97.39batch/s]

  * Average training loss: 1.36
  * Training epoch took: 0:00:00
Running Validation…
```

```
  * Accuracy: 0.04
  * Validation took: 0:00:00

Training (epoch 2): 100%|      | 14/14 [00:00<00:00, 96.73batch/s]

  * Average training loss: 1.22
  * Training epoch took: 0:00:00
Running Validation…
  * Accuracy: 0.06
  * Validation took: 0:00:00

Training (epoch 3): 100%|      | 14/14 [00:00<00:00, 96.49batch/s]

  * Average training loss: 1.14
  * Training epoch took: 0:00:00
Running Validation…
  * Accuracy: 0.09
  * Validation took: 0:00:00
Training complete in 0:00:00
```

[ ]:
```
plot_loss_and_acc(mathbert_frozen_loss_vals, mathbert_frozen_eval_accs)
```

### 2.2.4 Question 4: Effect of pretraining data [0.5pts]

Now, let's try fine-tuning the model again, except this time we will use BERTweets pretrained weights. BERTweets uses the same architecture as BERT (and MathBERT), but has been pretrained on 100s of millions of *tweets*.

Run the following cells to instantiate our model with BERTweets pretrained weights and fine-tune it. Then answer the follow-up questions in the assignment handout.

> **Note**: This may print a warning: *"You are using a model of type…"* which you can ignore.

[ ]:
```
bertweet = BertForSentenceClassification.from_pretrained(
    "vinai/bertweet-base",  # the name of the pretrained model
    num_labels=3,           # the number of classes in our downstream task
)
```

```
You are using a model of type roberta to instantiate a model of type bert. This
is not supported for all configurations of models and can yield errors.
Some weights of the model checkpoint at vinai/bertweet-base were not used when
initializing BertForSentenceClassification:
['roberta.encoder.layer.5.attention.output.LayerNorm.bias',
'roberta.encoder.layer.11.attention.self.key.bias',
'roberta.encoder.layer.6.intermediate.dense.bias',
'roberta.encoder.layer.3.output.dense.weight',
'roberta.encoder.layer.3.output.LayerNorm.weight',
'roberta.encoder.layer.10.intermediate.dense.weight',
'roberta.embeddings.word_embeddings.weight', 'lm_head.layer_norm.weight',
'roberta.encoder.layer.10.attention.self.query.bias',
'roberta.encoder.layer.4.intermediate.dense.weight',
'roberta.encoder.layer.7.attention.output.dense.weight',
'roberta.encoder.layer.0.intermediate.dense.weight',
'roberta.encoder.layer.10.output.LayerNorm.bias',
'roberta.encoder.layer.5.attention.output.dense.bias',
'roberta.encoder.layer.11.attention.output.LayerNorm.weight',
'roberta.encoder.layer.1.attention.self.query.weight',
'roberta.encoder.layer.1.attention.output.dense.weight',
'roberta.encoder.layer.2.intermediate.dense.weight',
'roberta.encoder.layer.5.attention.self.query.bias',
'roberta.encoder.layer.7.attention.self.key.bias',
'roberta.encoder.layer.1.output.LayerNorm.bias',
'roberta.encoder.layer.2.attention.self.key.weight',
'roberta.encoder.layer.0.attention.output.dense.bias',
'roberta.encoder.layer.10.attention.self.value.bias',
'roberta.encoder.layer.10.attention.self.key.bias',
'roberta.encoder.layer.9.output.LayerNorm.bias',
'roberta.encoder.layer.2.attention.self.value.weight',
'roberta.encoder.layer.7.attention.self.value.bias',
'roberta.encoder.layer.5.output.LayerNorm.weight',
'roberta.encoder.layer.5.attention.self.query.weight',
'roberta.encoder.layer.3.attention.output.dense.bias',
'roberta.encoder.layer.9.output.LayerNorm.weight',
'roberta.encoder.layer.0.output.dense.weight',
'roberta.encoder.layer.8.attention.self.value.bias',
'roberta.encoder.layer.10.attention.output.dense.bias',
'roberta.encoder.layer.6.attention.self.key.bias',
'roberta.encoder.layer.10.output.LayerNorm.weight',
'roberta.encoder.layer.5.attention.self.value.weight',
'roberta.encoder.layer.8.attention.self.value.weight',
```

```
'roberta.encoder.layer.4.output.dense.bias',
'roberta.encoder.layer.9.intermediate.dense.weight',
'roberta.encoder.layer.11.output.LayerNorm.weight',
'roberta.encoder.layer.8.attention.self.key.bias',
'roberta.encoder.layer.8.attention.self.query.bias', 'lm_head.dense.weight',
'roberta.encoder.layer.6.output.LayerNorm.bias',
'roberta.encoder.layer.2.intermediate.dense.bias',
'roberta.encoder.layer.11.attention.self.value.weight',
'roberta.encoder.layer.2.attention.output.LayerNorm.bias',
'roberta.encoder.layer.9.attention.output.dense.bias',
'roberta.encoder.layer.0.attention.output.dense.weight',
'roberta.encoder.layer.9.attention.output.LayerNorm.bias',
'roberta.encoder.layer.0.attention.self.key.weight',
'roberta.encoder.layer.1.attention.self.value.bias',
'roberta.encoder.layer.4.attention.self.query.weight',
'roberta.encoder.layer.8.output.dense.weight',
'roberta.encoder.layer.4.attention.self.query.bias',
'roberta.encoder.layer.1.attention.output.LayerNorm.bias',
'roberta.encoder.layer.6.attention.self.query.bias',
'roberta.encoder.layer.8.attention.self.key.weight',
'roberta.encoder.layer.9.attention.self.value.bias',
'roberta.encoder.layer.10.attention.output.LayerNorm.bias',
'roberta.encoder.layer.8.output.LayerNorm.bias',
'roberta.encoder.layer.4.output.LayerNorm.bias',
'roberta.encoder.layer.3.attention.output.dense.weight',
'roberta.encoder.layer.9.attention.self.value.weight',
'roberta.encoder.layer.7.output.dense.bias',
'roberta.encoder.layer.8.intermediate.dense.weight',
'roberta.encoder.layer.11.output.dense.weight',
'roberta.encoder.layer.6.output.dense.bias',
'roberta.encoder.layer.3.intermediate.dense.weight',
'roberta.embeddings.token_type_embeddings.weight',
'roberta.encoder.layer.2.attention.output.dense.weight',
'roberta.encoder.layer.0.intermediate.dense.bias',
'roberta.encoder.layer.4.attention.output.LayerNorm.bias',
'roberta.encoder.layer.0.attention.self.key.bias',
'roberta.encoder.layer.11.attention.self.value.bias',
'roberta.encoder.layer.0.output.LayerNorm.weight',
'roberta.encoder.layer.0.attention.self.value.weight',
'roberta.encoder.layer.11.attention.output.LayerNorm.bias',
'roberta.encoder.layer.7.attention.output.dense.bias',
'roberta.encoder.layer.7.attention.self.query.weight',
'roberta.encoder.layer.6.attention.output.LayerNorm.weight',
'roberta.encoder.layer.4.attention.self.value.bias',
'roberta.encoder.layer.8.attention.output.LayerNorm.bias',
'roberta.encoder.layer.9.attention.output.LayerNorm.weight',
'roberta.encoder.layer.11.attention.output.dense.weight',
'roberta.encoder.layer.4.attention.output.dense.weight',
```

```
'roberta.encoder.layer.6.attention.output.LayerNorm.bias',
'roberta.encoder.layer.8.output.LayerNorm.weight',
'roberta.encoder.layer.4.intermediate.dense.bias',
'roberta.encoder.layer.5.intermediate.dense.bias',
'roberta.encoder.layer.4.attention.output.dense.bias', 'lm_head.decoder.weight',
'roberta.encoder.layer.1.output.LayerNorm.weight',
'roberta.encoder.layer.5.attention.output.LayerNorm.weight',
'roberta.encoder.layer.3.attention.self.query.bias',
'roberta.encoder.layer.3.attention.output.LayerNorm.weight',
'roberta.encoder.layer.9.intermediate.dense.bias',
'roberta.encoder.layer.1.attention.self.key.bias',
'roberta.encoder.layer.6.attention.self.key.weight',
'roberta.encoder.layer.5.output.dense.weight',
'roberta.encoder.layer.1.attention.self.value.weight',
'roberta.encoder.layer.1.attention.output.dense.bias',
'roberta.encoder.layer.2.output.dense.weight',
'roberta.encoder.layer.1.intermediate.dense.bias',
'roberta.encoder.layer.6.attention.output.dense.weight',
'roberta.encoder.layer.9.attention.self.key.weight',
'roberta.encoder.layer.8.attention.self.query.weight',
'roberta.encoder.layer.8.attention.output.dense.weight',
'lm_head.layer_norm.bias',
'roberta.encoder.layer.6.attention.self.value.weight',
'roberta.encoder.layer.6.intermediate.dense.weight',
'roberta.encoder.layer.10.attention.output.dense.weight',
'roberta.encoder.layer.11.intermediate.dense.weight',
'roberta.encoder.layer.4.attention.self.key.weight',
'roberta.encoder.layer.2.attention.output.LayerNorm.weight',
'roberta.encoder.layer.6.attention.self.value.bias',
'roberta.encoder.layer.0.attention.output.LayerNorm.weight',
'roberta.encoder.layer.4.output.LayerNorm.weight',
'roberta.encoder.layer.4.attention.self.key.bias',
'roberta.encoder.layer.7.attention.output.LayerNorm.weight',
'roberta.encoder.layer.6.attention.self.query.weight',
'roberta.encoder.layer.10.attention.self.query.weight',
'roberta.encoder.layer.2.attention.self.query.weight',
'roberta.encoder.layer.10.attention.self.key.weight',
'roberta.pooler.dense.bias', 'roberta.encoder.layer.11.intermediate.dense.bias',
'roberta.encoder.layer.9.attention.output.dense.weight',
'roberta.encoder.layer.6.output.LayerNorm.weight',
'roberta.encoder.layer.5.intermediate.dense.weight',
'roberta.encoder.layer.3.attention.self.key.bias',
'roberta.encoder.layer.1.output.dense.weight',
'roberta.encoder.layer.0.attention.output.LayerNorm.bias',
'roberta.encoder.layer.0.output.LayerNorm.bias',
'roberta.encoder.layer.2.output.LayerNorm.weight',
'roberta.encoder.layer.3.intermediate.dense.bias',
'roberta.encoder.layer.0.attention.self.query.weight',
```

```
'roberta.encoder.layer.2.attention.self.value.bias',
'roberta.encoder.layer.2.attention.self.query.bias', 'lm_head.dense.bias',
'roberta.encoder.layer.11.attention.self.key.weight',
'roberta.encoder.layer.3.attention.self.query.weight',
'roberta.encoder.layer.2.attention.self.key.bias',
'roberta.encoder.layer.5.attention.self.key.weight',
'roberta.encoder.layer.5.output.dense.bias',
'roberta.encoder.layer.5.output.LayerNorm.bias',
'roberta.encoder.layer.8.output.dense.bias',
'roberta.encoder.layer.11.attention.self.query.weight',
'roberta.encoder.layer.9.attention.self.query.bias',
'roberta.encoder.layer.5.attention.self.key.bias',
'roberta.encoder.layer.7.output.dense.weight',
'roberta.encoder.layer.4.output.dense.weight',
'roberta.encoder.layer.10.output.dense.bias',
'roberta.encoder.layer.11.attention.output.dense.bias',
'roberta.encoder.layer.1.output.dense.bias',
'roberta.encoder.layer.0.attention.self.value.bias',
'roberta.encoder.layer.3.output.dense.bias',
'roberta.encoder.layer.0.output.dense.bias',
'roberta.encoder.layer.4.attention.output.LayerNorm.weight',
'roberta.encoder.layer.7.attention.self.key.weight',
'roberta.encoder.layer.9.attention.self.query.weight',
'roberta.embeddings.position_ids',
'roberta.encoder.layer.3.attention.self.value.bias',
'roberta.encoder.layer.10.attention.output.LayerNorm.weight',
'roberta.encoder.layer.11.output.dense.bias',
'roberta.encoder.layer.10.output.dense.weight', 'lm_head.decoder.bias',
'roberta.encoder.layer.11.output.LayerNorm.bias',
'roberta.encoder.layer.7.output.LayerNorm.weight',
'roberta.encoder.layer.1.attention.output.LayerNorm.weight',
'roberta.encoder.layer.2.attention.output.dense.bias',
'roberta.encoder.layer.6.attention.output.dense.bias',
'roberta.encoder.layer.6.output.dense.weight',
'roberta.encoder.layer.7.attention.self.query.bias',
'roberta.encoder.layer.7.intermediate.dense.weight',
'roberta.encoder.layer.10.attention.self.value.weight',
'roberta.encoder.layer.10.intermediate.dense.bias',
'roberta.encoder.layer.3.attention.output.LayerNorm.bias',
'roberta.encoder.layer.0.attention.self.query.bias',
'roberta.encoder.layer.7.output.LayerNorm.bias',
'roberta.encoder.layer.11.attention.self.query.bias',
'roberta.encoder.layer.3.attention.self.key.weight',
'roberta.encoder.layer.2.output.LayerNorm.bias',
'roberta.encoder.layer.9.attention.self.key.bias',
'roberta.pooler.dense.weight',
'roberta.encoder.layer.1.attention.self.key.weight',
'roberta.encoder.layer.1.intermediate.dense.weight',
```

```
'roberta.encoder.layer.2.output.dense.bias',
'roberta.encoder.layer.9.output.dense.weight',
'roberta.encoder.layer.7.attention.output.LayerNorm.bias',
'roberta.encoder.layer.7.intermediate.dense.bias', 'lm_head.bias',
'roberta.encoder.layer.4.attention.self.value.weight',
'roberta.encoder.layer.1.attention.self.query.bias',
'roberta.encoder.layer.9.output.dense.bias',
'roberta.encoder.layer.3.attention.self.value.weight',
'roberta.encoder.layer.8.attention.output.dense.bias',
'roberta.encoder.layer.5.attention.output.dense.weight',
'roberta.embeddings.LayerNorm.weight',
'roberta.encoder.layer.3.output.LayerNorm.bias',
'roberta.encoder.layer.8.attention.output.LayerNorm.weight',
'roberta.embeddings.LayerNorm.bias',
'roberta.encoder.layer.5.attention.self.value.bias',
'roberta.embeddings.position_embeddings.weight',
'roberta.encoder.layer.7.attention.self.value.weight',
'roberta.encoder.layer.8.intermediate.dense.bias']
- This IS expected if you are initializing BertForSentenceClassification from
the checkpoint of a model trained on another task or with another architecture
(e.g. initializing a BertForSequenceClassification model from a
BertForPreTraining model).
- This IS NOT expected if you are initializing BertForSentenceClassification
from the checkpoint of a model that you expect to be exactly identical
(initializing a BertForSequenceClassification model from a
BertForSequenceClassification model).
Some weights of BertForSentenceClassification were not initialized from the
model checkpoint at vinai/bertweet-base and are newly initialized:
['encoder.layer.6.intermediate.dense.bias',
'encoder.layer.0.output.LayerNorm.bias',
'encoder.layer.5.attention.self.value.bias', 'classifier.bias',
'encoder.layer.0.attention.self.value.weight',
'encoder.layer.1.attention.self.key.weight',
'encoder.layer.5.intermediate.dense.weight',
'encoder.layer.0.intermediate.dense.bias',
'encoder.layer.0.attention.self.query.bias',
'encoder.layer.3.attention.self.key.weight',
'encoder.layer.4.output.dense.weight',
'encoder.layer.10.attention.self.value.bias',
'encoder.layer.9.intermediate.dense.weight',
'encoder.layer.11.output.dense.bias',
'encoder.layer.5.attention.output.dense.weight',
'encoder.layer.6.output.dense.weight',
'encoder.layer.9.output.LayerNorm.weight',
'encoder.layer.9.attention.self.query.weight',
'encoder.layer.5.attention.output.LayerNorm.bias',
'encoder.layer.9.attention.output.dense.bias',
'encoder.layer.6.attention.output.LayerNorm.bias',
```

```
'encoder.layer.10.attention.self.query.weight',
'encoder.layer.4.attention.self.query.bias',
'encoder.layer.2.attention.self.query.bias',
'encoder.layer.10.attention.output.dense.weight',
'encoder.layer.3.output.LayerNorm.weight', 'encoder.layer.6.output.dense.bias',
'encoder.layer.7.attention.self.query.weight',
'encoder.layer.6.output.LayerNorm.bias',
'encoder.layer.2.attention.output.LayerNorm.weight',
'encoder.layer.2.intermediate.dense.bias',
'encoder.layer.5.intermediate.dense.bias',
'encoder.layer.2.attention.output.dense.weight',
'encoder.layer.1.intermediate.dense.bias',
'encoder.layer.10.intermediate.dense.weight',
'encoder.layer.4.output.dense.bias', 'pooler.dense.weight',
'encoder.layer.9.attention.output.LayerNorm.weight',
'encoder.layer.2.attention.output.dense.bias',
'encoder.layer.11.attention.output.dense.bias',
'encoder.layer.4.output.LayerNorm.bias',
'encoder.layer.9.attention.self.key.weight',
'encoder.layer.0.attention.self.key.weight',
'encoder.layer.11.attention.self.query.weight',
'encoder.layer.0.attention.self.query.weight',
'encoder.layer.0.attention.output.dense.weight',
'encoder.layer.6.attention.output.LayerNorm.weight',
'encoder.layer.8.attention.self.value.weight',
'encoder.layer.8.attention.self.value.bias',
'encoder.layer.9.intermediate.dense.bias',
'encoder.layer.10.attention.self.key.weight', 'classifier.weight',
'encoder.layer.6.attention.self.value.weight',
'encoder.layer.10.attention.output.LayerNorm.weight',
'encoder.layer.3.attention.output.LayerNorm.weight',
'encoder.layer.11.intermediate.dense.weight',
'encoder.layer.2.attention.self.value.bias',
'encoder.layer.5.output.LayerNorm.bias',
'encoder.layer.7.attention.output.dense.weight',
'encoder.layer.4.attention.self.query.weight',
'encoder.layer.10.intermediate.dense.bias',
'encoder.layer.10.attention.self.value.weight',
'encoder.layer.3.attention.self.query.bias',
'embeddings.word_embeddings.weight',
'encoder.layer.5.attention.self.query.bias',
'encoder.layer.7.attention.self.key.bias',
'encoder.layer.8.attention.output.LayerNorm.weight',
'encoder.layer.11.attention.self.value.weight',
'encoder.layer.5.attention.self.key.bias',
'encoder.layer.9.attention.output.LayerNorm.bias',
'encoder.layer.7.output.LayerNorm.weight',
'encoder.layer.3.output.dense.weight', 'encoder.layer.10.output.dense.weight',
```

```
'encoder.layer.5.output.dense.weight',
'encoder.layer.2.attention.output.LayerNorm.bias',
'encoder.layer.9.output.dense.weight',
'encoder.layer.2.attention.self.value.weight',
'encoder.layer.9.output.dense.bias', 'encoder.layer.2.attention.self.key.bias',
'encoder.layer.6.output.LayerNorm.weight',
'encoder.layer.11.attention.output.dense.weight',
'encoder.layer.6.attention.self.key.weight',
'encoder.layer.3.attention.output.dense.bias',
'encoder.layer.2.output.LayerNorm.weight',
'encoder.layer.3.attention.output.LayerNorm.bias',
'encoder.layer.4.attention.self.value.bias',
'encoder.layer.5.output.LayerNorm.weight',
'encoder.layer.2.attention.self.query.weight',
'encoder.layer.10.attention.output.dense.bias',
'encoder.layer.2.output.dense.bias', 'embeddings.position_embeddings.weight',
'encoder.layer.7.attention.self.value.bias',
'encoder.layer.10.output.LayerNorm.bias',
'encoder.layer.4.attention.output.LayerNorm.weight',
'encoder.layer.11.attention.output.LayerNorm.weight',
'encoder.layer.1.attention.output.LayerNorm.weight',
'encoder.layer.8.attention.self.query.bias',
'encoder.layer.1.output.dense.bias',
'encoder.layer.9.attention.self.query.bias',
'embeddings.token_type_embeddings.weight',
'encoder.layer.11.output.LayerNorm.weight',
'encoder.layer.4.attention.output.LayerNorm.bias',
'encoder.layer.8.output.dense.weight', 'encoder.layer.8.output.dense.bias',
'encoder.layer.1.output.dense.weight',
'encoder.layer.2.attention.self.key.weight',
'encoder.layer.7.output.dense.bias',
'encoder.layer.10.attention.self.query.bias',
'encoder.layer.6.attention.self.query.weight',
'encoder.layer.7.intermediate.dense.weight',
'encoder.layer.7.attention.output.dense.bias',
'encoder.layer.5.attention.self.value.weight',
'encoder.layer.8.intermediate.dense.bias',
'encoder.layer.6.attention.self.query.bias',
'encoder.layer.4.attention.self.key.weight',
'encoder.layer.7.attention.output.LayerNorm.weight',
'encoder.layer.0.attention.output.dense.bias',
'encoder.layer.3.output.LayerNorm.bias',
'encoder.layer.1.output.LayerNorm.weight',
'encoder.layer.2.intermediate.dense.weight',
'encoder.layer.7.attention.self.value.weight',
'encoder.layer.8.attention.self.key.weight',
'encoder.layer.8.intermediate.dense.weight',
'encoder.layer.0.attention.self.key.bias',
```

```
'encoder.layer.10.output.LayerNorm.weight',
'encoder.layer.11.intermediate.dense.bias',
'encoder.layer.9.attention.self.value.weight',
'encoder.layer.11.attention.self.key.weight',
'encoder.layer.4.attention.output.dense.weight',
'encoder.layer.6.attention.output.dense.bias',
'encoder.layer.4.attention.self.value.weight',
'encoder.layer.1.attention.output.dense.bias',
'encoder.layer.1.attention.self.query.weight',
'encoder.layer.7.output.dense.weight', 'encoder.layer.5.output.dense.bias',
'encoder.layer.11.attention.output.LayerNorm.bias',
'encoder.layer.3.intermediate.dense.bias',
'encoder.layer.3.attention.self.value.weight',
'encoder.layer.10.attention.output.LayerNorm.bias',
'encoder.layer.1.output.LayerNorm.bias',
'encoder.layer.11.attention.self.query.bias',
'encoder.layer.1.intermediate.dense.weight', 'embeddings.LayerNorm.weight',
'encoder.layer.7.attention.output.LayerNorm.bias',
'encoder.layer.8.output.LayerNorm.bias',
'encoder.layer.1.attention.output.dense.weight',
'encoder.layer.3.attention.self.key.bias',
'encoder.layer.8.attention.output.LayerNorm.bias',
'encoder.layer.3.intermediate.dense.weight',
'encoder.layer.1.attention.self.value.weight',
'encoder.layer.5.attention.self.key.weight',
'encoder.layer.8.output.LayerNorm.weight',
'encoder.layer.6.intermediate.dense.weight',
'encoder.layer.10.attention.self.key.bias',
'encoder.layer.4.output.LayerNorm.weight',
'encoder.layer.3.attention.output.dense.weight',
'encoder.layer.0.attention.output.LayerNorm.bias',
'encoder.layer.9.attention.output.dense.weight',
'encoder.layer.9.output.LayerNorm.bias', 'encoder.layer.11.output.dense.weight',
'encoder.layer.10.output.dense.bias',
'encoder.layer.5.attention.output.dense.bias',
'encoder.layer.5.attention.self.query.weight',
'encoder.layer.9.attention.self.key.bias',
'encoder.layer.8.attention.output.dense.weight',
'encoder.layer.0.output.dense.weight', 'pooler.dense.bias',
'encoder.layer.7.intermediate.dense.bias',
'encoder.layer.11.attention.self.value.bias', 'embeddings.LayerNorm.bias',
'encoder.layer.4.attention.self.key.bias',
'encoder.layer.2.output.LayerNorm.bias',
'encoder.layer.1.attention.self.value.bias',
'encoder.layer.3.output.dense.bias',
'encoder.layer.9.attention.self.value.bias',
'encoder.layer.0.attention.output.LayerNorm.weight',
'encoder.layer.1.attention.self.query.bias',
```

```
'encoder.layer.11.output.LayerNorm.bias',
'encoder.layer.4.attention.output.dense.bias',
'encoder.layer.1.attention.self.key.bias',
'encoder.layer.4.intermediate.dense.weight',
'encoder.layer.7.attention.self.query.bias',
'encoder.layer.7.attention.self.key.weight',
'encoder.layer.6.attention.self.value.bias',
'encoder.layer.0.output.dense.bias', 'encoder.layer.7.output.LayerNorm.bias',
'encoder.layer.2.output.dense.weight',
'encoder.layer.3.attention.self.query.weight',
'encoder.layer.8.attention.output.dense.bias',
'encoder.layer.0.intermediate.dense.weight',
'encoder.layer.1.attention.output.LayerNorm.bias',
'encoder.layer.8.attention.self.query.weight',
'encoder.layer.11.attention.self.key.bias',
'encoder.layer.3.attention.self.value.bias',
'encoder.layer.0.output.LayerNorm.weight',
'encoder.layer.6.attention.self.key.bias',
'encoder.layer.6.attention.output.dense.weight',
'encoder.layer.8.attention.self.key.bias',
'encoder.layer.4.intermediate.dense.bias',
'encoder.layer.0.attention.self.value.bias',
'encoder.layer.5.attention.output.LayerNorm.weight']
You should probably TRAIN this model on a down-stream task to be able to use it
for predictions and inference.
```

BERTweets has its own tokenizer, so we have to repeat the data loading process

[ ]:
```
from transformers import AutoTokenizer

bertweet_tokenizer = AutoTokenizer.from_pretrained('vinai/bertweet-base',␣
↪do_lower_case=True)

tokenized_inputs = bertweet_tokenizer(
    inputs.tolist(),
    add_special_tokens=True,
    padding='max_length',
    max_length=MAX_LEN,
    return_tensors='pt',
)

bert_train_dataloader, bert_validation_dataloader = train_valid_split(
    input_ids=tokenized_inputs['input_ids'],
    attention_masks=tokenized_inputs['attention_mask'],
    labels=labels,
    batch_size=32
)
```

```
emoji is not installed, thus not converting emoticons or emojis into text.
Install emoji: pip3 install emoji==0.6.0
Special tokens have been added in the vocabulary, make sure the associated word
embeddings are fine-tuned or trained.

example train_input:     tensor([    0,  2401, 12309, 27495,     2,     1,
1])
example attention_mask:  tensor([1, 1, 1, 1, 1, 0, 0])
```

[ ]:
```
# About 2-3 seconds per epoch on GPU
bertweet_loss_vals, bertweet__eval_accs = train_model(
    model=bertweet,
    epochs=3,
    train_dataloader=bert_train_dataloader,
    validation_dataloader=bert_validation_dataloader
)
```

```
Training (epoch 1): 100%|      | 14/14 [00:00<00:00, 21.91batch/s]

  * Average training loss: 0.62
  * Training epoch took: 0:00:00
Running Validation…
  * Accuracy: 0.75
  * Validation took: 0:00:00

Training (epoch 2): 100%|      | 14/14 [00:00<00:00, 21.86batch/s]

  * Average training loss: 0.53
  * Training epoch took: 0:00:00
Running Validation…
  * Accuracy: 0.78
  * Validation took: 0:00:00

Training (epoch 3): 100%|      | 14/14 [00:00<00:00, 21.95batch/s]

  * Average training loss: 0.44
  * Training epoch took: 0:00:00
Running Validation…
  * Accuracy: 0.77
  * Validation took: 0:00:00
Training complete in 0:00:02
```

[ ]:
```
plot_loss_and_acc(bertweet_loss_vals, bertweet__eval_accs)
```

Training loss and validation accuracy
Loss
Accuracy
0.625
0.600
0.575
0.550
0.525
0.500
0.475
0.450
0.775
0.770
0.765
0.760
0.755
0.750
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00

# clip

March 27, 2023

## 1 Colab FAQ

For some basic overview and features offered in Colab notebooks, check out: Overview of Colaboratory Features

You need to use the colab GPU for this assignment by selecting:

**Runtime** → **Change runtime type** → **Hardware Accelerator: GPU**

## 2 Part 4: Connecting Text and Images with CLIP

Acknowledgement: This notebook is based on the code from https://colab.research.google.com/github/openai/clip/blob/master/notebooks/Interacting_with_CLIP.ipynb. Credit to OpenAI.

### 2.1 Section I: Interacting with CLIP [0 pts]

This is a self-contained notebook that shows how to download and run CLIP models, calculate the similarity between arbitrary image and text inputs, and perform zero-shot image classifications. The next cells will install the clip package and its dependencies, and check if PyTorch 1.7.1 or later is installed.

**Note**: Although these steps are done for you, please take a moment to look through them and make sure you understand their purpose! Understanding how CLIP works will help you solve Section II.

[ ]:
```
import numpy as np
import torch

print("Torch version:", torch.__version__)
torch_version = torch.__version__.split(".")
assert (int(torch_version[0]) == 1 and int(torch_version[1]) >=7) or␣
↪int(torch_version[0]) > 1, "PyTorch 1.7.1 or later is required"
```

```
/home/yuling/miniconda3/envs/cuda11.6/lib/python3.9/site-
packages/tqdm/auto.py:22: TqdmWarning: IProgress not found. Please update
jupyter and ipywidgets. See
https://ipywidgets.readthedocs.io/en/stable/user_install.html
  from .autonotebook import tqdm as notebook_tqdm
```

```
Torch version: 1.13.1
```

### 2.1.1 Loading the model

`clip.available_models()` will list the names of available CLIP models.

[ ]:
```
import clip

clip.available_models()
```

[ ]:
```
['RN50',
 'RN101',
 'RN50x4',
 'RN50x16',
 'RN50x64',
 'ViT-B/32',
 'ViT-B/16',
 'ViT-L/14',
 'ViT-L/14@336px']
```

[ ]:
```
model, preprocess = clip.load("ViT-B/32")
model.cuda().eval()
input_resolution = model.visual.input_resolution
context_length = model.context_length
vocab_size = model.vocab_size

print("Model parameters:", f"{np.sum([int(np.prod(p.shape)) for p in model.
 ↪parameters()]):,}")
print("Input resolution:", input_resolution)
print("Context length:", context_length)
print("Vocab size:", vocab_size)
```

```
Model parameters: 151,277,313
Input resolution: 224
Context length: 77
Vocab size: 49408
```

### 2.1.2 Image Preprocessing

We resize the input images and center-crop them to conform with the image resolution that the model expects. Before doing so, we will normalize the pixel intensity using the dataset mean and standard deviation.

The second return value from `clip.load()` contains a torchvision `Transform` that performs this preprocessing.

[ ]:
```
preprocess
```

[ ]:
```
Compose(
    Resize(size=224, interpolation=bicubic, max_size=None, antialias=None)
    CenterCrop(size=(224, 224))
```

```
    <function _convert_image_to_rgb at 0x7f417bae3dc0>
    ToTensor()
    Normalize(mean=(0.48145466, 0.4578275, 0.40821073), std=(0.26862954,
0.26130258, 0.27577711))
)
```

### 2.1.3 Text Preprocessing

We use a case-insensitive tokenizer, which can be invoked using `clip.tokenize()`. By default, the outputs are padded to become 77 tokens long, which is what the CLIP models expects.

[ ]:
```python
clip.tokenize("Hello World!")
```

[ ]:
```
tensor([[49406,  3306,  1002,   256, 49407,     0,     0,     0,     0,     0,
             0,     0,     0,     0,     0,     0,     0,     0,     0,     0,
             0,     0,     0,     0,     0,     0,     0,     0,     0,     0,
             0,     0,     0,     0,     0,     0,     0,     0,     0,     0,
             0,     0,     0,     0,     0,     0,     0,     0,     0,     0,
             0,     0,     0,     0,     0,     0,     0,     0,     0,     0,
             0,     0,     0,     0,     0,     0,     0,     0,     0,     0,
             0,     0,     0,     0,     0,     0,     0]], dtype=torch.int32)
```

### 2.1.4 Setting up input images and texts

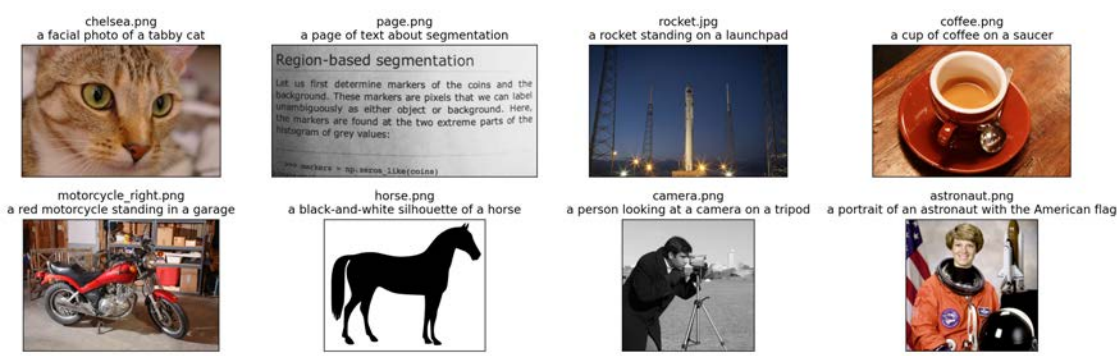
We are going to feed 8 example images and their textual descriptions to the model, and compare the similarity between the corresponding features.

The tokenizer is case-insensitive, and we can freely give any suitable textual descriptions.

[ ]:
```python
import os
import skimage
import IPython.display
import matplotlib.pyplot as plt
from PIL import Image
import numpy as np

from collections import OrderedDict
import torch

%matplotlib inline
%config InlineBackend.figure_format = 'retina'

# images in skimage to use and their textual descriptions
descriptions = {
    "page": "a page of text about segmentation",
    "chelsea": "a facial photo of a tabby cat",
    "astronaut": "a portrait of an astronaut with the American flag",
    "rocket": "a rocket standing on a launchpad",
```

```
    "motorcycle_right": "a red motorcycle standing in a garage",
    "camera": "a person looking at a camera on a tripod",
    "horse": "a black-and-white silhouette of a horse",
    "coffee": "a cup of coffee on a saucer"
}
```

[ ]:
```
original_images = []
images = []
texts = []
plt.figure(figsize=(16, 5))

for filename in [filename for filename in os.listdir(skimage.data_dir) if␣
 ↪filename.endswith(".png") or filename.endswith(".jpg")]:
    name = os.path.splitext(filename)[0]
    if name not in descriptions:
        continue

    image = Image.open(os.path.join(skimage.data_dir, filename)).convert("RGB")

    plt.subplot(2, 4, len(images) + 1)
    plt.imshow(image)
    plt.title(f"{filename}\n{descriptions[name]}")
    plt.xticks([])
    plt.yticks([])

    original_images.append(image)
    images.append(preprocess(image))
    texts.append(descriptions[name])

plt.tight_layout()
```



### 2.1.5 Building features

We normalize the images, tokenize each text input, and run the forward pass of the model to get the image and text features.

[ ]:
```
image_input = torch.tensor(np.stack(images)).cuda()
text_tokens = clip.tokenize(["This is " + desc for desc in texts]).cuda()
```

[ ]:
```
with torch.no_grad():
    image_features = model.encode_image(image_input).float()
    text_features = model.encode_text(text_tokens).float()
```

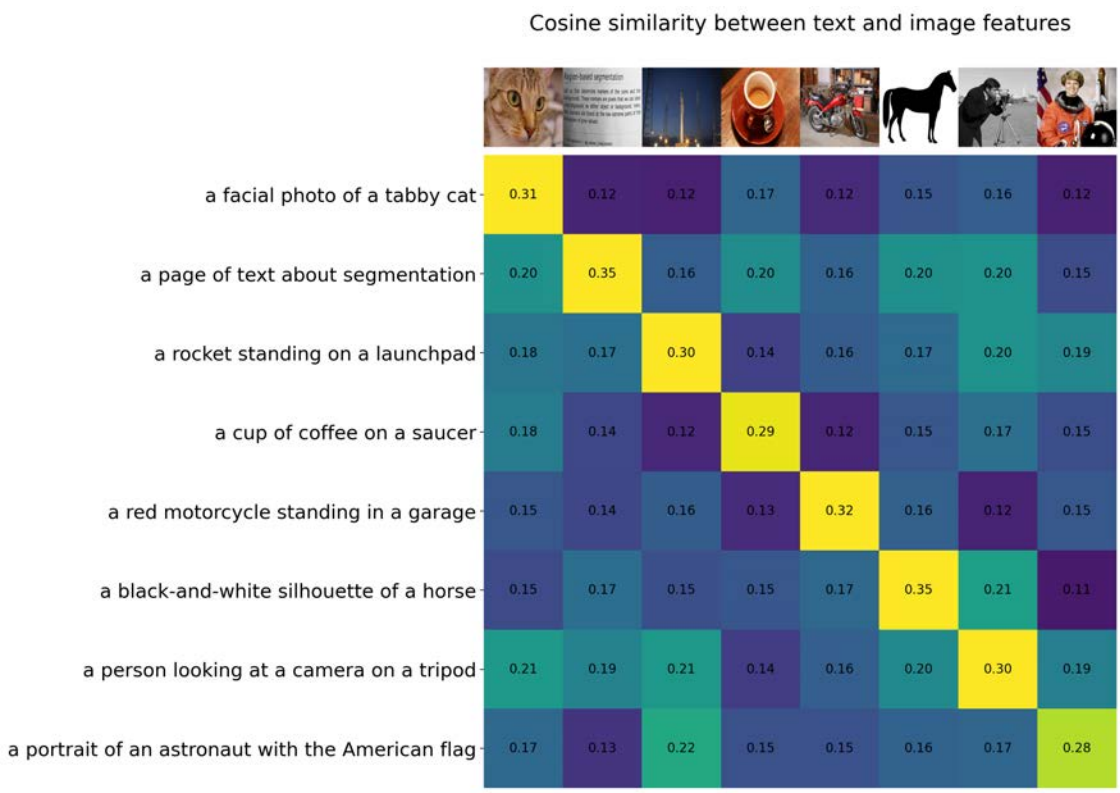
### 2.1.6 Calculating cosine similarity

We normalize the features and calculate the dot product of each pair.

[ ]:
```
image_features /= image_features.norm(dim=-1, keepdim=True)
text_features /= text_features.norm(dim=-1, keepdim=True)
similarity = text_features.cpu().numpy() @ image_features.cpu().numpy().T
```

[ ]:
```
count = len(descriptions)

plt.figure(figsize=(20, 14))
plt.imshow(similarity, vmin=0.1, vmax=0.3)
plt.yticks(range(count), texts, fontsize=18)
plt.xticks([])
for i, image in enumerate(original_images):
    plt.imshow(image, extent=(i - 0.5, i + 0.5, -1.6, -0.6), origin="lower")
for x in range(similarity.shape[1]):
    for y in range(similarity.shape[0]):
        plt.text(x, y, f"{similarity[y, x]:.2f}", ha="center", va="center",␣
 ↪size=12)

for side in ["left", "top", "right", "bottom"]:
  plt.gca().spines[side].set_visible(False)

plt.xlim([-0.5, count - 0.5])
plt.ylim([count + 0.5, -2])

plt.title("Cosine similarity between text and image features", size=20)
```

[ ]:
```
Text(0.5, 1.0, 'Cosine similarity between text and image features')
```




### 2.1.7 Zero-Shot Image Classification

You can classify images using the cosine similarity (times 100) as the logits to the softmax operation.

[ ]:
```
from torchvision.datasets import CIFAR100

cifar100 = CIFAR100(os.path.expanduser("~/.cache"), transform=preprocess,␣
↪download=True)
```

```
Files already downloaded and verified
```

[ ]:
```
text_descriptions = [f"This is a photo of a  {label}" for label in cifar100.
↪classes]
text_tokens = clip.tokenize(text_descriptions).cuda()
```

[ ]:
```
with torch.no_grad():
    text_features = model.encode_text(text_tokens).float()
    text_features /= text_features.norm(dim=-1, keepdim=True)

text_probs = (100.0 * image_features @ text_features.T).softmax(dim=-1)
```

```
top_probs, top_labels = text_probs.cpu().topk(5, dim=-1)
```

[ ]:
```
plt.figure(figsize=(16, 16))

for i, image in enumerate(original_images):
    plt.subplot(4, 4, 2 * i + 1)
    plt.imshow(image)
    plt.axis("off")

    plt.subplot(4, 4, 2 * i + 2)
    y = np.arange(top_probs.shape[-1])
    plt.grid()
    plt.barh(y, top_probs[i])
    plt.gca().invert_yaxis()
    plt.gca().set_axisbelow(True)
    plt.yticks(y, [cifar100.classes[index] for index in top_labels[i].numpy()])
    plt.xlabel("probability")

plt.subplots_adjust(wspace=0.5)
plt.show()
```

## 2.2 Section II: Let's do a Scavenger Hunt! [1 pt]

Now, lets see if we can write a caption to retrieve a particular image using CLIP. Specifically, your task is to **write a caption that best matches the following image**:

You can think of your caption as a *prompt* to the model, designed to retrieve the above image from a large collection of images. As you will see, getting this prompt correct can be tricky! To test your caption, we will use it to retrieve images from ImageNet with CLIP. You should continue to tweak your caption until the above image is returned!

First, we will download a subset of ImageNet called *"Tiny ImageNet"*. Tiny ImageNet has only 200 classes, with each class having 500 traininig images, 50 validation images and 50 test images.

In order to reduce time & memory consumption, we will consider only **one** category of images from

the train set (can you guess what this category corresponds to?) Run the following cell to load these images

[ ]:
```
import os
img_paths = []
for rootdir, subdir, filenames in os.walk("./IMagenet/tiny-imagenet-200/train/
 ↪n09256479/images"):
    for file_ in filenames:
        if not file_.endswith(".JPEG"):
            continue
        img_paths.append(os.path.join(rootdir, file_))
```

**[1pt] Write your caption below.** Remember, you should think of this caption like a *prompt*; you want to write a caption that will best match the above image under the CLIP model.

> **Hint**: Be specific! This works best when your caption is short, but descriptive and specific.

[ ]:
```
caption = "several orange and white fish in coral"
```

Below, we will display the image that produces the highest probability under the model given your caption. You should continue to tweak your caption until the target image (see above) is produced!

[ ]:
```
original_images = []
images = []

for img_path in img_paths:

    image = Image.open(img_path).convert("RGB")
    original_images.append(image)
    images.append(preprocess(image))

image_input = torch.tensor(np.stack(images)).cuda()
with torch.no_grad():
    image_features = model.encode_image(image_input).float()
image_features /= image_features.norm(dim=-1, keepdim=True)

# Delete these to avoid consuming up too much memory
del images
del image_input
torch.cuda.empty_cache()

text_tokens = clip.tokenize(caption).cuda()

with torch.no_grad():
    text_features = model.encode_text(text_tokens).float()
    text_features /= text_features.norm(dim=-1, keepdim=True)
```

```
text_probs = (100.0 * image_features @ text_features.T).softmax(dim=0).cpu().
 ↪detach().numpy()
highest_prob = np.argmax(text_probs)
plt.axis('off')
plt.imshow(original_images[highest_prob])

print(img_paths[highest_prob])
```

./IMagenet/tiny-imagenet-200/train/n09256479/images/n09256479_452.JPEG