# Fine-Tuning the Muse: Enhancing Quote Generation through T5 and GPT-2 Optimization

Landon Morin | $morinlandon@berkeley.edu$
Vineeta Kumar | $vineetakumar@berkeley.edu$

## Abstract

In this paper, we present a study on fine-tuning a pre-trained T5 and distilGPT2 language models for quote generation using a large corpus of textual data that we scraped from Goodreads and BrainyQuotes. We explore the impact of different hyperparameters and training strategies on the model's performance using hard prompts and evaluate its effectiveness using three commonly used metrics: perplexity, BLEU, and GLUE. Our results show that fine-tuning the model leads to a significant improvement in quote generation quality, as evidenced by higher BLEU and GLUE scores relative to baselines. Surprisingly, we introduced greated perplexity by fine-tuning our models, on which this paper will elaborate. Specifically, we achieve a best perplexity score of 36.9 with distilGPT2, GLUE score of 0.1635 with T5-small, and BLEU score of 0.1462 with T5-small outperforming our baseline models which were pre-trained T5 and fine-tuned distilGPT2. Our study provides insights into the effectiveness of fine-tuning for quote generation and demonstrates the importance of careful model selection, corpus size, hyperparameter-tuning, and evaluation metrics criteria.

## 1 Introduction

People have sought inspiration from muses of all generations. From Sun Tzu to Barack Obama, people all over the world have listened to and read the spoken and written word to gain inspiration, seek solace, and entertain. How can we keep generating such meaningful quotes to continue to inspire humanity? Text generation has been a challenging research area, where the quality of generated text is often limited due to the lack of large-scale and high-quality annotated corpus data for NLG model training. In addition, due to the limited knowledge contained in the input text, NLG models generally suffer from inability to understand context well. These problems often limit the quality of the output text by the model. With recent advancements in state of the art language models, such a task has been made readily available through fine-tuning pre-trained large language models. One such model is T5, which is a state-of-the-art language model developed by Google AI Language. T5's unique feature is that it can be fine-tuned for specific downstream tasks by simply providing a few examples and a task-specific prompt, making it a highly flexible and versatile language model. Because of T5's flexibility and open API, it is the most powerful model that is available for parameter fine-tuning. To motivate our research, and to ensure the T5 parameters would see enough data, we scraped the internet for over 2.5 million quotes, which is larger than other quote-specific datasets that the authors could find on sites such as Kaggle at the time of writing this paper.

This paper will first describe the problem, and the operationalization of the model and all experiments. We will then analyze the results and compare to key benchmarks from our baselines and from other research in text generation. Finally, we will end with suggestions for future research, given more time and financial resources.

## 2 Problem Statement and Definition

To enable quote generation, we architect a standardized hard prompt using the 'author' and 'theme' from the quotes data, to be used across all fine-tuned models such that given a prompt of length $T <= 15$, $\mathbf{p} = \{p_1, p_2, \ldots, p_T\}$, the model learns a function $f(\mathbf{p}) = \mathbf{q}$, to generate a relevant sequence of words of length $J <= 50$, $\mathbf{q} = \{q_1, q_2, \ldots, q_J\}$ that resembles a coherent quote that is on theme and reads like the specified author. We achieve this by minimizing the cross-entropy loss of T5 and distilGPT2, as defined in our methodology section.

**Example Prompt and Quote**

*Prompt:* Write a quote about love from the perspective of Nancy Reagan
*Quote Generated:* “The greatest gift you can give someone is your love. It is not always easy, but it is always worth it. Love is a force that can transcend even the most difficult times and can bring joy to the darkest moments. It is a bond that can never be broken and a feeling that can never be forgotten.”

## 3 A Review of Previous Work

In 2023, with the release of GPT4, text generation is a hot topic not only in AI research, but in the public eye. The technology is cutting edge and use cases are surfacing daily. Numerous studies on controllable text generation using large language models have also taken place in the recent years and previous work makes use of pre-trained transformer based large language models such as GPT-2 and BERT, and the generation is guided by the pre-specified objectives ((Liu et al., 2021); (Wang et al., 2021); (Zhang et al., 2020)). In almost all of these studies the researchers have made use of the most widely used approach of fine-tuning all the parameters of the entire large pre-trained model which requires a considerable amount of time and computational resources as these pre-trained large language models consist of 100s of millions if not billions of parameters (Yang and Klein, 2021).

In 2022, Dr. Julia Ive and Damith Chamalke Senadeera published a paper titled “Controlled Text Generation using T5 based Encoder-Decoder Soft Prompt Tuning and Analysis of the Utility of Generated Text in A.I.” (Chamalke Senadeera and Ive, 2022) which inspired the construction of our evaluation framework and model operationalization. In this paper, the authors propose a new way to fine-tune Encoder-Decoder models with “soft-prompts.” Whereas most Natural Language Generation techniques involve fine-tuning entire pre-trained models, “soft-prompts” allow for the Encoder-Decoder architecture to be frozen. To compensate for the loss of trainable parameters, the authors attached trainable input embeddings to the main inputs of the encoder and the decoder. This was shown to outperform hard-prompting, and to be faster than fine-tuning an entire language model.

Additionally in 2022, Moniba Keymanesh et al. published a paper titled “What Makes Data-to-Text Generation Hard for Pretrained Language Models?”(Keymanesh et al., 2022) Here too, the authors explore the impact of carefully crafted soft-prompt tuning for text generation. While the authors of this paper (Vineeta Kumar and Landon Morin) do not implement soft prompting, we were inspired by the evaluation techniques and by the experimental framework of this paper and try to implement and overcome the challenges of hard prompting along with fine-tuning carefully chosen models and hyper parameters.

## 4 Models and Methodology

In this section we introduce the data mining process, baseline T5 and GPT-2 models, as well as incremental experiments on T5 to enhance our key model metrics: Perplexity, BLEU, and GLUE.

### 4.1 Methodology

We built two scrapers that ran for one week to pull quotations, contextual tags, and authors from Goodreads and BrainyQuotes. For Goodreads, we scraped 150 pages of quotations from 30 main tags. These tags included concepts such as “love, inspiration, and humor.“ Each of these quotes contained many alternative tags/sub tags, which we used to expand the number of examples in our dataset. BrainyQuotes contained thousands of tags, so we limited scraping to 2 pages for each tag. For both sites, we limited our scraping to 2 million rows of quotations, as we wanted to conserve time and cost for experimentation. With these harvested quotes, we were then able to create prompts for both T5 and GPT-2. The prompt was constructed as such:

**Prompt**

Write a quote about `{insert tag}`from the perspective of `{insert author}`

These standardized prompts became inputs for the tokenizer and encoder, where the quotations became labels for the decoder. For T5, we leveraged a T5-large tokenizer for all models as this provided us with a wider range of vocabulary to train on, with small incremental cost to time and memory. For distilGPT2, we utilized the standard GPT-2 Tokenizer.

### 4.2 Baselines

We begin by testing non-fine-tuned distilGPT2 and T5-large on the task of generating quotes given a tag and an author. For both T5 and GPT-2, our baseline model was the standard pre-trained weights.

| **Model** | **Parameters** | **Examples** | **Hyper Parameters** |
|---|---|---|---|
| distilGPT2 | 82 mm | 150 k | Top-p/k, Repeating ngram, Temperature |
| t5-small | 60 mm | 892k | Beam width + Repeating ngram - (2,3,4) |
| t5-base | 220 mm | 892k mm | Beam width + Repeating ngram - (2,3,4) |
| t5-large | 770 mm | 267k | Beam width + Repeating ngram - (2,3,4) |

Table 1: We tested four different model architectures, with hyperparameter tuning on the decoder part of all four models. Each t5 decoder was tested on perplexity, BLEU, and GLUE, using a grid search, for the optimal combination of beam width and repeating ngrams. distilGPT2 hyperparameters were tuned manually.

| | **t5-small** | | | **t5-large** | | | **t5-base** | | |
|---|---|---|---|---|---|---|---|---|---|
| **BW, Ngram** | **PPL** | **BLEU** | **GLUE** | **PPL** | **BLEU** | **GLUE** | **PPL** | **BLEU** | **GLUE** |
| **Base - No Finetune** | 2 | 0.0201 | 0.0446 | 84 | 0.0005 | 0.0096 | 16 | 0.0112 | 0.0298 |
| 2, 2 | 3943 | 0.1354 | 0.1468 | 277 | 0.1036 | 0.1239 | 1693 | 0.1231 | 0.1417 |
| 2, 3 | 3943 | 0.1434 | 0.1578 | 277 | 0.1169 | 0.1356 | 1693 | 0.1205 | 0.1392 |
| 2, 4 | 3943 | 0.1342 | 0.1483 | 277 | 0.1172 | 0.1338 | 1693 | 0.1236 | 0.1413 |
| 3, 2 | 3943 | 0.1399 | 0.1571 | 277 | 0.1092 | 0.1294 | 1693 | 0.1295 | 0.1464 |
| 3, 3 | 3943 | 0.1447 | 0.1579 | 277 | 0.1054 | 0.1273 | **1693** | **0.1361** | **0.1552** |
| 3, 4 | **3943** | **0.1462** | **0.1635** | 277 | 0.1158 | 0.1357 | 1693 | 0.1194 | 0.1392 |
| 4, 2 | 3943 | 0.1225 | 0.1367 | 277 | 0.1018 | 0.1231 | 1693 | 0.1296 | 0.1463 |
| 4, 3 | 3943 | 0.1486 | 0.1634 | 277 | 0.1138 | 0.1334 | 1693 | 0.1301 | 0.1489 |
| 4, 4 | 3943 | 0.1292 | 0.1464 | **277** | **0.1198** | **0.1400** | 1693 | 0.1303 | 0.1480 |

Table 2: Results of grid search on beam width and number of ngram repeats. Best results for each model architecture are highlighted in bold. These metrics are an average from 400 quotes from the test set.

| **distilGPT2** | | | | | | |
|---|---|---|---|---|---|---|
| **Temp** | **No Repeat ngram** | **Top p** | **Top k** | **PPL** | **BLEU** | **GLUE** |
| 0.5 | 2 | 0.00 | 0 | 36.885 | 0.0049 | 0.0071 |
| 0.5 | 3 | 0.75 | 10 | **36.885** | **0.0053** | **0.0076** |
| 1.0 | 2 | 0.00 | 0 | 36.885 | 0.0052 | 0.0075 |
| 1.0 | 3 | 0.75 | 10 | 36.885 | 0.0049 | 0.0069 |
| 2.0 | 2 | 0.00 | 0 | 36.885 | 0.0051 | 0.0072 |
| 2.0 | 3 | 0.75 | 10 | 36.885 | 0.0049 | 0.0069 |

Table 3: Results for distilGPT2 with different decoding strategies tested through manual hyperparameter tuning. Best results for each model architecture are highlighted in bold. These metrics are an average from 400 quotes from the test set.

### 4.3 Experiments

We first tested quote generation on pretrained distilGPT2 and pretrained T5-small, base, and large. We employed a grid search technique to tune the hyperparameters of the T5 decoder, which involved testing a beam-width of 2,3, and 4 and a maximum ngram repeat of 2, 3, and 4. For distilGPT2 we leveraged manual hyperparameter tuning by using different decoding strategies outlined in Table 3. Each baseline was evaluated on perplexity, BLEU, and GLUE. Our evaluation metrics are defined as the following:

$$(1)\ Perplexity = 2^{-\frac{1}{N}\sum_{i=1}^{N}\log_2 p(w_i)}$$

where $N$ is the total number of words in the predicted quote, and $p(w_i)$ is the probability assigned by the language model to the $i$-th word in the predicted quote.

$$(2)\ BLEU = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right)$$

$BP$ is the brevity penalty, $w_n$ is the weight assigned to $n$-grams of length $n$, and $p_n$ is the precision of the n-grams. We also leveraged a smoothing function with BLEU as the model over-penalizes non-precise ngrams. Given that our model is tasked with generating mostly new text, a non-smoothed BLEU score would not be a tenable evaluator (Papineni et al., 2002).

$$(3)\ GLUE = \frac{1}{N}\sum_{i=1}^{N} Task_i Score$$

where $N$ is the total number of tasks in the GLUE benchmark, and $Task_i Score$ is the evaluation metric score for the $i^{th}$ task. From the paper introducing GLUE, determined this to be a fitting metric as it is trained on a variety of tasks (much like T5) such as sentiment analysis, textual entailment, and question answering (Wang et al., 2019).

For each model architecture, we retrained the parameters of the underlying neural net to learn from our dataset, utilizing a cross-entropy loss function and Adafactor optimizer with a dynamic learning rate.

The loss function can be defined as

$$L = -\frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{T}\log p(y_{i,t}|y_{i,<t}, x_i)$$

where $N$ is the number of training examples, $T$ is the length of the output sequence, $y_{i,t}$ is the true next word in the sequence for the $i$-th training example at time step $t$, $y_{i,<t}$ is the prefix of the output sequence up to time step $t-1$, $x_i$ is the input to the model, and $p(y_{i,t}|y_{i,<t}, x_i)$ is the predicted probability of the true next word at time step $t$. This loss function compares the probability distribution of the predicted next word with the probability distribution of the actual next word. By minimizing this loss, we penalize predicted words that may be hallucinated by the model.

### 4.4 T5 Experimentation

The experiments on T5 sought to answer three questions: (1) how much better will our results be with added complexity (i.e. parameters), (2) how much better will our model be with more examples, and (3) how can we fine-tune the decoder to generate the best quote? Given the limited financial and temporal resources for this project, the authors of this paper were forced to limit the number of examples that T5-large was trained on (see **Table 1**). However, this provided for a natural experiment and an interesting question. Is more data or a bigger model more important?

### 4.5 GPT-2 Experimentation

For the GPT-2 experiment, we chose to use the pretrained model **distilGPT2**, which is an optimized version of GPT-2 developed using knowledge distillation (Sanh et al., 2019). We selected this model for the sake of experimenting quickly. Unlike T5, GPT-2 is a *decoder only* model. In the absence of an encoder, we had to adjust the model slightly to let it understand the difference between the input (i.e. prompt) and the label (i.e. quote). We introduced two special delimiter tokens to separate the two kinds of information: `<prompt>` and `<quote>`. These tokens were placed before the corresponding sections, so there is more information for the model to distinguish between the two sections. We also applied extra transformation on our token embeddings to "annotate" tokens from each section differently like below and added the special tokens to the tokenizer.

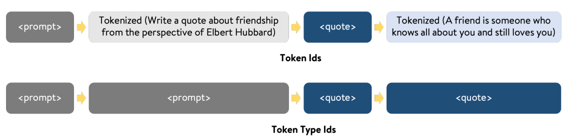


Figure 1: Token manipulation to fine-tune distilGPT2

## 5 Results

To the human eye, our baseline model generated unimpressive results, as they simply repeated a version of the prompt (see **example 1**). However these models performed surprisingly well on one metric, perplexity. Since these models mostly repeated human created sentences, perplexity was lower than the sentences that our fine-tuned models created. However, the baseline models performed poorly on BLEU and GLUE as the precision of the predicted quotes compared to the labels was poor.

**Example 1 - T5 Not Fine Tuned**

*Prompt:* Write a quote about love from the perspective of Nancy Reagan
*Quote Generated:* "from the perspective of Nancy Reagan. Write a love quote from the point of view of the late Nancy Reagan and write a quote from Nancy Reagan's perspective."

### 5.1 Fine Tuned T5 Results

All fine-tuned T5 models performed poorly against their respective baselines on perplexity, because the decoder had been trained to generate new text that was differentiable from the human-written prompt. On the other hand, the decoders that were not fine-tuned re-generated the prompt, sometimes with variations in syntax and grammar.

While the newly generated quotes for the fine-tuned T5 architectures were not perfect, they mostly made syntactic and logical sense (see appendix for examples of quotations). For more frequently occurring tags and authors, the quotes were not differentiable from human writing. For less frequently occurring tags and authors, the decoder was more likely to output hallucinated or irrelevant text.

The fine-tuned models generated better BLEU and GLUE scores against their respective baselines, even considering hallucinated texts and poorly generated quotes. As can be seen in **Figure 2** and **Figure 3**, when fine-tuning was completed, the attention weights changed to reflect more attention to key parts of the encoder ids. The untrained attention weights were almost uniformly distributed, while fine-tuning allowed the attention weights to adjust to more important tokens. This is a key factor in why the fine-tuned models generate higher BLEU and GLUE relative to our baselines. Additionally, in **Figure 3**, we can see that the decoder started paying more attention to previous words in the sequence, which helped predict future words in the sequence.

#### 5.1.1 T5-Small

T5-Small performed better than the larger T5 models on GLUE and BLEU, and demonstrated remarkable improvements over all baselines, and at times created beautiful quotes. An empirical look at the generated quotes shows that the smaller T5 model has a higher propensity to hallucinate or generate quotes that are irrelevant to the prompt. The metrics back this empirical analysis, as the perplexity of T5-small is higher and the BLEU and GLUE scores are lower than the fine-tuned base and large versions of T5. T5-small's better performance over T5-large on BLEU and GLUE can be attributed to the fact that T5-small saw a little more than 3x more examples than T5-large. Interestingly, it still did not outperform T5-large on perplexity, indicating that the smaller model's decoder is limited in predicting what the next word should be. This can be attributed to fewer model parameters.

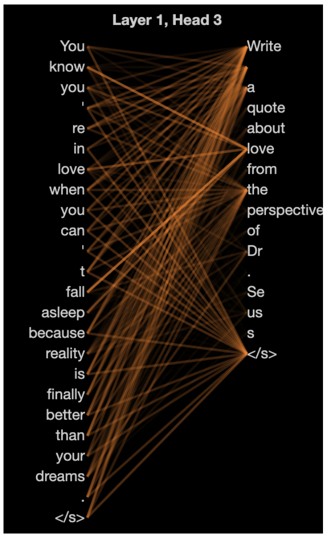


Figure 2: Cross attention for fine-tuned T5-large.


Figure 3: Cross attention for not fine-tuned T5-large.


Figure 4: Decoder attention.

More interesting is T5-small's better performance on BLEU and GLUE over T5-base, which contains almost 4x more parameters. Despite both models, having an early stopping mechanism with a patience of 3 epochs, T5-small reached 20 epochs while T5-base was cut short at 8. We hypothesize that with a longer training life-cycle, T5-base can reach parity with T5-small on these metrics.

Readability, and the ability to captivate the reader, while subjective measures, are also important considerations. T5-small produced less captivating, and less fluid quotes than the larger models as can be seen from the three examples in **Table 3** in the appendix and **example 2**.

#### 5.1.2 T5-Base

T5-Base performed markedly better on perplexity scores, but slightly worse than T5-small on BLUE and GLUE. We believe that this model performed worse than T5-small on BLEU and GLUE since it was stopped early by our early stopping mechanism. Since this model contains 4x the number of parameters, we should have raised the patience on T5-base's training to higher than T5-small as it is a slower model to train. That said, even with 12 fewer epochs of training, it almost reached parity with T5-small on BLEU and GLUE and it beat T5-small by a factor of 2.5x on perplexity.

The improvement over T5-small in perplexity supports our hypothesis that larger models are better at predicting the next word in a sequence, thereby creating more fluid text.

Again, we empirically evaluated quotes generated by T5-base, and there is a demonstrable improvement in language fluidity and complexity, as can be seen in **Table 4** in the appendix and **example 3**.

**Example 3 - Tuned T5 base**

*Prompt:* Write a quote about love from the perspective of Jane Austen
*Quote Generated:* "It is not fair to compare the love of someone with whom nothing could be more disagreeable."

#### 5.1.3 T5-Large

T5-Large far surpassed any of the other fine-tuned architectures on perplexity. This is due to its size and ability to understand and create complexity in language. However, it performed worse on BLEU and GLUE scores as it saw less of the data.

Empirically, this model generates the most interesting and aesthetically pleasing quotes. At times, it shows signs of hallucinating, but this is often restricted to lower density tags or authors. This model shines on creating quotes from the most popular tags in the dataset, such as "love," "philosophy," or "writing." Examples of T5-large quotes can be seen in **Table 5** in the appendix and **example 4**.

**Example 4 - Tuned T5 large**

*Prompt:* Write a quote about love from the perspective of Jane Austen
*Quote Generated:* "I love a man who can make me laugh, but not make me cry."

### 5.2 Fine Tuned distilGPT2 Results

distilGPT2 performed better than all fine tuned T5 models in terms of perplexity, however both the BLEU and GLUE scores were worse off across all decoding strategies. Despite the better perplexity scores, the generated quotes from distilGPT2, while syntactically correct, did not make much logical sense especially when compared to the labels and T5 generated quotes. For less common combinations of tags and authors, the decoder performed as seen in the appendix, Table 7.

## 6 Implications and Future Research

Our experimentation showed that increasing the amount of data used for training the models improved the relevance of the generated quotes. This finding suggests that the availability of more data can lead to better performance and accuracy in the generated text.

Additionally, the study showed that using bigger models improved the fluidity, aesthetic, and complexity of the generated quotes. This suggests that bigger models can capture more complex relationships and patterns in the corpus, leading to the generation of more sophisticated and engaging quotes. However, the trade-off for using bigger models is that it requires more computational power and time, which can be a significant constraint for academia. Future work could leverage more powerful computer infrastructure.

In addition to the impact of data and model size, the study also revealed the importance of considering the cost implications of training these models. Training T5 models can be an expensive undertaking that requires substantial financial investment, especially for large-scale projects. While T5-large is demonstrably better at generating text than T5-small, T5-small can be more quickly and cheaply fine-tuned to the specific task.

Lastly, we recommend alternative methods of evaluation for this type of problem. Since we intended to train a model to generate "new" quotes, we wanted a BLEU score that was not too high but also not too low. A high BLEU score means that the generated quotes were copies of the author's quotes, but a low BLEU score means that the model did not learn to write like the author or to write about a given tag. Therefore, identifying an optimal BLEU score is not probable. For future iterations of a project such as this, we recommend hiring Mechanical Turks to incorporate more human evaluation. While the authors incorporated some human evaluation by analyzing the models themselves, Mechanical Turks would reduce bias and scale better.

Overall, the findings of this study provide valuable insights into the factors that affect the performance of T5 models in generating quotes from tags and authors. The study highlights the importance of considering the trade-offs between data size, model size, and cost when designing and implementing these models. These considerations can help organizations make informed decisions when deploying T5 models for various applications, including quote generation.

## References


Damith Chamalke Senadeera and Julia Ive. 2022. Controlled text generation using t5 based encoder-decoder soft prompt tuning and analysis of the utility of generated text in ai. *arXiv e-prints*, pages arXiv–2212.

Moniba Keymanesh, Adrian Benton, and Mark Dredze. 2022. What makes data-to-text generation hard for pretrained language models? *arXiv preprint arXiv:2205.11505*.

Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A. Smith, and Yejin Choi. 2021. On-the-fly controlled text generation with experts and anti-experts. *CoRR*, abs/2105.03023.

Kishore Papineni, Salim Roukos, Todd Ward, and Wi-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 311–318.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. *GLUE: A MULTI-TASK BENCHMARK AND ANALYSIS PLATFORM FOR NATURAL LANGUAGE UNDERSTANDING*, volume 1. NYU, DeepMind, University of Washington, New York, NY.

Yufei Wang, Ian D. Wood, Stephen Wan, Mark Dras, and Mark Johnson. 2021. Mention flags (mf): constraining transformer-based text generators. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, volume 1, pages 103–113. Association for Computational Linguistics (ACL). Version archived for private and non-commercial use with the permission of the author/s and according to publisher conditions. For further rights please contact the publisher.; 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing ; Conference date: 01-08-2021 Through 06-08-2021.

Kevin Yang and Dan Klein. 2021. FUDGE: controlled text generation with future discriminators. *CoRR*, abs/2104.05218.

Yizhe Zhang, Guoyin Wang, Chunyuan Li, Zhe Gan, Chris Brockett, and Bill Dolan. 2020. POINTER: Constrained progressive text generation via insertion-based generative pre-training. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8649–8670, Online. Association for Computational Linguistics.


## 7 Appendices

| Type | Prompt | Actual | Predicted |
|---|---|---|---|
| **Popular author, common tag** | Write a quote about love from the perspective of George R.R. Martin | I crossed a thousand leagues to come to you, and lost the best part of me along the way. Don't tell me to leave. | How do you grieve the loss of a loved one? How do you know that? **PPL - 147** |
| **Popular author, uncommon tag** | Write a quote about being entangled from the perspective of Toni Morrison | I'm not entangled in shaping my work according to other people's views of how I should have done it. | I'm not a "future" type of person. I've always been a little tense and I'll always be entangled in a silly way. **PPL - 61** |
| **Less popular author, uncommon tag** | Write a quote about being headstrong from the perspective of Michelle Williams | Listen, I've always been very headstrong | I've always been a very shy person. I'm very shy. I don't know how I could handle being headstrong **PPL - 91** |

Table 4: T5 - Small Example Quotes

| Type | Prompt | Actual | Predicted |
|---|---|---|---|
| **Popular author, common tag** | Write a quote about love from the perspective of George R.R. Martin | I crossed a thousand leagues to come to you, and lost the best part of me along the way. Don't tell me to leave. | I'm a romantic, but I've never been in love, and I never have been in a relationship with a woman. **PPL - 46** |
| **Popular author, uncommon tag** | Write a quote about being entangled from the perspective of Toni Morrison | I'm not entangled in shaping my work according to other people's views of how I should have done it. | I'm not interested in being entangled in the complexities of the world. I've always been interested in the minutia of my life. **PPL - 75** |
| **Less popular author, uncommon tag** | Write a quote about being headstrong from the perspective of Michelle Williams | Listen, I've always been very headstrong | I'm not going to stop being headstrong. **PPL - 97** |

Table 5: T5 - Large Example Quotes

| Type | Prompt | Actual | Predicted |
|---|---|---|---|
| **Popular author, common tag** | Write a quote about love from the perspective of George R.R. Martin | I crossed a thousand leagues to come to you, and lost the best part of me along the way. Don't tell me to leave. | I don't care what you say, or who you say you love, or what you do, or where you are from, or how you talk to each other. I care about you. **PPL - 85** |
| **Popular author, uncommon tag** | Write a quote about being entangled from the perspective of Toni Morrison | I'm not entangled in shaping my work according to other people's views of how I should have done it. | I love being entangled in a web of my own work. **PPL - 181** |
| **Less popular author, uncommon tag** | Write a quote about being headstrong from the perspective of Michelle Williams | Listen, I've always been very headstrong. | I don't want to be the girl that sits in the booth and says, 'I'm a girl,' and I want people to know that I'm just being headstrong. **PPL - 115** |

Table 6: T5 - Base Example Quotes

| Type | Prompt | Actual | Predicted |
|---|---|---|---|
| **Popular author, common tag** | Write a quote about love from the perspective of George R.R. Martin | I crossed a thousand leagues to come to you, and lost the best part of me along the way. Don't tell me to leave. | I have no patience for anyone who loves me. |
| **Popular author, uncommon tag** | Write a quote about being entangled from the perspective of Toni Morrison | I'm not entangled in shaping my work according to other people's views of how I should have done it. | I'm not interested in being entangled in my own identity. |
| **Less popular author, uncommon tag** | Write a quote about being headstrong from the perspective of Michelle Williams | Listen, I've always been very headstrong. | I'm not just an actress, but I'm also a part-time storyteller. |

Table 7: distilGPT2 - Example Quotes