Solution Building Report

Nikolay Pavlenko
n.pavlenko@innopolis.university

November 5, 2023

1 Baseline: paraGeDi model

As the baseline for my model I have chosen the pre-trained paraGeDi model, as it was also employed in [1] by Dale D. et al.

The GeDi model (short for Generative Discriminator) comprises two essential components: a generation model, such as GPT-2, and a discrimination model, also built upon GPT-2 but trained on sentences labeled with additional sentence-level styles. During training, the style label is added to the beginning of the sentence. This process allows the discrimination model to learn word distributions conditioned on a specific label. At each generation step, the predicted distribution of the next token by the main language model is modified using an additional class-conditional language model based on Bayes' rule. GeDi successfully guided a GPT-2 language model to generate texts on specific topics and reduce the toxicity of the generated text.

ParaGeDi is an extension of GeDi and focuses on preserving the meaning of the input text. It replaces the typical language model with a paraphrasing model. It models the probability of generating the next token in a sequence by considering the text, desired style, and the paraphrasing probabilities. ParaGeDi involves the training of a paraphraser and a style model independently. Additionally, a reranker, which could be a pre-trained toxicity classifier, can reweigh the hypotheses generated by the ParaGeDi model. The model is trained with a combination of generative loss and discriminative loss, aiming to push different classes away from each other. The ParaGeDi model is further enhanced with several inference heuristics to improve content preservation and style transfer accuracy.

The model is chosen as a baseline due to the demonstrated effectiveness of GPT-2 in reducing toxicity of generated text, so it can serve as an established benchmark against which newer detoxification models can be compared.

2 CondBERT model

The second model which is going to be used in our detoxification task is also taken from [1]. It is a modification of conditional BERT, where BERT is trained to select and replace toxic words in sentences by training a logistic bag-of-words toxicity classifier, which assigns weights to each word based on its importance for classification. Words with higher weights typically indicate toxicity.

The adapted CondBERT model involves multiple steps. First, toxic words are detected in the input sentence. Possible substitutes for these toxic words are generated using BERT, and these substitutes are then reranked based on their similarity and toxicity scores. To determine toxic words, a toxicity score is computed for each word in a sentence, and words with scores above a specified threshold are considered toxic. This threshold is adaptive, balancing the percentage of marked toxic words in a sentence.

To preserve the meaning of replaced words, content preservation heuristics are employed. The model prioritizes original tokens, reranks replacements based on the similarity of their embeddings with the original words' embeddings, and penalizes the predicted probabilities of tokens with positive toxicities. The model also allows BERT to replace a single masked token with multiple tokens using a beam search approach, scoring multitoken sequences by the harmonic mean of their token probabilities. Of course, the more tokens and their combinations are considered, the longer the inference time, so in order to reduce the costs I usually pick the simplest combination for translation.

3 Hypothesis: expansion of the dictionary

I hope to improve the performance of CondBERT by expanding its dictionary size. I will divide the sentences from the original dataset into tokens and will add them to the existing dictionary of the pre-trained model. I will generate a new logistic function to evaluate the toxicity of each token and will re-compile all the datafiles that are used by CondBERT model and depend on it. Since the model will know more toxic and non-toxic words, it should be able to make more reliable predictions.

References

[1] Dale D. et al. Text detoxification using large pre-trained neural models //arXiv preprint arXiv:2109.08914. – 2021.