Bidirectional Modern English-Early Modern English Using Seq2Seq LSTM, Marian MT, and T5 Transformer

Pranav Prabu
pprabu@ucsd.edu

Gwendolyn Wong
g5wong@ucsd.edu

Siddhant Kuwar
skuwar@ucsd.edu

Nicole Go
nbgo@ucsd.edu

Ivan Shen
ishen@ucsd.edu

Ryan Cheung
rycheung@ucsd.edu

Abstract

Translating between Modern English and Early Modern English, particularly the Shakespearean style, presents significant challenges due to vocabulary sparsity, syntactic variation, and stylistic differences. In this work, we construct a curated parallel corpus from the *No Fear Shakespeare* dataset and investigate the performance of three model families: a bidirectional Seq2Seq LSTM, a bidirectional Marian MT transformer, and a fine-tuned T5 transformer model. Translation quality is evaluated using standard metrics such as BLEU, perplexity, and lexical similarity, in addition to Word Error Rate (WER) and a custom stylistic classifier capable of distinguishing Shakespearean from Modern English text. **Our results show that transformer-based models outperform the LSTM baseline in fluency and style preservation, with both T5 and Marian MT demonstrating strong bidirectional translation capabilities.** This study highlights the effectiveness of modern NLP architectures for historical language translation and provides insights into how these models capture temporal linguistic and stylistic variation.

1 Introduction

It is well known that translation from Modern English to Shakespearean English is challenging. After all, the corpus of all Shakespearean words is limited and the stylistic phrasing that defines Shakespearean literature is especially unique. The vocabulary of "[s]entences in the Shake-spearean style" is composed of 8559 words where "almost 60% of them appear less than 10 times" while the source domain of Modern English contains 19962 words of which only 5211 are common in both vocabularies according to (Zhao et al., 2018). In spite of the challenges, having a Shakespearean English to Modern English translator could be incredibly helpful for those studying the origins of Modern English or the changes of English literature over time. Curious individuals could also recreationally use this translator to see how different or similar Modern English is to Shakespearean English.

An initial effort to tackle these problems resulted in the pioneering of the Copy-Enriched Architecture (CopyNMT) which directly copied common factual words, proper nouns, and rare words for the purpose of content preservation and to reduce generative strain on the model (Jhamtani et al., 2017). Another approach applied Cycle Consistency Loss to ensure the meaning of the input sentence is maintained throughout disruptive stylistic changes by computing loss according to how well the model is able to recover the input sentence from its own translated output (**?**). These approaches showed that simple neural architectures could achieve fair results for unidirectional translation, but were limited in that they could not achieve bidirectional translation.

We aim to resolve this issue by introducing the Transformer architecture for the task of bidirectional translation between Shakespearean and Modern English. We investigate the performance of Marian MT and T5 fine-tuning for bidirectional models and compare them against our baseline Seq2Seq LSTM bidirectional model. We assess performance based on each model's ability to maintain fidelity, fluency, and style during translation in addition to standardized metrics such as BLEU scores, perplexity, and lexical similarity. These models were selected to observe breadth of performance and complexity across separate points along the timeline of natural language processing as a field. By building bidirectional translators and testing them across multiple architectures, we are aiming to see not just which model performs best, but how well modern natural language processing

can actually "learn time," capturing the meaning, structure, and artistic tone of English across centuries.

2 Background

Translating between historical stages of English poses significant linguistic and computational challenges. Early Modern English—the variety characteristic of Shakespeare—differs markedly from contemporary usage in morphology (e.g., distinct verb inflections such as *doth*, *hath*), pronoun systems (e.g., *thou*, *thee*, *thy*), syntactic constructions, and lexical semantics. Many words found in Shakespeare either no longer occur in Modern English or have undergone substantial semantic drift, making direct translation non-trivial for both humans and models. As a result, systems must learn to preserve meaning while simultaneously manipulating archaic morphology and stylistic conventions.

Prior computational work on this task has focused largely on parallel corpora derived from the No Fear Shakespeare editions, where Shakespearean lines are aligned with SparkNotes' modern paraphrases. As discussed in the Introduction, early neural systems such as CopyNMT and later cycle-consistency models demonstrated that even lightweight architectures can learn style transfer in one direction. However, these models operate under severe vocabulary sparsity and typically struggle to generate fluent, stylistically coherent Early Modern English in the reverse direction. The corpus itself also introduces limitations: paraphrastic modernization often reflects interpretive choices rather than literal translation, reducing lexical and syntactic alignment that models depend on (Zhao et al., 2018b).

Beyond Shakespeare-focused work, research on historical text normalization highlights similar issues. Transformer-based models, including GPT-2, T5, and Marian MT, generally outperform recurrent architectures on long-range dependencies and fluency (Zhu, 2024). Yet several studies note persistent challenges in bidirectional style transfer, especially when the target style differs from the source not only lexically but grammatically and idiomatically. Statistical systems such as Moses serve as strong baselines for monotonic token-level transformations but are insufficient for capturing stylistic register or syntactic flexibility, both of which are essential in Shakespearean language (Zhu, 2024).

Taken together, existing literature underscores the lack of robust bidirectional systems capable of translating both Modern ↔ Early Modern English while maintaining fidelity, fluency, and style. Few studies directly compare classical seq2seq models, statistical machine translation, and modern transformer architectures under a unified dataset and evaluation framework. Motivated by these gaps, our work constructs an augmented corpus and evaluates multiple model families to better understand how contemporary NLP systems learn and reproduce temporal stylistic variation.

3 Methods

3.1 Data Description

Because no publicly available parallel dataset exists for translating between Early Modern English and contemporary English, we constructed our own corpus using *No Fear Shakespeare* provided by SparkNotes (nof). The website presents Shakespeare's original Early Modern English text alongside a modern English paraphrase, making it one of the few resources suitable for supervised training. We manually extracted the parallel text from the website on a play-by-play basis, collecting pairs of lines from the original and modern versions.

After extraction, we removed all HTML formatting, act and scene labels, and non-dialogue metadata to isolate the linguistic content relevant to translation. Because SparkNotes varies in granularity—sometimes translating a single Shakespearean line into multiple modern lines or compressing several original lines into one—we manually merged or segmented entries to preserve one-to-one semantic alignment. Entries containing interpretive additions or explanatory expansions in the modern paraphrase were filtered out to maintain semantic equivalence.

Text normalization included lowercasing, punctuation standardization, whitespace cleanup, and tokenization using a regex-based approach designed to preserve Early Modern English morphology (e.g., *'tis*, *ne'er*, *doth*, *hath*). This ensured that historically meaningful lexical

forms were not broken apart or mis-tokenized. The resulting dataset contains aligned passage- or phrase-level pairs spanning multiple plays, providing diverse syntactic structures and stylistic phenomena characteristic of Early Modern English. This manually curated corpus serves as the foundation for training and evaluating our Seq2Seq LSTM, Marian MT, and T5 models in both Early Modern → Modern and Modern → Early Modern translation tasks.

3.2 Model Description

Sequence-to-Sequence (Seq2Seq)

Our sequence-to-sequence (Seq2Seq) translation model uses an encoder–decoder setup with Bahdanau additive attention. The encoder first turns input token indices into embeddings using a learned embedding layer, and these embeddings are passed through a multi-layer bidirectional LSTM. Because the LSTM reads the sequence in both forward and backward directions, each output state is formed by concatenating the two directions. After the full sequence is processed, the code combines the final hidden states by averaging the forward and backward components for each layer, while the cell states come only from the forward direction. This provides the decoder with an initial state that summarizes the whole input sentence without increasing dimensionality.

To help the model focus on the most important parts of the input during translation, we use Bahdanau-style additive attention. At every decoding step, the decoder's current hidden state is expanded and concatenated with all encoder outputs. This merged representation is passed through two linear layers with a $\tanh$ activation to compute a score for each source position. After applying a softmax, these scores become attention weights that tell the model how much to "pay attention" to each token. The model then forms a context vector by taking a weighted sum of the encoder outputs, giving the decoder direct access to the most relevant parts of the input sentence.

The decoder itself is a multi-layer bidirectional LSTM. For each timestep, it embeds the previous output token (or the correct token during teacher forcing) and concatenates that embedding with the attention-based context vector. This combined input is sent through the LSTM, and the output is then concatenated again with the context vector and the token embedding. A final linear projection converts this into a probability distribution over the target vocabulary. The decoder generates the translation one token at a time, using either its own predictions or ground-truth tokens depending on the teacher forcing ratio, which we set to 0.5.

For training, we optimize the model with Adam using a learning rate of 0.0005 and compute cross-entropy loss over the predicted sequence. To keep training stable, we apply gradient clipping at every update to avoid exploding gradients. We also use early stopping with a patience of three epochs so the model does not overfit once the validation loss stops improving. Altogether, this setup combines bidirectional encoding, attention, and autoregressive decoding to translate between Early Modern and Modern English in a way that is both flexible and interpretable.

Marian MT

We also built a bidirectional transformer by fine-tuning a Marian MT sequence-to-sequence model. In this case, the model has been adapted to translate both Early Modern → Modern and Modern → Early Modern English.

The Marian MT model uses a pretrained transformer encoder–decoder architecture for machine translation.[1] We treated Early Modern → Modern and Modern → Early Modern English as a machine translation task, training the Marian MT model on paired examples of Modern and Early Modern English as described in the Data Description section.

To indicate the desired output variety, we introduced two control tokens: `<to_shakespeare>` and `<to_modern>`. The token `<to_shakespeare>` is prepended to a Modern English sentence to instruct the model to translate it into Early Modern English, while `<to_modern>` is prepended to an Early Modern English sentence to instruct the model to translate it into Modern English. We used the MarianTokenizer, which applies subword unit tokenization via SentencePiece.[2]

[1] https://huggingface.co/docs/transformers/en/model_doc/marian

[2] https://github.com/google/sentencepiece

For training, we fine-tuned the model for 100 epochs with a batch size of 16, a learning rate of 5e-5, weight decay of 0.01, no evaluation during training to keep training continuous and fast, and dynamic padding via DataCollatorForSeq2Seq. At inference time, the model accepts either Early Modern or Modern English sentences, generates output in the opposite style conditioned on the appropriate control token, and then decodes the result back into readable text.

T5

One of our original implementations was a T5-small model to perform Modern English $\leftrightarrow$ Early Modern English. The T5 is an encoder-decoder transformer that is pretrained on large datasets of text-to-text objectives, making it a strong fit for translation tasks. Since our project involves both preserving the meaning of the statement and changing it to a different style, T5 gives the perfect balance of flexibility and structure for supervised learning.

We used instruction-style prefixes to best keep the model aligned to its purpose, using prompts like "translate modern to Shakespeare" for Modern English to Shakespearean, or "translate shakespeare to modern" for the inverse. This makes the target style clear and helps the model avoid any confusion about the desired output.

We used the T5TokenizerFast, which applied SentencePiece-style subword tokenization. This allows for the creation of subword tokens for the purpose of allowing the model to work with words, phrases, and sentences that contain older or less commonly used forms of words. The way that rare tokens in Early Modern English can be handled is by breaking them down into smaller units or pieces so that they may have some correlation to other common words. The dataset is loaded from a tab-separated file and lightly normalized during preprocessing, which includes basic whitespace cleanup and removing any numbers before constructing paired examples.

For training, we fine-tuned t5-small with a batch size of 16 on GPU (reduced automatically on CPU), a learning rate of $3e-4$, warmup ratio of 0.05, weight decay of 0.01, and dynamic padding via DataCollatorForSeq2Seq. The training procedure evaluates and saves checkpoints after each epoch, retains up to three recent checkpoints, and logs progress using the Weights & Biases API.

At inference time, the model accepts Early Modern English or Modern English sentences, applies the appropriate translation, and generates the inverse output using beam search decoding.

3.3 Code Description

All of our code is described in the notebooks in our Github repository.

3.4 Evaluation Methods

Shakespeare-Modern English Classifier

To assess the stylistic efficacy of translations, we developed a classifier capable of distinguishing between Early Modern (Shakespearean) and Modern English text. This classifier serves as an auxiliary evaluation metric beyond automated measures such as BLEU or perplexity, allowing us to quantify stylistic fidelity rather than semantic accuracy alone.

We first experimented with a TF–IDF vectorization approach combined with a logistic regression classifier. In this baseline model, each sentence is represented as a sparse vector of term weights, and the logistic regression predicts the probability of belonging to either the Shakespearean or Modern class. While this method provided a useful starting point, its limited ability to capture deeper syntactic and stylistic signals motivated us to explore transformer-based alternatives.

As a result, we adopted a DistilRoBERTa-based classifier, which encodes each sentence into contextualized embeddings using a distilled RoBERTa architecture and predicts the class label through a feed-forward layer. This model more effectively captured stylistic nuances characteristic of Early Modern English.

Both models were trained on our curated parallel corpus using an 80/20 train–test split. The final classifier enables us to evaluate whether translated sentences preserve the intended stylistic register, complementing quantitative metrics such as BLEU, perplexity, and lexical similarity.

Word Error Rate (WER)

We also implemented calculations to get the WER of the output of our models on the test set. Once again, this serves as another way to assess how our models performed on the test set and it is also directly comparable across the models to compare their performance.

WER is computed as:

$$\frac{S + I + D}{N}$$

where S is the number of times the system substitutes one source word for a different word in its transcript, D is the number of times the system deletes a source word, I is the number of times the system inserts a word in the transcript where there is no corresponding source word, and N is the total number of words in the source. Better scores are closer to 0, which gives us a way of quantifying the amount of errors made by each model and on which type of translation (Early Modern English to Modern English or Modern English to Early Modern English).

4 Results

4.1 Loss

Seq2Seq LSTM: After stopping early at 16 epochs (aiming for 100), this model had a training cross-entropy loss of 4.0289 and a validation cross-entropy loss of 5.4364. In Figure 1 below, we can see the training and validation loss of this model over the 16 epochs.

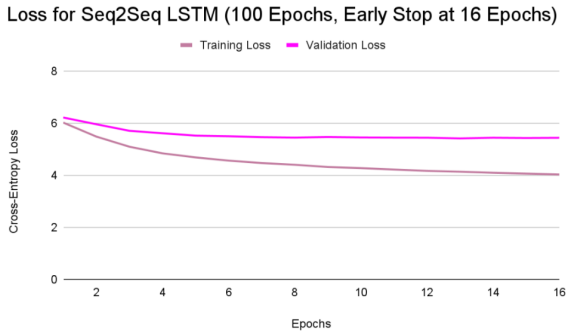

Figure 1: Training and Validation Loss for Seq2Seq LSTM (100 Epochs, Early Stop at 16 Epochs

Marian MT: After 100 epochs, this model had a training cross-entropy loss of 0.0186, which showed signs of potentially continuing given that more epochs were provided. In Figures 2 and 3 below, we can see the training and validation loss of this model over the 100 epochs.

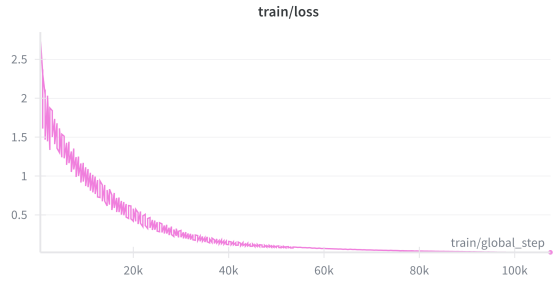

Figure 2: Training Loss for Marian MT (100 Epochs)

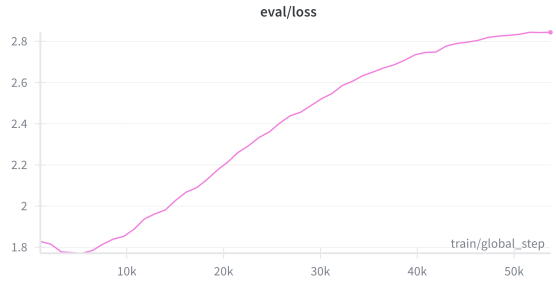

Figure 3: Evaluation Loss for Marian MT (100 Epochs)

After 50 epochs, the Marian MT model had a training cross-entropy loss of 0.075900 and a validation cross-entropy loss of 2.844259. Similar to the version with 100 epochs, both losses showed signs of continued improvement.

T5: After 100 epochs, this model had a training cross-entropy loss of 0.9194, which showed signs of potentially continuing given that more epochs were provided. As for the validation cross-entropy loss, the model reported a value of 2.1214. In Figures 4 and 5 below, we can see the training and validation loss of this model over the 100 epochs.

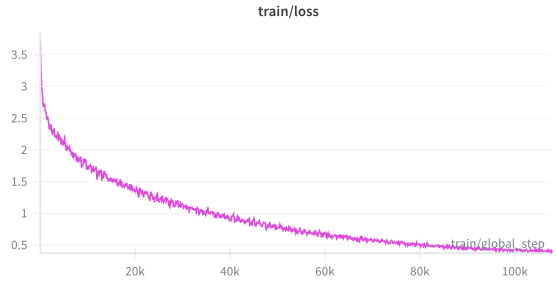

Figure 4: Training Loss for T5 (100 Epochs)

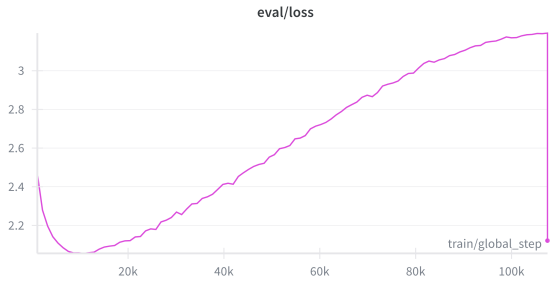

Figure 5: Evaluation Loss for T5 (100 Epochs)

After 50 epochs, the T5 model has a training cross-entropy loss of 1.3380. As for the validation cross-entropy loss, the model reported a value of 2.2118. In Figures 6 and 7 below, we can see the training and validation loss of this model over the 50 epochs.

Figure 6: Training Loss for T5 (50 Epochs)

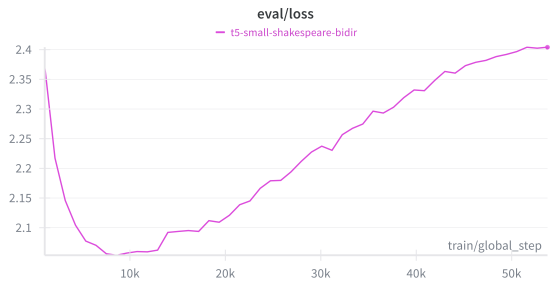

Figure 7: Evaluation Loss for T5 (50 Epochs)

4.2 Shakespeare-Modern English Classifier

Seq2Seq LSTM: With 16 epochs (attempted 100) for the Seq2Seq LSTM model, our Shakespeare-Modern English Classifier achieved a Modern English test accuracy of 0.0005 and an Early Modern English test accuracy of 0.9995.

Marian MT: With 100 epochs for the Marian MT model, our Shakespeare-Modern English Classifier achieved a Modern English test accuracy of 0.6165 and an Early Modern English test accuracy of 0.7501.
With 50 epochs for the Marian MT model, our Shakespeare-Modern English Classifier achieved a Modern Test Accuracy of 0.6435 and an Early modern English test accuracy of 0.7478.

T5: With 100 epochs for the T5 model, our Shakespeare–Modern English classifier achieved a Modern English test accuracy of 0.6277 and an Early Modern English test accuracy of 0.7154.

4.3 WER

Seq2Seq LSTM: After stopping early at 16 epochs (aiming for 100), for translating from Modern English to Early Modern English, this model had a WER of 2.2163 with 17867 substitutions, 156 deletions, and 17416 insertions. It had a WER of 2.0864 with 19151 substitutions, 154 deletions, and 17361 insertions when translating from Early Modern English to Modern English.

Marian MT: After 100 epochs, for translating from Modern English to Early Modern English, this model had a WER of 0.6762 with 9586 substitutions, 1654 deletions, and 2038 insertions. It had a WER of 0.6514 with 10064 substitutions, 2101 deletions, and 1809 insertions when translating from Early Modern English to Modern English. After 50 epochs, for translating from Modern English to Early Modern English, this model had a WER of 0.6609 with 9468 substitutions, 1652 deletions, and 1994 insertions. It had a WER of 0.6354 with 9854 substitutions, 2102 deletions, and 1677 insertions.

T5: After 100 epochs, for translating from Modern English to Early Modern English, this model had a WER of 0.6520 with 8960 substitutions, 1646 deletions, and 1965 insertions. It had a WER of 0.6226 with 8794 substitutions, 2982 deletions, and 1232 insertions when translating from Early Modern English to Modern English.

4.4 General Results Comparison

As expected, the Seq2Seq LSTM model performed worse than the Marian MT and T5 models in a holistic manner.

With regard to cross-entropy loss, we found that the Seq2Seq LSTM model saw a value of 5.4364 on the validation set, which was much

higher than that of the Marian MT model (50 epochs) of 2.8443. The best performance on the validation set was by the T5 model (100 epochs), which had a loss of 2.1214. The slope of change of training loss with more training epochs was much less in the Seq2Seq LSTM model, too, as can be seen when comparing Figure 1 to Figure 2, Figure 4, and Figure 6. Where the Marian MT mode and the T5 model saw exponential decay in terms of training loss, the Seq2Seq LSTM model's training loss was much closer to linear. However, with regard to overfitting the models, it appears that this happened the least for the Seq2Seq LSTM model, and the Marian MT model and T5 model were both saw a decent amount of overfitting when judging from Figure 3, Figure 5, and Figure 7. It thus appears that we could have used a better number of epochs to train our Marian MT model and T5 model, but even as is, the loss of these two (both training and validation) were better than that of the Seq2Seq LSTM model.

For the classifier, the Seq2Seq LSTM (16 epochs) had an incredibly high accuracy performance unidirectionally (translating from Modern English to Early Modern English) with a score of matching the classifier's results at 0.9995. However, it sacrificed a drastically lower accuracy performance in the other direction (translating from Early Modern English to Modern English) as it's score was 0.0005. The Marian MT results had me consistency in both directions with decent accuracy scores, that slightly improved with more epochs. After training for 50 epochs, it's accuracy against the classifier translating from Modern English to Early Modern English is 0.7501 while it's accuracy translating from Early Modern English to Modern English is 0.6165. After training for 100 epochs, it's accuracy against the classifier translating from Modern English to Early Modern English is 0.7478 while it's accuracy translating from Early Modern English to Modern English is 0.6435. More training made it'd performance in Modern test accuracy increase and performance in Early Modern test accuracy decrease, but it caused a closer consistency between the two bidirectionally. Finally, T5 achieved slightly similar results than both of the previously mentioned models. On 100 epochs of training, it's Early Modern test accuracy was 0.7154 while it's Modern Test Accuracy was 0.6435. It's performance is slightly lower than Marian MT and it is also slightly less consistent in both directions as the gap between the two values is larger.

WER reflected similar results as those stated above. The Seq2Seq LSTM model had the highest WER by far when compared to the Marian MT and T5 models. There was a general trend across all three models, however, that translating from Early Modern English to Modern English had a lower WER and thus better performance than translating from Modern English to Early Modern English. Similar to for the case of cross-entropy loss, the Seq2Seq LSTM model was the worst of the three models by far, followed by a decent gap, then the performance of the Marian MT model which was closely followed by the performance of the T5 model, which was once again established as the best performing of the three models. All models had substitution errors happen the most. Interestingly, while the Seq2Seq LSTM model had very few deletions and a nearly equal amount of substitutions and insertions, the Marian MT and T5 models both had the most substitution errors, followed by a nearly equivalent amounts of deletions and insertions. This reflects the differences in structural outputs of the models.

5 Discussion/Conclusion

5.1 Limitations

Despite achieving promising results, our study has several limitations. First, the parallel corpus we constructed from *No Fear Shakespeare* is limited in size and scope, covering only the subset of Shakespeare's works adapted by SparkNotes. This restricts the diversity of linguistic structures and vocabulary available for training, potentially affecting model generalization to other Early Modern English texts.

Second, while our models capture style and meaning reasonably well, they sometimes produce outputs that are either overly literal or outdated, particularly when handling rare or archaic words not well represented in the training data. Bidirectional translation remains more challenging than unidirectional, and our Marian MT and T5 models occasionally fail to maintain both fluency and stylistic fidelity simultaneously.

Third, evaluation metrics such as ROUGE/BLEU,

perplexity, and lexical similarity may not fully capture stylistic nuance, literary tone, or historical correctness, which are inherently subjective and difficult to quantify. Similarly, automated evaluation cannot fully account for semantic preservation in cases where multiple valid Early Modern English renderings exist.

Fourth, our work does not address domain-specific or regional variations in Early Modern English, focusing instead on a generalized Shakespearean style. This limits the applicability of the models to texts outside the Shakespearean corpus or to specialized registers of Early Modern English such as legal, religious, or medical texts.

Finally, a limitation included the amount of time it took to train models and run code given our resources and GPUs provided by Colab. Having more resources could lead to faster runtimes, and more time and capabilities to test a high number of epochs more and more vigorously. It also limits factors like how much data that can be handled and how powerful the chosen models could be.

5.2 Expansions

A possible expansion is to have specialized data to fine-tune the models in certain aspects. For instance, the models could be tuned to a certain domain of Early Modern English. This includes solely focusing on only plays, long-term texts, etc. This can also extend to specific dialects of Early Modern English from different regions. For registers, there could be a focus on formal texts like laws or informal texts like conversations. There could also be specific knowledge areas of different professions at he time such as medicine, religion, etc. There is also overlap between these categories, so there can be interesting possible combinations. Experimenting with different types of texts can lead to exploration of which models do well with which specialities.

In terms of methods, with the necessary resources, there could be further expansion done with the hyperparameters. Such hyperparameters include more epochs and adjusting learning rate, batch size, and weight decay. This would explore how each fine tunes the model and what combination could help the model perform better.

There can also be additional features added to create a client-side translator tool. A helpful feature could include a text-to-speech for pronunciation. Calibration is also helpful for the translator to admit where it is not confident. Connotation clarification would be helpful too, so that the user could learn what Shakespeare words should be used in what contexts.

To add on to this, there could be translators made for other eras of English or other languages with the same methods. There can be evaluations made on how the models react to the different dialects and languages, and if they can handle certain languages more due to factors like syntax, sentence structures, grammar rules, etc. It can also be helpful as it would visualize how these modern models react to how language changed throughout time.

References

No Fear Shakespeare. https://www.sparknotes.com/nofear/shakespeare/%7D%7D. Accessed: 2025-12-07.

Harsh Jhamtani, Varun Gangal, Eduard Hovy, and Eric Nyberg. 2017. Shakespearizing modern language using copy-enriched sequence-to-sequence models. In *Proceedings of the Workshop on Stylistic Variation*, pages 10–19. Association for Computational Linguistics.

Yanpeng Zhao, Wei Bi, Deng Cai, Xiaojiang Liu, Kewei Tu, and Shuming Shi. 2018a. Language style transfer from sentences with arbitrary unknown styles. *arXiv preprint arXiv:1808.04071*.

Zhiting Zhao, Junjie Kim, Tianyi Zhang, Alexander Rush, and Eric Zhou. 2018b. Back-translation style transfer. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 872–883. Association for Computational Linguistics.

Lin Zhu. 2024. Neural approaches to historical text normalization. Master's thesis, University of Edinburgh.

(Jhamtani et al., 2017) (Zhao et al., 2018a) (Zhao et al., 2018b) (Zhu, 2024) (nof)