Practical Machine Learning and Deep Learning

Assignment 1

Andrey Starodumov

B20-DS-01

1 Introduction

The goal of the solution is to take the toxic sentence and try to translate it into the sentence with less toxic words.

2 Data analysis

The data provided to us was explored. And here I want to show the distributions of some features.

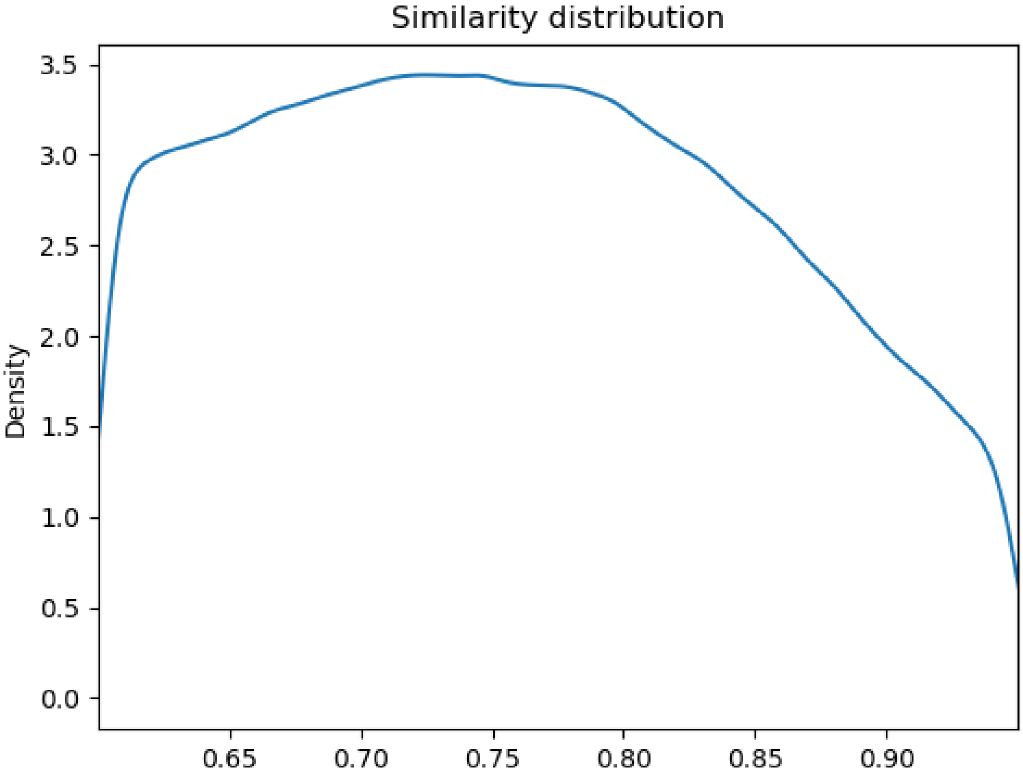

Figure 1: Similarity distribution

It is obvious that the similarity metric is not good [Fig. 1]. Since we need to keep the semantic of the sentence, we do not aim to similarity. So if we want to use this as one of the metric, that the threshold should be 0.6 and not greater.

All non-toxic sentences have the level of toxity up to 0.5, and most of them are lower than 0.1 [Fig. 2].

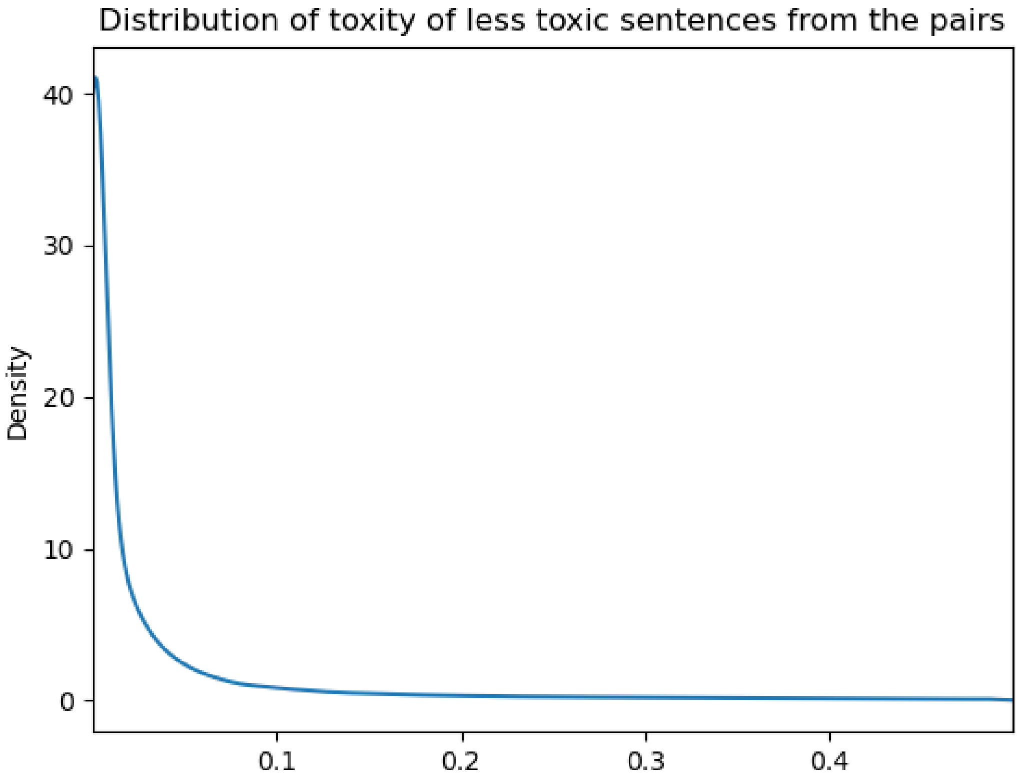

Figure 2: Neutral sentences toxity distribution

All toxic sentences have the level of toxity greater than 0.5, and most of them are greater than 0.9 [Fig. 2]. But you can see that it is not an obvious distribution, that can be called as a **heavy-tailed** one which means that the standard method could not be applied to all of the data.

It means that several methods should be applied.

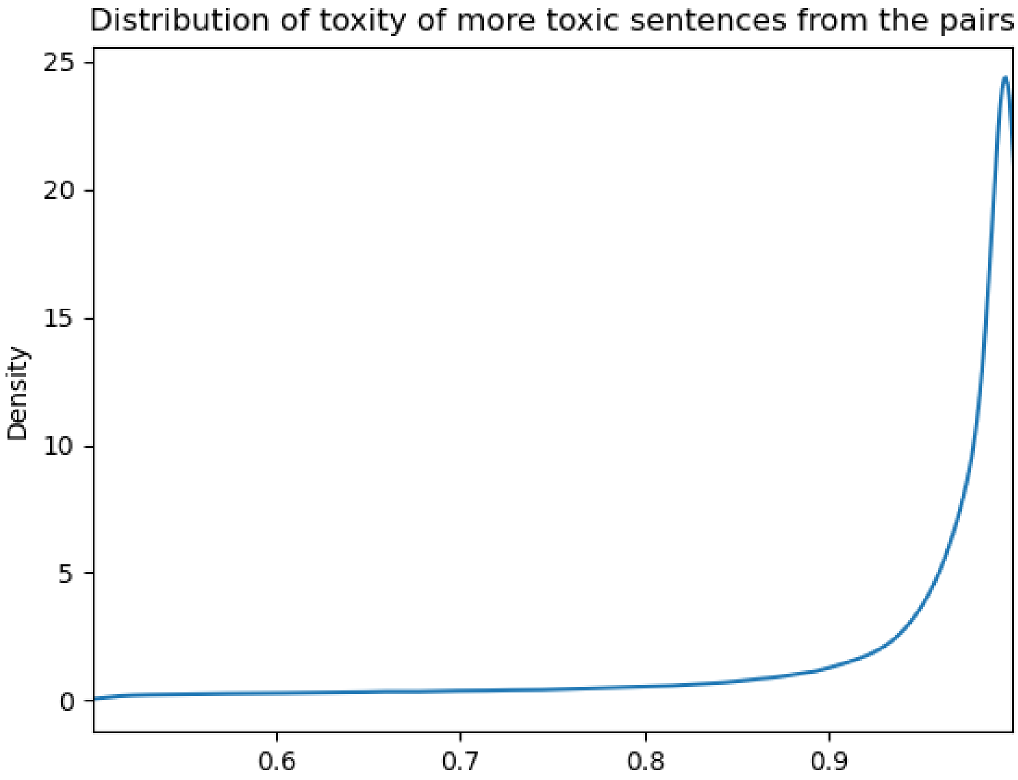

Figure 3: Toxic sentences toxity distribution

3 Model Specification

Because of the reason that was mentioned in the end of 2 section, I decided to use the Pegasus paraphrasing model, since it consider semantic and grammar of sentences. And if the sentence is obviously toxic and consists bad words, the model delete such words or replace with the synonyms.

4 Training process

For the training process I used the simple training loop which is based on model training mode:

1. Tokenize the input sentences;
2. Generate the paraphrased tokens;
3. Tokenize the expected output sentences based on the length of generated one;
4. Put all of that into the model and get the loss function;
5. Backward propogation.

Since it takes too long to preprocess the data as an input, I limited in to use 10 batches.

5 Results

The Evaluation step is just and empirical (visual). I just see the the model paraphrase the sentenses well.

The final state of the model highly depends on the pre-trained model itself. So our training could help in some small changes, but to see them we need more time and data to train the model on some server, not locally.

Overall, the model could really help to paraphrase the sentences in less toxic way, following the rules of the grammar.