PA2: Transformer Text Recovery

1 Introduction

The central task of this project was to train a transformer-based sequence-to-sequence model to reverse sentence preprocessing by restoring determiners, conjunctions, prepositions, and returning words to the correct unlemmatized form. In the process, the model's performance was evaluated via experiments with different types of positional embeddings, different decoding algorithms, and variations on the model's architecture. Section 2 details the base transformer architecture and attention and masking mechanisms, as well as my implementation of the datasets, training/testing process, and hyperparameter selection. The results of all three experiments are documented in Section 3, with further analysis presented in Section 4. Section 5 concludes with a discussion of the project's findings, limitations, and future changes to my approach.

2 Methods

2.1 Transformer architecture

At the highest level, the PyTorch-based sequence-to-sequence transformer architecture used in this project uses an encoder stack to process source sequences and a decoder to decode the target sequence.

The transformer's encoder consists of the following components:

- Token embedding layer: converts token IDs to vectors.
- Positional encoding: Because transformers do not inherently retain token order information like RNNs do, positional information must be manually injected. To achieve this, this architecture uses fixed sinusoidal patterns that are added to the input token embeddings.
- Stack of encoder layers
- Final layer normalization

During the forward method, the encoder takes in the input token IDs, converts them to embeddings, adds the positional encoding, creates a padding mask from the input, passes the input through the encoder layers, and returns the encoder output, padding mask, and attention weights. Each encoder layer utilizes multi-head self-attention to attend to all positions in the input.

The transformer's decoder consists of the following components, similar to the encoder:

- Token embedding layer (separate from source vocab embeddings)
- Positional encoding
- Stack of decoder layers
- Final layer normalization

During the forward method, the decoder takes in the target token IDs along with the encoder outputs and mask, creating a combined causal and padding mask for self-attention. The token IDs are then converted to embeddings and the positional encoding added. These are passed through all decoder layers with cross-attention to the encoder, returning the decoder output and attention weights. Each decoder layer utilizes masked multi-head self-attention to attend to only the previous positions in the target and cross-attention to attend to all positions in the encoder output.

The multi-head attention mechanism utilizes linear projection layers for the queries, keys, and values, splitting them into multiple heads and then computing scaled dot-product attention. If given a mask, masked positions are set to negative infinity before the heads are concatenated and the output projection layer is applied. The padding mask utilized by the encoder allows it to ignore padding tokens in the input source sequences. The causal mask utilized by the decoder is a triangular mask guaranteeing that a token at a given position can only attend to the tokens that have come before it, and not any future positions. By combining the causal and padding masks, the decoder cannot attend to padding tokens or future tokens.

After the attention mechanism in each sublayer of the encoder and decoder architectures, the output is passed to a feedforward network, consisting of two linear layers with ReLU activation between them. This network is applied to each position in the output independently. Each sublayer uses residual connections, which add sublayer transformations to the input; after each residual connection, layer normalization is applied. Dropout is applied to all attention weights and feedforward activations.

The transformer's forward method uses teacher forcing during the decoding phase, returning the logits projected to the vocabulary size. In the generate method, it instead generates sequences autoregressively without the use of teacher forcing, returning a list of generated token ID sequences.

2.2 Implementation and training

Each of the train, dev, and test datasets was implemented with a custom SeqPairDataset class, allowing the source and target data to be tokenized, bookended by <bos> and <eos> tokens, padded or trimmed to the maximum sequence length as needed, and converted to input IDs. All models used Adam optimization with cross-entropy loss and were trained using the training/dev sets before being evaluated on the test set by computing BLEU score.

2.2.1 Grid search

To select the optimal base configuration, I performed a grid search operation over different values of encoder/decoder layers, model dimension, attention heads, and feedforward dimension in order to maximize performance. The value options were as follows:

- Encoder/decoder layers: 1, 2, 4
- Model dimension: 128, 256
- Attention heads: 2, 4, 8
- Feedforward dimension: 256, 512

All iterations used 3 epochs, a learning rate of 0.001, a batch size of 64, a maximum sequence length of 50, and a dropout value of 0.1.

Unfortunately, due to time limitations, I was only able to perform grid search with training and testing on the dev set, and was unable to evaluate BLEU scores on the test set to determine the best model; therefore, I based my judgment on which models had the lowest final loss on the dev set. The absolute best-performing model by this metric had all of its hyperparameters maximized and produced a loss of 0.64142 on the dev set. The next-best model had all hyperparameters except for feedforward dimension (256) maximized, with a loss of 0.65193, and the third-best similarly had all hyperparameters except for attention heads (4) maximized, with a loss of 0.66309. In performing all of the subsequent experiments, I utilized hyperparameters from these best three models as a baseline.

2.2.2 Experiment design

The first experiment's goal was to compare the performance of the baseline sinusoidal positional encodings with that of learnable positional embeddings. One model (M1) was trained with no modifications. For the second model (M2), learnable embeddings were initialized randomly using nn.Embedding and added to the input instead of the fixed sinusoidal embeddings in both the encoder and decoder.

I used the following hyperparameters for both models:

- Epochs: 3
- Learning rate: 0.001
- Batch size: 64
- Max. sequence length: 50
- Encoder/decoder depth: 4
- Model dimension: 256
- Attention heads: 4
- Feedforward dimension: 512
- Dropout: 0.1

Results are reported in Section 3.1.

The second experiment focused on evaluating strategies for autoregressive text generation, examining both performance and time efficiency. Due to time constraints, I was unable to evaluate beam search, and instead solely analyzed greedy decoding.

I used the following hyperparameters when training and testing the model:

- Epochs: 7
- Learning rate: 0.001
- Batch size: 64
- Max. sequence length: 50
- Encoder/decoder depth: 4
- Model dimension: 256
- Attention heads: 8
- Feedforward dimension: 512
- Dropout: 0.1

Results are reported in Section 3.2.

The third experiment evaluated two different aspects of model architecture variation: number of attention heads and number of encoder/decoder layers. Each part of the experiment held all other hyperparameters fixed while comparing three models with different values for either attention heads (2, 4, 8) or encoder/decoder depth (1, 2, 4). I used the same hyperparameters as in Experiment 1 above for all models in each part of the experiment, with the exception of the single variable hyperparameter. Results are reported in Section 3.3.

3 Results

3.1 Experiment 1: Positional encoding strategies

The number of trainable parameters and the BLEU scores for the two different models are reported in Table 1. Loss curves for the train and dev sets are presented in Figure 1. Example outputs from each model were obtained every 1400 samples and are presented in Table 2 along with the corresponding reference sequence for comparison.

Model	Trainable Params.	BLEU Score
Sinusoidal	32914281	0.85110
Learnable	32939881	0.84930

Table 1: Evaluation results for each model by positional encoding type.

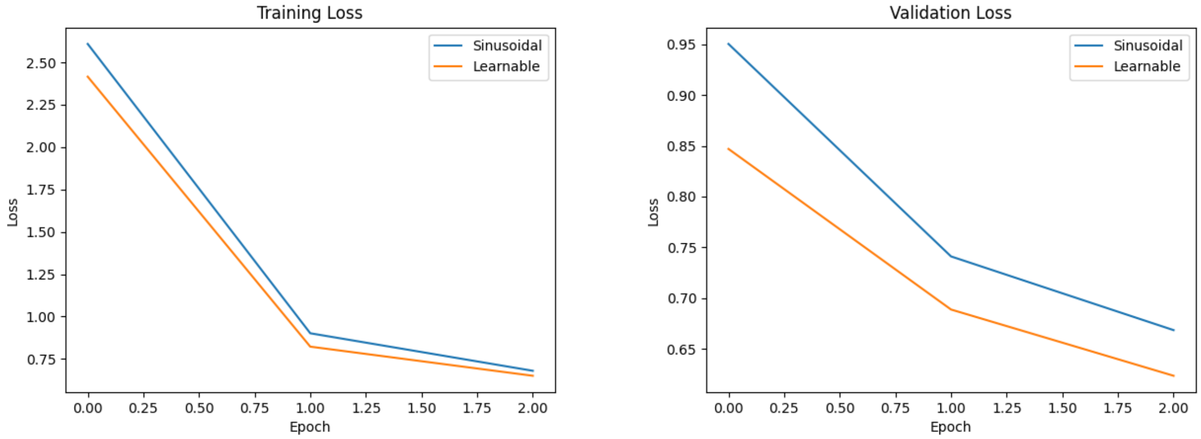

Figure 1: Training and validation loss curves for each model by positional encoding type.

Reference	be not <unk> then , nor cloud those looks , that wont to be more cheerful and serene , than when fair morning first smiles on the world ; and let us to our fresh employments rise among the groves , the fountains , and the flowers that
M1 Hyp.	is not the dishearten then , the cloud looks , wont to be more cheerful serene serene , than when fair morning first smiled on the world ; let us our fresh employments rises among groves , and flowers that open now their <unk> smell bosom , from
M2 Hyp.	these are not dishearten then , clouds look , wont is more cheerful serene , than when fair morning first smile on the world ; let us our fresh employments risen rose among the grove , the fountain that opened now their <unk> bosom smells from from from
Reference	“ well , listen to that ,” he said , “ is that a dog – anybody ’ s dog ?”
M1 Hyp.	“ well , listen ,” he said , “ is that the dog – anybody ’ s dog ?”
M2 Hyp.	“ well , listening ,” he said , “ is that dog – anybody ’ s dog ?”
Reference	either because he did not dance himself , or because the plan had been formed without his being consulted , he seemed resolved that it should not interest him , determined against its exciting any present curiosity , or affording him any future amusement .
M1 Hyp.	either either either because he did not dance himself , because the plan had been form without his being consulted , he seemed resolved that it should not interest him , determined against it excited present curiosity , affording him future amusement .
M2 Hyp.	either because because he did not dance himself , because the plan had been formed without his being consulted , he seemed resolved that it should not interest him , determined against its excite present curiosity , but afford him the future amusement .
Reference	18 : 9 and at what instant i shall speak concerning a nation , and concerning a kingdom , to build and to plant it ; 18 : 10 if it do evil in my sight , that it obey not my voice , then i will repent
M1 Hyp.	18 : 9 at what instant i shall speak concerning the nations , and concerning the kingdom , and build the plant it ; 18 : 10 and if it do evil in my sight , that it obey not my voice , then i will repent of
M2 Hyp.	18 : 9 at what instant i shall speak concerning nations , concerning the kingdom , and build plant it ; 18 : 10 if it does evil in my sight , that it obeyed not my voice , then i will repent of good , wherewith wherewith
Reference	“ i got a right to speak .
M1 Hyp.	“ i got right to speak .
M2 Hyp.	“ i got right to speak .
Reference	when they had gone a little way , they heard a cow <unk> .
M1 Hyp.	and when they had gone the little way , they heard the cow <unk> .
M2 Hyp.	when they had gone a little way , they heard the cow <unk> .
Reference	“ dear sir ,
M1 Hyp.	“ dear sir ,
M2 Hyp.	“ dear sir ,

Table 2: Example outputs for each model.

3.2 Experiment 2: Decoding algorithms

The average generation time per sample and average sequence length for the model are reported in Table 3 along with BLEU score. Example outputs are presented in Table 4.

Strategy	Avg. Time/Sample	Avg. Seq. Len	BLEU Score
Greedy	1.40006	22.47687	0.86735

Table 3: Evaluation results for greedy decoding.

Reference	be not <unk> then , nor cloud those looks , that wont to be more cheerful and serene , than when fair morning first smiles on the world ; and let us to our fresh employments rise among the groves , the fountains , and the flowers that
M1 Hyp.	this is not dishearten then , the cloud looks , wont is more cheerful and serene , than when fair morning first smile on the world ; let us our fresh employment rise among the groves , fountain , flowers that open now their <unk> smell , and
Reference	“ well , listen to that ,” he said , “ is that a dog – anybody ’ s dog ?”
M1 Hyp.	“ well , listen ,” he said , “ is that the dog – anybody ’ s a dog ?”
Reference	either because he did not dance himself , or because the plan had been formed without his being consulted , he seemed resolved that it should not interest him , determined against its exciting any present curiosity , or affording him any future amusement .
M1 Hyp.	either because he did not dance himself , because the plan had been formed without his being consulted , he seemed resolved that it should not interest him , determined against it excited the present curiosity , and afford him the future amusement .
Reference	18 : 9 and at what instant i shall speak concerning a nation , and concerning a kingdom , to build and to plant it ; 18 : 10 if it do evil in my sight , that it obey not my voice , then i will repent
M1 Hyp.	18 : 9 at what instant i shall speak concerning the nations , concerning the kingdom , and build the plant it ; 18 : 10 if it do evil in my sight , that it obey not my voice , then i will repent of good ,
Reference	“ i got a right to speak .
M1 Hyp.	“ i got right to speak .
Reference	when they had gone a little way , they heard a cow <unk> .
M1 Hyp.	when they had gone a little way , they heard the cow <unk> .

Table 4: Example outputs from greedy decoding.

3.3 Experiment 3: Model architecture variants

3.3.1 Number of attention heads

The average generation time per sample and BLEU score for each model are reported in Table 5. Loss curves for the train and dev sets are presented in Figure 2. Example outputs are presented in Table 6.

Att. Heads	Avg. Time/Sample	BLEU Score
2	1.41857	0.85103
4	1.40476	0.85249
8	1.40079	0.84921

Table 5: Evaluation results for each model by number of attention heads.

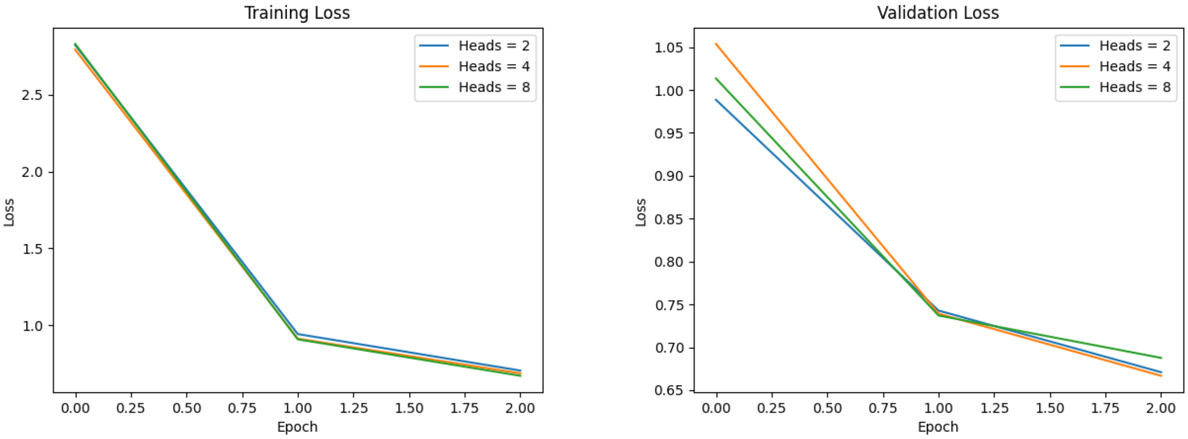

Figure 2: Training and validation loss curves for all models by number of attention heads.

Reference	be not <unk> then , nor cloud those looks , that wont to be more cheerful and serene , than when fair morning first smiles on the world ; and let us to our fresh employments rise among the groves , the fountains , and the flowers that
M1 Hyp.	this was not the dishearten then , the cloud looks , wont to be more cheerful serene , than when the fair morning first smiles on the world ; and let us our fresh employment rise among the grove , the fountain , the flower that open now
M2 Hyp.	this is not the dishearten then , the clouds looking , wont is more cheerful serene , than when the fair morning first smile on the world ; let us our fresh employments rise among the grove , fountain , and flowers that opened now their <unk> bosoms
M3 Hyp.	this is not dishearten then , clouded looks , wont is more cheerful serene , than when fair morning first smiles on the world ; let us our fresh employment to rise among the grove , fountain , flowers , flowers that open now their <unk> , reserved
Reference	“ well , listen to that ,” he said , “ is that a dog – anybody ’ s dog ?”
M1 Hyp.	“ well , listen ,” he said , “ that is that dog – anybody ’ s dog ?”
M2 Hyp.	“ well , listen ,” he said , “ is that the dogs – anybody ’ s the dogs ?”
M3 Hyp.	“ well , listen ,” he said , “ is that dog – anybody ’ s dog ?”
Reference	either because he did not dance himself , or because the plan had been formed without his being consulted , he seemed resolved that it should not interest him , determined against its exciting any present curiosity , or affording him any future amusement .
M1 Hyp.	either either either because he did not dance himself , because the plan had been formed without his being consulted , he seemed resolved that it should not interest him , determined against it excite the present curiosity , and afforded amusement .
M2 Hyp.	either either either because he did not dance himself , because the plan had been formed without his being consulted , he seemed resolved that it should not interest him , determining against it excited curiosity , and afforded him a future amusement .
M3 Hyp.	either because he did not dance himself , because the plan had been formed without his being to consult , he seemed resolved that it should not interest him , and determined against its excited curiosity , afford him future amusement .
Reference	18 : 9 and at what instant i shall speak concerning a nation , and concerning a kingdom , to build and to plant it ; 18 : 10 if it do evil in my sight , that it obey not my voice , then i will repent
M1 Hyp.	18 : 9 at what instant i shall speak concerning the nations , concerning the kingdom , and build the plant it ; 18 : 10 and if it do evil in my sight , that it obey not my voice , then i will repent of good
M2 Hyp.	18 : 9 at what instant i shall speak concerning nation , concerning the kingdom , and build plant it ; 18 : 10 if it did evil in my sight , that it obey not my voice , then i will repent of good , wherewith i
M3 Hyp.	18 : 9 at what instant i shall speak concerning the nations , concerning the kingdom , and build plant it ; 18 : 10 if it do evil in my sight , that it obey not my voice , then i will repent of good , wherewith
Reference	“ i got a right to speak .
M1 Hyp.	“ i got right to speak .
M2 Hyp.	“ i got right to speak .
M3 Hyp.	“ i got right to speak .
Reference	when they had gone a little way , they heard a cow <unk> .
M1 Hyp.	and when they had gone a little way , they heard the cow <unk> .
M2 Hyp.	when they had gone a little way , they heard the cow <unk> .
M3 Hyp.	when they had gone a little way , they heard a cow <unk> .

Table 6: Example outputs for each model.

3.3.2 Encoder/decoder depth

The average generation time per sample, number of trainable parameters, and BLEU score for each model are reported in Table 7. Loss curves for the train and dev sets are presented in Figure 3. Example outputs are presented in Table 8.

Layers	Avg. Time/Sample	Trainable Params.	BLEU Score
1	0.48600	28960617	0.82707
2	0.81566	30278505	0.83919
4	1.39813	32914281	0.85031

Table 7: Evaluation results for each model by number of encoder/decoder layers.

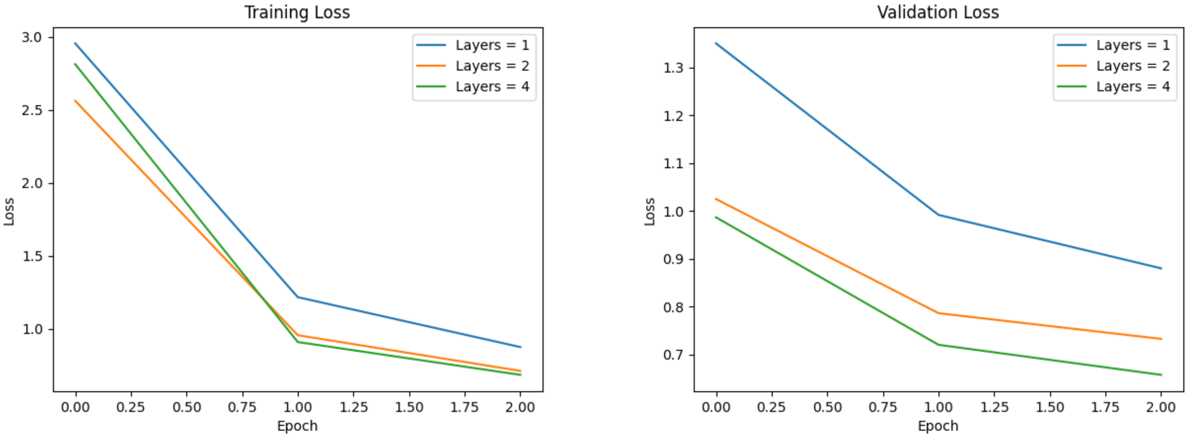

Figure 3: Training and validation loss curves for all models by number of encoder/decoder layers.

Reference	be not <unk> then , nor cloud those looks , that wont to be more cheerful and serene , than when fair morning first smiles on the world ; and let us to our fresh employments rise among the groves , the fountains , and the flowers that
M1 Hyp.	is not a dishearten then , cloud look , wont is more cheerful , than when fair morning first morning first smile on the world ; let us our fresh employments employments employments employments rise among the groves , the fountains , that opened now their <unk> and
M2 Hyp.	this is not dishearten then , the cloud look , wont is more cheerful , than when the fair morning first smiled on the world ; let us our fresh employment rise among the grove , fountain , the flower that opened now their <unk> and now their
M3 Hyp.	this was not dishearten then , clouds look , wont is more cheerful serene , than when the fair mornings first smiled on the world ; let us our fresh employments rise among groves , fountain , flowers that opened now their <unk> , and smell , reserved
Reference	“ well , listen to that ,” he said , “ is that a dog – anybody ’ s dog ?”
M1 Hyp.	“ well , listen ,” he said , “ is that dog – anybody ’ s dog ?”
M2 Hyp.	“ well , listen ,” he said , “ is that dog – anybody ’ s dog ?”
M3 Hyp.	“ well , listen ,” he said , “ is that dogs – anybody ’ s dogs ?”
Reference	either because he did not dance himself , or because the plan had been formed without his being consulted , he seemed resolved that it should not interest him , determined against its exciting any present curiosity , or affording him any future amusement .
M1 Hyp.	either because he did not dance himself , because the plan had been formed without his being consulted , he seemed resolved that it should not interest him , determined against it .
M2 Hyp.	either because he did not dance himself , because the plan had been formed without his being consulted , he seemed resolved that it should not interest him , determined against it excited to be a present curiosity , and afford him future amusement .
M3 Hyp.	either either either because he did not dance himself , because the plan had been formed without his being consulted , he seemed resolved that it should not interest him , determined against it excited presents presents , affording him to afford .
Reference	18 : 9 and at what instant i shall speak concerning a nation , and concerning a kingdom , to build and to plant it ; 18 : 10 if it do evil in my sight , that it obey not my voice , then i will repent
M1 Hyp.	18 : 9 at what an instant i shall speak concerning the nations , concerning the kingdom , and built plant it ; 18 : 10 if it did evil in my sight , that it obey it not my voice , then i will repent of good
M2 Hyp.	18 : 9 at what instant i shall speak concerning the nations , concerning the kingdom , and building it ; 18 : 10 if it did evil in my sight , that it obeyed not my voice , then i will repent of good , wherewith good
M3 Hyp.	18 : 9 at what an instant i shall speak concerning the nations , and concerning the kingdom , and build plant it ; 18 : 10 if it do evil in my sight , that it obeyed not my voice , then i will repent of good
Reference	“ i got a right to speak .
M1 Hyp.	“ i got right to speak right .
M2 Hyp.	“ i got right to speak .
M3 Hyp.	“ i get the right to speak .
Reference	when they had gone a little way , they heard a cow <unk> .
M1 Hyp.	when they had gone a little way , they heard the cow <unk> .
M2 Hyp.	when they had gone a little way , they heard cows <unk> .
M3 Hyp.	when they had gone a little way , they heard the cows <unk> .

Table 8: Example outputs for each model.

4 Analysis

In Experiment 1, the model using learnable positional encoding had 25,600 more trainable parameters than the model with fixed sinusoidal encoding. Despite having lower loss scores on both the train and dev sets, the learnable model had a lower BLEU score on the test set than the sinusoidal model did. Comparing the example outputs, it appears that the learnable model had more of a tendency to generate fewer tokens, but performed a greater degree of lemmatization reversal. This worked beneficially for example 6, where it almost perfectly matched the reference, but otherwise seems to have caused more incongruence with the reference than the sinusoidal model had. I suspect that this may be due to overfitting on the part of the learnable model. Both models perfectly generated example 7, as did all other models in other experiments, so it does not appear in later tables.

In Experiment 2, the greedy decoding model performed slightly better on the test set than the models models in Experiment 1 above, likely due to training until convergence. While not all the example predictions are exceptionally good, the longer and more complex example 3 stands out as a very close prediction relative to other models' attempts. The average sequence length was slightly under half of the maximum sequence length (50).

Experiment 3 provided the most insight into model functionality. Loss trajectory did not vary significantly between different numbers of attention heads, but contrary to my expectations, more attention heads actually resulted in a faster testing time per sample. The 4-head model had the best BLEU score, followed by 2 heads and 8 heads. Looking at the example outputs, the 8-head model performed the most extreme un-lemmatization, usually to its detriment. As the number of heads increases, it seems like the models tend to generate fewer tokens, much like the difference shown in Experiment 1 between sinusoidal and learnable positional encoding.

Increasing the number of layers had a straightforward effect on testing time and scores, with more layers taking longer but scoring higher, as well as having lower loss values. As the number of layers doubled, the average time per sample also approximately doubled. The exact effects on the example outputs are not fully clear, beyond the fact that more layers generally produced more complex predictions by token count and lemmatization reversal (which was not always beneficial, as seen in example 3). Interestingly, these models were the only ones to produce different predictions for example 5, with the 4-layer model even producing a grammatical sentence (though not a match to the reference).

Based on the experimental results, I believe the best architectural configuration has a model dimension of 256, a feedforward dimension of 512, 4 attention heads, and 8 encoder/decoder layers. The greater the model's capacity for learning at certain stages (e.g. positional encodings, attention heads), the more likely it is to produce fewer tokens and perform more lemmatization reversal. There appears to be a sweet spot for these configurations, past which point more complex models may overfit, resulting in too few tokens and too much change in word forms when generating predictions. Increasing the number of layers provides a boost in performance, but also increases the testing time much more than any other hyperparameter evaluated and substantially increases the number of trainable parameters.

5 Conclusion

This project was a valuable experience for learning the challenges of working with more advanced ML models, particularly the increased resource demands and level of abstraction. My method for computing BLEU scores was not implemented correctly the first time, which I did not realize until after I had performed a full grid search, so I had to re-run a more limited grid search (the one described in this paper) after correcting it. A more thorough exploration of the hyperparameter space would be a crucial first step in future work on this topic. Future improvements could also involve exploring beam search strategies, as well as modifying other parts of the architecture such as the attention mechanism.