Text Detoxification

Final Solution Report
Roman Makarov

Introduction

This report provides a comprehensive overview of my solution to the text-detoxification task. The following sections detail my processes of data analysis, model selection, training parameters, results, and potential improvements.

Data Analysis

Upon reviewing our primary dataset, I have identified some paraphrases that do not appear natural to me. To address this concern, I decided to incorporate an additional dataset comprising 20,000 human-translated sentences to improve the quality of my dataset.

For the original dataset, I selected only those entries with a reference toxicity score higher than 0.95 and a translation toxicity score lower than 0.01. This method allowed me to gather approximately 170,000 examples of sentences.

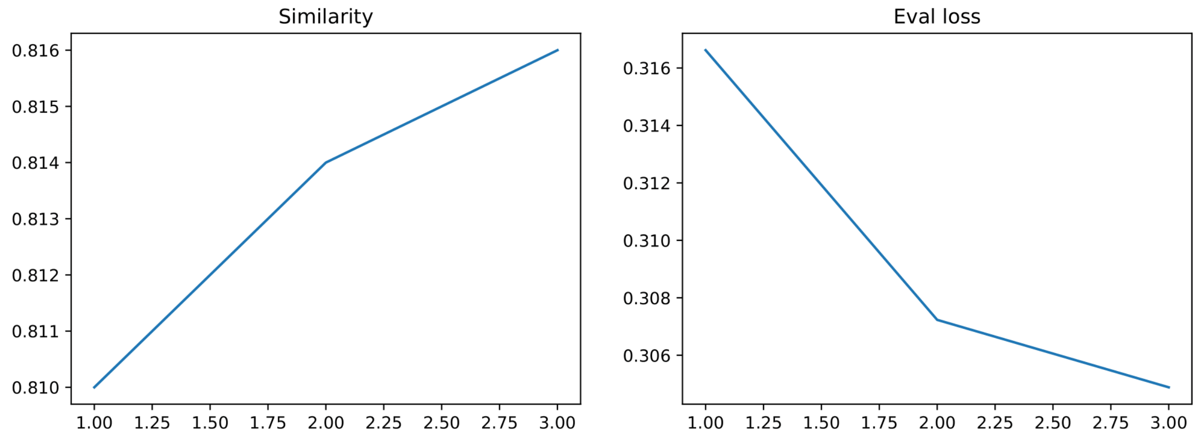
I merged the two acquired datasets to create a final dataset, which has been saved as "paradox.csv." In total, there are around 190,000 samples, with approximately 170,000 (about 90%) originating from the main dataset.

Model Specification

As my final model, I chose pre-trained T5 text-to-text transformer and fine-tuned it on the dataset that I described above.
I used AutoTokenizer from this model's checkpoint to tokenize sentences. I chose 70 to be the maximum length of the sentence, since all sentence examples are shorter than 70 words.

Training process

While training, I calculated the fundamental loss function from Hugging Face, which is the exponential of the cross-entropy loss. Furthermore, I computed BERT text similarity. In my 'Solution Process' report, I explain why I do not use toxicity as a metric during training.
I trained the model for 3 epochs for the reason I discussed in my *Hypothesis 2*.
Upon completing the training process, I achieved a loss value of 0.3 and a BERT similarity score of 0.82 for translations on the test dataset. Below, I present the graphs generated after training the model.



Results

Note, that I intentionally do not measure toxicity rate during training and testing, as it depends on what classifier model considers to be toxic. Additionally, my model has not seen some paraphrases during training, e.g. "to kill someone" - "to put someone down", and a toxicity rate classifier gives 0.99 toxicity rate for the word "kill". Hence, it would not be an appropriate metric and instead, I demonstrate a range of model's paraphrases.

Below, I present translation examples of the most toxic sentences from the main dataset, illustrating how the model successfully removes toxic language and maintains grammatical coherence:

Original: You have to send those idiots back in.
Paraphrase: you have to send them back.

Original: And don't let those idiots in radiology hold you up.
Paraphrase: and don't let those radiologists hold you up.

Original: It's coz of those two idiots. They escaped through a tunnel.
Paraphrase: they escaped through a tunnel.

Original: Audrey Cruz is an idiot. He's the boss.
Paraphrase: Audrey Cruz is a boss.

Original: Your shit is so tired, Justice.
Paraphrase: you're so tired, Justice.

As we can see, model learns to effectively eliminate toxic words and retain grammatical coherence in the translated versions.
What surprised me more, is that the model also learned to paraphrase a sentence in a way that it still makes sense from a grammatical point of view.
The are only two potential drawbacks:

1. At times, the model eliminates entire portions of the text, such as in the case of "*It's coz of those two idiots.*" Nevertheless, this does not significantly impact the overall meaning.

2. I trained the model not on the whole dataset, so there are figures of speech that it has not seen yet, like ‘to kill someone’. Ideally, it should be paraphrased into ‘to put someone down’, but my model just leaves it as it is.

Potential improvements

If resources allow, the large t5 pretrained model can be fine-tuned, instead of a small one that I used due to computational limitations.
Additionally, the model can be trained on the whole corpus of data.