Final Solution Report

Karina Denisova, BS21-DS-01

November 5, 2023

1 Introduction

Text detoxification refers to automatically transforming toxic or offensive text into non-toxic text while preserving the content of the original text. It aims to remove offensive or harmful language from text data, making it more suitable for various applications, such as training language models, chatbots, or ensuring user interactions remain non-toxic. In this report, I will present my approach to solving this task. In my report, I am going to talk about baselines and model training process.

2 Data analysis

The main dataset we use is a subset of the ParaNMT-detox corpus (500K sentence pairs). During EDA we found out that referenced sentences are about 90 percent toxic

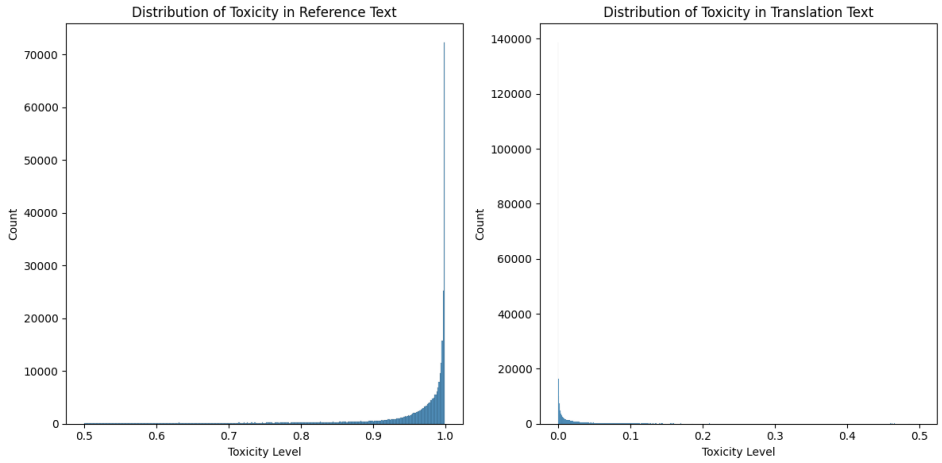

Figure 1: EDA

As it seen from figure 1, the dataset performs extremely high toxic sentences vs non-toxic. these significant differences help us to better teach models to detoxify toxic sentences.

3 Model Specification

In this section, I will describe the model architectures that I use to detoxify text.

3.1 Baseline: Delete toxic words

The first baseline is to delete toxic words from the sentence. To do this, we need to find toxic words in the sentence and delete them. To find toxic words, I use kaggle detest [1]. The dataset contains 1617 unique bad words. To find toxic words in the sentence, I use the following algorithm:

1. Tokenize the sentence

2. Check if the word is toxic
3. If the word is toxic, delete it

3.2 T5 model: Fine-tuning

The second baseline is to fine-tune T5 model [2]. T5 is a transformer-based model that was trained on a large parallel text corpus.

4 GPT-2 model: Fine-tuning

In the future, I plan to fine-tune GPT-2 model [3]. GPT-2 is a transformer-based model that was trained on a large text corpus.

5 Training Process

5.1 T5 model

I use T5 model from huggingface [2]. I fine-tuned the model on the ParaNMT-detox corpus. I use the following hyperparameters:

- batch size: 32
- learning rate: 2e-5
- epochs: 10

To perform better results you can use more epochs and bigger batch sizes or use other model checkpoints, but small. However I have limited resources, so I use these hyperparameters.

6 Evaluation

To evaluate the model, I use the following metrics:

- Accuracy - that performs style transfer accuracy" which is usually estimated by toxicity classifier
- SIM - content similarity: which can be estimated either via cosine similarity between embeddings or as a score from a classifier
- FL - fluency: which can be estimated either via perplexity from LM or as a score from language acceptability classifier
- J - joint metric: multiplication of the three individual metrics
- BLEU - which is a standard metric for machine translation

These metrics are used in the paper [4]. And taken from open source code [5].

6.1 Resuting metrics

For the baseline model, I got the following results:

Model	ACC	SIM	FL	J	BLEU
baseline.txt	0.7739	0.7307	0.7219	0.3861	0.5060

For the T5 model, I got the following results:

Model	ACC	SIM	FL	J	BLEU
result.txt	0.7993	0.6640	0.7617	0.4180	0.4627

For better visualization of the results, I use the following plot:

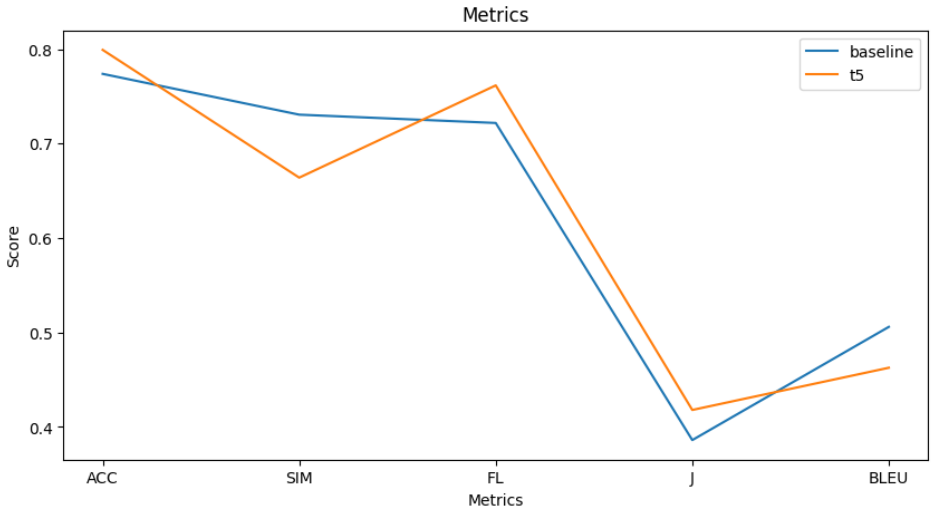

Figure 2: Results

7 Results

As we can see from the table 2, the T5 model performs better than the baseline model. But the results are not very good. I think that the reason for this is the small size of the dataset and small T5 model. In the future, I plan to fine-tune GPT-2 model [3] and use more data and the bigger model to get better results. But I have limited resources, so I for future work I need to finde solution how to use lesser resources, but get better results.

To sum up, I made detoxification experements. As practice show, I need to improve my way to solution to get better result, but I think that I am on the right way.

References

[1] https://www.kaggle.com/datasets/nicapotato/bad-bad-words/

[2] https://huggingface.co/t5-base

[3] https://huggingface.co/gpt2

[4] https://aclanthology.org/2022.acl-long.469.pdf

[5] https://github.com/s-nlp/detox/tree/main