Toxic Comment Detoxification and Summarization

Abstract

Due to the abundant NLP research in the field of Toxic Comment Classification, well known social media applications, eradicate the toxic comments on a post to maintain the peace between the post's author and commentee. However, in content focused applications like Instagram Reels, Medium, YouTube and the like, such toxic comments provide certain value. Even though they are a result of the commentee's anger, he/she does attempt to put forward some constructive criticism on any part of the content. Since on such platforms authors are not merely people who post content for leisure, but, as content-creators, are people who aim to effectively mirror the society's culture in a domain through their content. For such creators, it's generally their full time profession at which they would like to be good at. Hence, in order to generate better and more relatable content, such constructive criticism is of utmost need.

In this work, we explore the less worked on area of automatic extraction of constructive criticism from toxic comments. We, first, rephrase the toxic comments in a non-toxic style, called Text Detoxification [1]. We, then, summarize this detoxified text to extract the crux of the criticism provided. We aim to test the final constructive criticism extracted from a given set of comments by human evaluation, after using the Joint Metric for rephrasing and ROUGE [10] for summarization. We also study some applications of text detoxification in other social media application systems.

Problem Statement

The task of Automatic Constructive Criticism Extraction can be broken down into 2 tasks: Text Detoxification and Text Summarization. This is a two-step process since we found no datasets, which, using a set of toxic comments, provided a non-toxic summary of the content in the text given.

Text Detoxification

Text detoxification [1] is the process of converting a text in a toxic style to a text in a non-toxic style. This is a style transfer problem, which means that the content of the text being detoxified should not change. The only attributes which should change are politeness and sentiment. In Cloak's Hypothesis [14] of languages we saw that language of a text is nothing but a 'cloak' which ‘covers’ the latent meaning of the text. In order to understand the meaning of the text, we must remove the cloak. The aim of text detoxification is to discard toxic language constructs, and train the machine to understand the latent meaning of the text and rephrase it into another ‘cloak’, i.e. language style, which is non-toxic and more polite to the reader.

Text Summarization

At the end of this step, we would have the toxic comments written in a non-toxic style and hence, would contain all the criticism that we aimed to extract. However, since multiple comments can criticize the same aspect, reporting only the rephrased version, will not provide the intended feedback, since it’ll be just a big block of text with the criticism having repeated points. This is where text summarization is required. For the practical use of extracted criticism, it should be well summarized for clear understanding of the criticism by the reader and hence, he/she would be able to rectify those points in their next content release. Hence, we then build a model to summarize the criticism text obtained in the last step. This would give us a well extracted crux of the criticism given in various toxic comments.

Final Human Evaluation Steps

We scrape toxic comments from the comments section of a piece of content from any of the content focused applications. We then combine all the toxic comments to one text and send it through the text detoxification model. We send the resultant detoxified text through the text summarizer model, in order to get the comments in a detoxified, summarized style. This text is now sent for human evaluation. Figure 1 diagrammatically shows this flow.

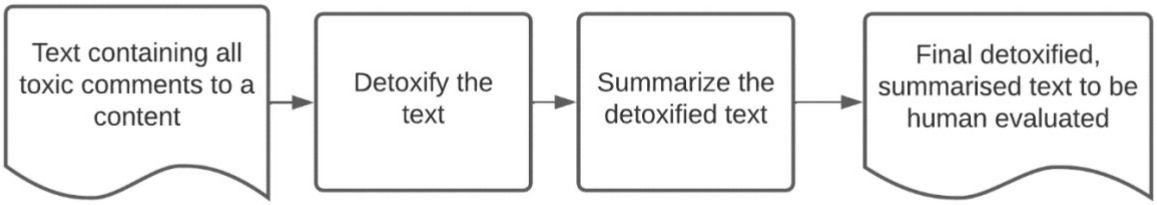

Text containing all toxic comments to a content
Detoxify the text
Summarize the detoxified text
Final detoxified, summarised text to be human evaluated

Fig 1. Flowchart depicting the inference steps the final model does.

Datasets

We intend to use the following datasets for the above mentioned tasks:

- Jigsaw Toxic Comment Classification Dataset [13]:

We use this dataset for estimating the toxicity of each word in a sentence. We then take the words whose toxicity is above a threshold and feed those to the BERT model. This model now predicts the possible non-toxic, content-preserving replacements for this word. We replace the toxic word by the words predicted by BERT and hence, removing all toxic phrases from the sentence, in order to get the same text in a detoxified, civil style. This method is elaborated in the 'Models' section below.

- CNN Daily Mail Summarization Dataset [12]:
  We use the CNN Daily Mail dataset which is inbuilt into TensorFlow to test the summarizer algorithm. It is a rich dataset containing 2 fields 'article', where each entry is the text to be summarized and 'highlights', the summary of the aforementioned text. We selected a journalism based dataset as it covers a wide range of topics and hence, texts having varying content are summarized in such datasets. The rephrased comment text could be based on a wide variety of domains and hence, the wide variety of text in the dataset helps in the effective summarization of the provided comment text.

- Dataset for human evaluation:
  In order to perform a final evaluation of the obtained summary from the set of toxic comments, we intend to present to humans and evaluate the summarization result. For this task, we require to curate a set of toxic comments from any content on content focused applications. The need to curate such a dataset arises as we did not find a set of comments related to one content in any dataset. The dataset would, hence, be curated by scraping from the sites containing the content and its comments. It would then be passed through the models we build for doing the above two tasks to generate the summary. We, then, present the summary to a human who would then rate the summarization on a certain scale.

Models

- Text Detoxification:
  Our approach to text detoxification involves replacing toxic words with their non-offensive synonyms. To meet the objective, **BERT (Bidirectional Encoder Representations from Transformers) model** is trained with the **MLM (Masked Language Modeling)** task [1]. Masked Language Modeling is a fill-in-the-blank task, where a model uses the context words surrounding a mask token to predict what the masked word should be. The masked word can then be substituted with the predicted word, generating a new sentence that shares similar context and same label with the original

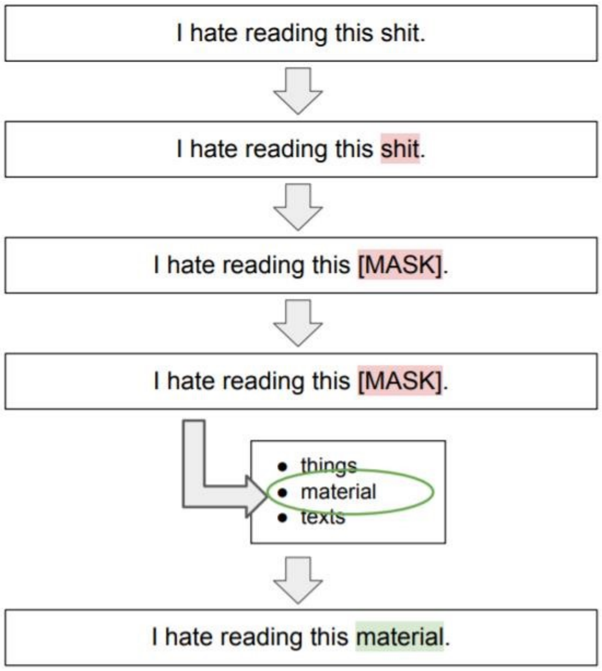
sentence. MLM consists of giving BERT a sentence and optimizing the weights inside BERT to output the same sentence on the other side. However, before actually giving BERT the input sentence, a few tokens are masked. While in the original BERT model, the words are masked randomly, for our purpose, we would be selecting the words associated with toxicity.

Now, in order to identify toxic words, we will train a **Bag-of-Words (BoW) toxicity classifier**, a logistic regression model that classifies sentences as toxic or neutral using their words as features. Since we'd be using a simple logistic regression model, with the help of the coefficient values of the model, we should be able to identify which features are important or which words make a sentence toxic. Therefore, the weight acquired for a word will be interpreted as its toxicity level and it will be considered toxic if its weight is higher than a predefined threshold.

Not to forget, we would be doing some **pre-processing steps**, namely, converting all characters to lowercase, converting chat conversion words to normal words, removing special characters, removing numbers, removing extra spaces, removing stop words, spelling correction, stemming, and lemmatization, to increase the performance of BoW.

Finally, so as to make a choice of the best replacement word, the replacement words suggested by BERT will be re-ranked by their similarity and toxicity. Similarity will be determined by the **cosine similarity (CS)** between the vector representations of the input and output sentences, as taught in class. And since BERT may predict toxic words [1], to force the model to replace toxic words with words that have close meanings but are not toxic, the toxicity levels of the words in the BERT vocabulary, obtained earlier, can be used to penalize the retrieved toxic words.

The approach described above is as illustrated below [2]:


I hate reading this shit.
I hate reading this shit.
I hate reading this [MASK].
I hate reading this [MASK].
things
material
texts
I hate reading this material.

- Text Summarization:
  We shall use **TextRank**, which is an extractive and unsupervised text summarization technique, to generate a summary that conveys useful information, without losing the overall meaning.

Evaluation Metrics

- Joint metric:
  The performance of any style transfer model is typically evaluated by accessing how well it:
  - Changes the text style
  - Preserves content of the input text
  - Produces a fluent text

  The geometric mean (GM) of style transfer accuracy (STA), sentence similarity score (SIM), and inverse of fluency measured by perplexity (PPL), takes these parameters into account and so, can be used to evaluate text detoxification. We are yet to decide upon the pre-trained models that we plan to use for each of the individual metrics.

- ROUGE [10]:

We wish to explore ROUGE, standing for Recall-Oriented Understudy for Gisting Evaluation, which is essentially package containing a set of metrics for evaluating text summarization.

- Human evaluation:
  So as to draw attention to our point that we need not have to do away with every toxic comment, we'd like to conduct manual evaluation on the dataset previously described.

Applications

Various online platforms nowadays allow any Web user to share his or her opinion on an arbitrary content with a broad audience. For example, the comment sections of most online platforms are an essential space where users exercise their right to freedom of expression in the Web. However, 'toxic comments', which are defined as a rude, disrespectful, or unreasonable comments that are likely to make other users leave a discussion, are a problem for these platforms. Effective countermeasures for toxic comments online involve identifying such comments so as to **remove or moderate** them. Many social media platforms, like Instagram, provide tools for removal of messages based on automatically identifying harmful content. Of course, another way to approach the problem is rewriting toxic messages instead of removing them altogether. Or suggesting edits rather than performing them automatically. Which is where, text detoxification to eliminate toxicity in text, can find an important application.

That said, we'd also like to highlight another application, related to 'constructive criticism', which is what had inspired our interest in the given task. **Constructive criticism** is a useful method of giving criticism that involves providing actionable feedback in a friendly manner. It can give one a new perspective and even open their eyes to things they may have overlooked or never considered, thus helping them grow by giving them the opportunity to improve. Constructive criticism embodied in online comments is of significant value to bloggers, vloggers, writers, YouTubers and other content-creators. Constructive criticism in online reviews is very helpful to companies, and also to other consumers evaluating their products and services. Now, it is entirely possible to extract such criticism from 'toxic comments' if they can only be detoxified. For example, a comment such as, "Your characters have very stupid names," detoxified as "Your characters have very strange names," can be very beneficial to a writer. Our intention behind text summarization post text detoxification is to ensure that given a substantial number of toxic comments pertaining to anything particular, we're looking at non-redundant data while making sure that we are not making a mountain out of a molehill. Also, it is to ease observability for those participating in human evaluation of our model.

References

[1] Text Detoxification using Large Pre-trained Neural Models
https://aclanthology.org/2021.emnlp-main.629.pdf

[2] Methods for Detoxification of Texts for the Russian Language
https://www.mdpi.com/2414-4088/5/9/54/htm

[3] Masked-Language Modeling with BERT
https://towardsdatascience.com/masked-language-modelling-with-bert7d49793e5d2c#

[4] An introduction to Bag-of-Words (BoW)
https://www.mygreatlearning.com/blog/bag-of-words/

[5] How Bag-of-Words (BoW) works in NLP
https://dataaspirant.com/bag-of-words-bow/

[6] Popular NLP Text Pre-processing techniques
https://dataaspirant.com/nlp-text-preprocessing-techniques-implementation-python/

[7] Text Classification - From Bag-of-Words to BERT
https://medium.com/analytics-vidhya/text-classification-from-bag-of-words-to-bert1e628a2dd4c9

[8] Toxic Comment Detection in Online Discussions
https://hpi.de/fileadmin/user_upload/fachgebiete/naumann/publications/2019/risch2019toxic.pdf

[9] An Introduction to Text Summarization
https://www.analyticsvidhya.com/blog/2018/11/introduction-text-summarizationtextrank-python/

[10] ROUGE: A Package for Automatic Evaluation of Summaries
https://aclanthology.org/W04-1013.pdf

[11] An introduction to ROUGE, and how to use it to evaluate summaries

https://www.freecodecamp.org/news/what-is-rouge-and-how-it-works-for-evaluationof-summaries-e059fb8ac840/

[12] CNN Daily Mail Summarization Dataset
https://www.tensorflow.org/datasets/catalog/cnn_dailymail

[13] Jigsaw Toxic Comment Classification
https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge/data

[14] Class Notes: Mandate 1
https://docs.google.com/presentation/d/1QgHnpi2lRSGoCPkOTcwIXxASr945CEdXDbjNhZ5qqzo/edit#slide=id.p