Final Report

Leila Khaertdinova, BS21 DS-02

November 2023

Introduction

Text Detoxification Task is a process of transforming the text with toxic style into the text with the same meaning but with neutral style. The report aims to present potential approaches to address this task, while also providing insights about the data analysis, training process, and the final results.

Data analysis

The main dataset used for text detoxification task is the filtered ParaNMT-detox corpus including 500K sentence pairs. These pairs are typically a sample with high toxicity level and its paraphrazed version with low toxicity level.

Based on data exploration, it was found that the dataset consists of sentence pairs with high cosine similarity scores from 0.6 to 0.95. However, a significant portion of translations were found to have higher toxicity levels than their references. To address this issue, data preprocessing was performed. Additionally, the sentence pairs were filtered to ensure they had sufficiently low and high toxicity levels. During the term frequency analysis, it was discovered that there are a lot of swear words among the most frequent words (See Figure 1).

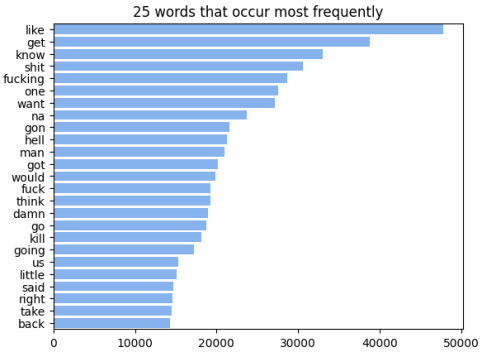


Figure 1: Term frequency analysis result

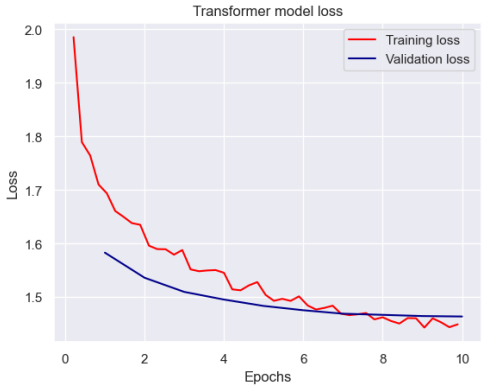


Figure 2: Pytorch transformer losses during training

Model Specification

To address this task, I have created three solutions that include different techniques and models. These approaches include:

1. **Baseline: Removing Toxic Words**

   No ML model was used. This approach focuses on just removing offensive words from the source text. Offensive words are defined using an open-source dictionary of swear words.

2. **Seq2Seq Transformer using PyTorch**

   Transformer is a Seq2Seq model introduced in [1]. This model consists of three parts: An embedding layer that converts input indices into the corresponding input embeddings. These embedding are further augmented with positional encodings to provide position information of input tokens to the model. The second part is the actual Transformer model. Finally, the output of the Transformer model is passed through linear layer that provides probabilities for each token in the target.

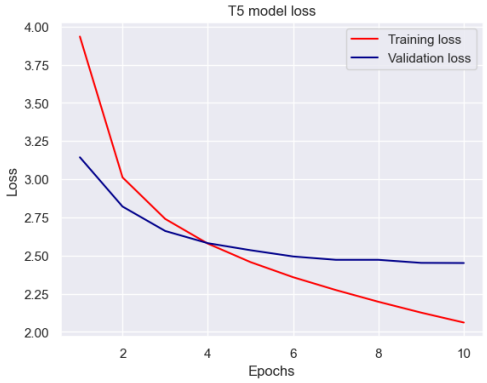
3. **Fine-tuning T5 Model**

   T5 model is Text-To-Text Transfer Transformer, which structure is just a standard sort of vanilla encoder-decoder transformer. For this approach, I used T5-base checkpoint with 220M parameters from [2].

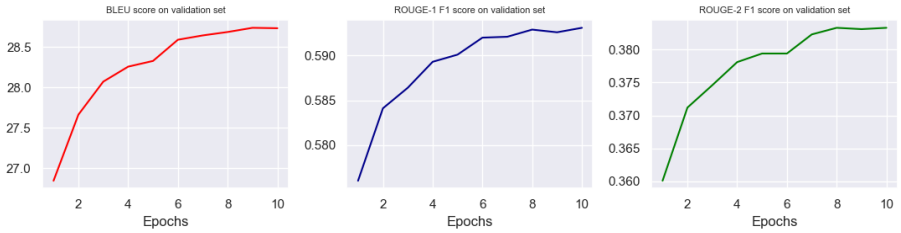
Training Process

1. **Seq2Seq Transformer Training**

   I trained transformer from scratch using PyTorch with batch size 32 during 10 epochs. The training and validation losses are shown in Figure 2.



(a) Train and validation losses



(b) Metrics on validation set

Figure 3: T5 model fine-tuning results

2. **Fine-tuning T5 Model**

   I used a pre-trained T5 model with T5-Base checkpoint and fine-tune it with batch size 16 for 10 epochs (for bigger batch size the GPU limit appeared). See Figure 3 to check the results.

Evaluation

Metrics used for a final evaluation:

- **BLEU-2** and **BLEU-4**: These metrics are useful for assessing the quality of generated text. They measure how closely the generated text matches the reference text in terms of 2-grams and 4-grams, respectively.
- **ROUGE-1** and **ROUGE-2**: These metrics are used to evaluate the quality of texts by comparing them to reference texts. They measure the overlap of unigrams and bigrams between the generated text and the reference text. This can be useful in this task to ensure that the generated text maintains the same meaning as the original text.

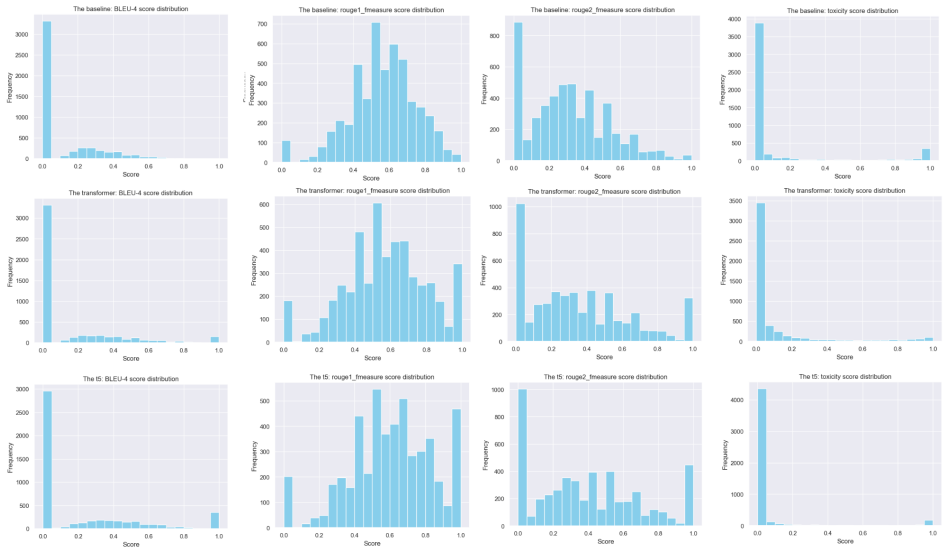


Figure 4: Evaluation on test set

- **Toxicity score** and **toxicity ratio**: These metrics [3] measure the toxicity score values and a proportion of toxic comments in the generated text respectively, using a pre-trained hate speech classification model 'roberta-hate-speech-dynabench-r4' [4]. The goal of a text detoxification task is to reduce toxicity score metrics, while maintaining the meaning and quality of the text.

The metrics were evaluated on a test set containing 5000 sentence pairs. Figure 4 shows the distribution of each evaluation metric. Moreover, the table below presents the performance comparison of created solutions:

	BLEU-2	BLEU-4	ROUGE-1	ROUGE-2	Toxicity ratio
Baseline	0.3129	0.1148	0.5578	0.3063	0.1182
Pytorch transformer	0.3443	0.1499	0.5656	0.3469	0.089
T5 fine-tuned	**0.4236**	**0.2119**	**0.5982**	**0.3921**	**0.0592**

Table 1: Metrics on test set

As observed, the last approach, which involves fine-tuning the T5 model, demonstrates the best metrics. For example, for t5 the toxicity ratio metric is nearly 0.059 while the target detoxified sentences metric stands at 0.062. This indicates that the approach delivers efficient results.

Results

To sum up, the different approaches were created, however, the fine-tuning T5 model provides the most promising results.

However, due to limitations in computational resources (specifically, the absence of a GPU on my laptop, I used the Colab and Kaggle, but the training time was huge), I trained models for a relatively small number of epochs (10). So, increasing the number of epochs would likely provide even better performance.

References

[1] A. Vaswani *et al.*, *Attention Is All You Need*, 2017. Available online: https://doi.org/10.48550/arXiv.1706.03762.

[2] https://huggingface.co/t5-basemodel-details.

[3] https://huggingface.co/spaces/evaluate-measurement/toxicity.

[4] B. Vidgen, T. Thrush, Z. Waseem, and D. Kiela, *Learning from the Worst: Dynamically Generated Datasets to Improve Online Hate Detection*, 2021. Available online: https://doi.org/10.48550/arXiv.1706.03762.