Progress Report 1

Jayveersinh Raj, Makar Shevchenko, Nikolay Pavlenko

March 7th, 2023

1 Current Progress

1.1 Research

1.1.1 Primary NLP Topics

Before moving forward with the project, we had to map out some important NLP-relevant topics we would be trying to cover while building this model. After discussing them, we have accepted **Sentiment Analysis** to be the primary NLP technique we will be implementing in our model. To narrow down the task, we have chosen the implementation of **Toxic Comment Classification** as our current goal. Hopefully, in the future we will also be able to implement **Machine Translation** from English into Romanian, thereby demonstrating that our model can work on more than one language effectively.

1.1.2 Datasets Exploration

After setting our goals as described above, we started searching for various datasets that would allow us to develop the model. In the end, three most relevant datasets were found:

1. Personal attacks dataset from Wikipedia's Detox project[1],
2. Jigsaw and Google toxic comment analysis dataset[2],
3. IMDB ratings sentiment dataset[3].

1.1.3 Results

To jump-start our project, we have begun working with the Jigsaw dataset first. It was chosen over the other 2 datasets, as it is more vivid than the competition, and its classification of toxicity into several levels is very useful in the particular problem we are trying to solve. This dataset is also high-quality and easy to clean, so it makes for a good starting point from where we can get some results from the model.

1.2 Data Understanding

1.2.1 Initial Data Collection

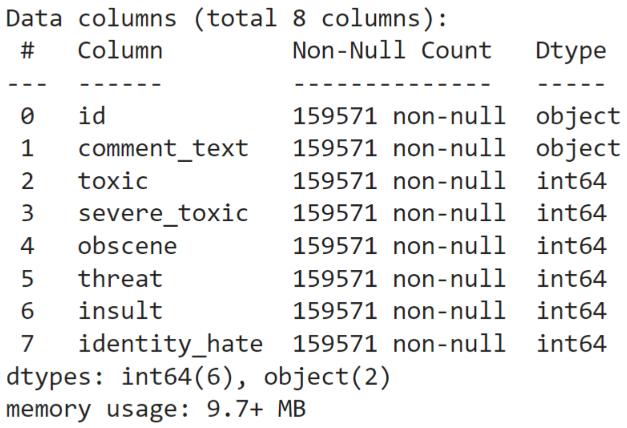
In order for a dataset to be relevant in a Sentiment Analysis task (and specifically in our case of toxic comment classification), it has to fulfill certain conditions:

1. It must contain text data of messages that are to be classified.
2. It must contain labeled data, that would attach a specific level of toxicity to each message.
3. It has to be large enough for the model to be able to train effectively.
4. The dataset has to be diverse enough for the developed model to generalize better.
5. The dataset should ideally be high-quality, devoid of noise, missing data and having balanced labels across different classes.

We have used all those conditions as guidelines while searching for datasets for our model and performing data preprocessing.

1.2.2 Data Description

Dataset chosen for initial development of our model contained the following features:

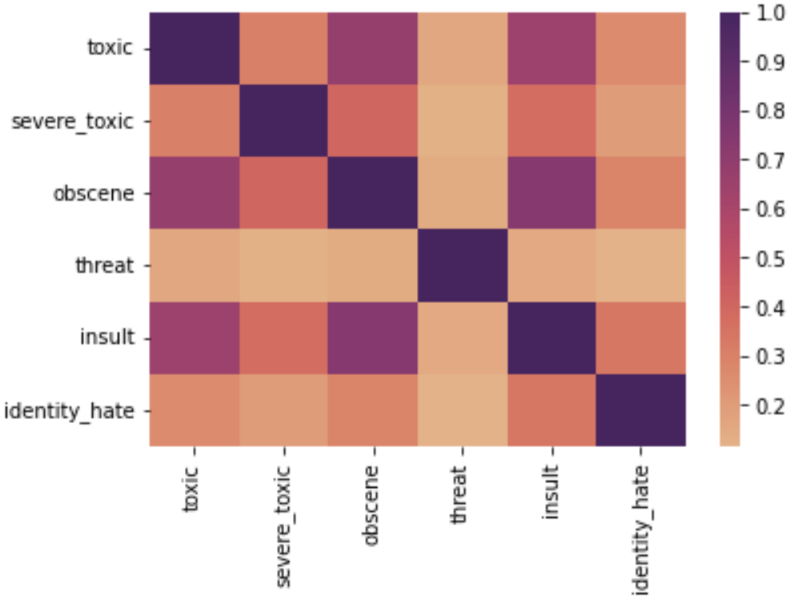


```
Data columns (total 8 columns):
 #   Column         Non-Null Count   Dtype 
---  ------         --------------   ----- 
 0   id             159571 non-null  object
 1   comment_text   159571 non-null  object
 2   toxic          159571 non-null  int64 
 3   severe_toxic   159571 non-null  int64 
 4   obscene        159571 non-null  int64 
 5   threat         159571 non-null  int64 
 6   insult         159571 non-null  int64 
 7   identity_hate  159571 non-null  int64 
dtypes: int64(6), object(2)
memory usage: 9.7+ MB
```

comment_text column is the one we will be using to develop the toxicity classification, while columns from **toxic** to **identity_hate** are binary non-exclusive labels describing various toxicity levels present in the comments.

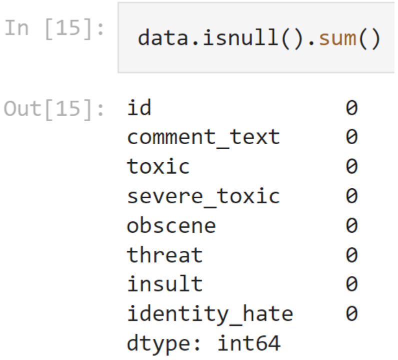
1.2.3 Data Exploration

In order to explore our dataset, we have gathered important statistical characteristics of all the labels in it, and plotted graphs and a heatmap that highlight class imbalances and correlations between different features:



1.2.4 Verification of Data Quality

In order to make sure that the dataset is of a relatively high quality, we have checked if it contained any null values - thankfully, it did not:



```
In [15]:  data.isnull().sum()

Out[15]:  id               0
          comment_text     0
          toxic            0
          severe_toxic     0
          obscene          0
          threat           0
          insult           0
          identity_hate    0
          dtype: int64
```

1.3 Data Preprocessing

During data preprocessing a few important changes were introduced to the dataset:

1. **id** column was dropped, as it was irrelevant in the task our model is set to achieve.

2. Due to extremely sparse nature of the binary labels in the dataset, we have decided to merge them into a single true/false column, that is supposed to represent any level of toxicity found in a comment. It is a temporary measure, and the initial dataset with multiple toxicity labels was preserved for future work.

3. Since various features were found to be heavily imbalanced during data exploration, data was resampled to allow for the same number of toxic and non-toxic comments in the dataset.

1.4 Modeling

1.4.1 Architecture

In order to be able to classify the comments according to their toxicity level, we first have to encode the text data. We use the pre-trained **BERT encoder** for that purpose, transforming the text to classifiable tensors.

The initial attempt at classification was done with the help of **Support Vector Machine(SVM)**, implementation taken from sklearn library. However, it failed due to extremely long inference time of the model - from 15 to 35 seconds per sample, which is far too long considering the large size of our dataset.

For that reason we have decided to avoid SVM and instead focused on **XLM-R-Classifier**, which is supposed to work well on multilingual data - that makes it very relevant for the particular conditions of our project.

1.4.2 Hardware

Currently we have trained our model using Google Colab, as it provides a decent platform to start developing our model on, though we are considering options on improving available hardware resources in the future.

1.5 Evaluation

As mentioned prior, no results have been achieved with the help of the BERT encoder + SVM classifier, with BERT working extremely slowly on Google Colab, taking 3 seconds per sample.

On the other hand, XLM-Roberta embedder used in conjunction with XLM-Roberta classifier led to very good results. It worked well on several Indo-European languages: English, Hindi, French, etc., and reached an F1-score of 0.96 in particular on the english dataset, using a 90-10 training/validation split.

2 Team Member Contribution

- **Jayveersinh Raj**: datasets exploration, development and evaluation of XLM-Roberta embedder-classifier pipeline;
- **Makar Shevchenko**: data understanding and preprocessing, development of BERT encoder - SVM classifier pipeline, organizational responsibilities;
- **Nikolay Pavlenko**: progress evaluation, composition of intermediary report.

3 Plan for the Next Three Weeks

Considering the results achieved during the previous weeks, we have decided to continue the development of the XML-Roberta embedder + XLM-Roberta classifier, while pausing and reevaluating the BERT-SVM pipeline until better hardware resources could be obtained. Until then, some other pipeline could be developed using other classifiers from sklearn library.
We will further seek to collect more data to augment the existing selection, particularly focusing on non-English datasets, so we would have the possibility of testing zero-shot cross-lingual transfer.

4 Github Link

Notice: The repository is private, so if you want to check the current state of the project, please write to *Makar Shevchenko* on Telegram to get access to it.
Link:*https://github.com/SyrexMinus/cross_lingual_nlp*

References

[1] https://figshare.com/articles/dataset/Wikipedia_Talk_Labels_Personal_Attacks/4054689

[2] https://www.kaggle.com/competitions/jigsaw-toxic-comment-classification-challenge/data

[3] https://ai.stanford.edu/ amaas/data/sentiment/