Natural Language Processing

Project Report

Hate Speech Spreader Detection

Miki Mizutani
miki.mizutani@studio.unibo.it

Anna Fabris
anna.fabris@studio.unibo.it

Leonidas Gee
leonidas.gee@studio.unibo.it

Academic Year: 2021 / 2022

Abstract

A classification task was successfully performed on a dataset of Twitter tweets to accurately identify whether Twitter users are spreading hate speech or not. Four different methods (SVM, BiLSTM, BiGRU, and BERTweet) were trained and evaluated on the dataset. Our results show that the BERTweet transformer method produces the best results in terms of accuracy on the test set. We further analysed the model results and discussed our findings and possible improvements for the models.

1 Introduction

In the age of online debate, the spread of divisive discussions through social media has become a source of concern. The ability to automatically detect hate speech spreaders has become a crucial task for social media giants. Due to the way tweets are structured, containing short and frequently truncated terms, the challenge is the semantic understanding of whether a tweet contains a hateful undertone with limited semantic information.

For this project, we utilised the PAN-AP-2021 dataset that was used to profile hate speech spreaders in social media, more specifically on Twitter, addressing the problem in English. The organisers of PAN 2021 aimed to investigate the problem of hate speech detection from the author profiling perspective. The final goal was to profile authors who had previously published hate speech. This would represent a first step in preventing hate speech from being propagated among social media users by identifying potential hate speech spreaders on Twitter.
The remainder of the report is organised as follows:

- **Section 2 - Related works**: covers the related works to hate speech spreader identification
- **Section 3 - Method**: describes the corpus and the methodologies used
- **Section 4 - Results**: discusses the results
- **Section 5 - Conclusion**: draws the conclusions

2 Related Works

While the focus on online hate speech has increased lately, most studies focus on hate speech detections on a text (or tweet), with little research focusing on detecting hate at the user account level. On the

other hand, the recent release of the PAN-AP-2021 dataset is a continuation of the contribution to author-profiling. From the PAN-AP-2021 hate speech detection competition submissions, to classical hate speech detection research, there were several resources to base our project on. An example is [1]. It uses two encoding approaches, TF-IDF and Bertweet (an encoder pre-trained on English tweets). Another one [5] proposes an SVM approach, achieving an accuracy of 0.78 - 0.74. Lastly, [3] proposes hate speech detection on social media via transfer learning with BERT. These previous works show that a wide range of methods have been implemented successfully, from traditional approaches such as an SVM, to more modern approaches such as using twitter specific transformers.

Below is a table of baseline performances using different architectures provided by PAN, using the dataset on Hate Speech Spreaders identification.

Method	**Accuracy**
LDSE	70.0
SVM + char n-grams	69.0
NN + word n-grams	65.0
USE-LSTM	56.0
XLMR-LSTM	62.0
MBERT-LSTM	59.0
TFIDF-LSTM	61.0

Table 1: Baseline accuracy on the dataset provided by PAN [2].

3 Method

3.1 Dataset

The PAN-AP-2021 English corpus contains a set of 60000 tweets from 300 different anonymised Twitter users. The users were identified as potential hate speech spreaders using a keyword-based approach (e.g. by searching for hateful words towards a particular group) and a user-based approach (e.g. by searching for users appearing in reports and by following their social network).

For each user, the last 200 tweets have been retrieved using the Twitter API and annotated. The corpus is uniformly balanced with 150 users for each class (hate and non-hate speech spreaders), as well as the corresponding number of tweets per user.

A partial preprocessing had already been carried out: images, URLs, code snippets, block quotes and bullet lists were already removed. For this project, we only had access to the training set, which includes 40000 tweets from 200 users.

Each tweet is fed separately into the models. Grouping tweets of the same user into a single or multiple vectors was tested, but as this reduced the number of training instances, it did not provide optimal results.

3.2 Preprocessing

Due to the short structure of tweets, users will typically compress semantic information using emojis or abbreviations. As such, it is necessary to pre-process the text to create a clearer semantic structure. Each tweet was first pre-processed using the following methods:

- Remove the dataset specific terms and other html leftovers.
- Expand contractions (e.g. you're → you are).
- Normalise sequences of at least 3 repeated characters with a maximum of two letters (e.g. hiiiiii → hii).
- Remove numbers.
- Remove punctuations.
- Transform emojis into their aliases.
- Remove extra white spaces.
- Remove any left or right spacing

The cleaned dataset is then divided and shuffled into training, validation and test sets of 28000, 4000, and 8000 instances respectively.

3.3 Modelling

To model the data, 4 different models were chosen: SVM, BiLSTM, BiGRU, and BERTweet. The models represent both classical approaches as well as the latest developments in language modelling for NLP.

Encoding methods

Each model utilised a different encoding method for the cleaned tweets with the exception of the BiLSTM and BiGRU which shared the same encodings. The following encodings were used:

- SVM: TF-IDF with no stop words removal.
- BiLSTM/BiGRU: GloVe embeddings with random vectors for out-of-vocabulary tokens.
- BERTweet: Embeddings from pre-trained BERTweet on sentiment analysis.

Hyperparameters tuning

Each model with the exception of the transformer was optimised using random search on their respective hyperparameters based on their validation accuracy. The following hyperparameters were tuned:

- SVM: 'C' (distribution between 0 and 1), 'kernel' ('poly', 'rbf', 'sigmoid')
- BiLSTM: 'LSTM units' (distribution between 64 and 320 with step 64)
- BiGRU: 'GRU units' (distribution between 64 and 320 with step 64)

Using random search, the optimal parameters for the three models above were found to be:

- SVM: 'C' (0.59), 'kernel' ('rbf')
- BiLSTM: 'LSTM units' (128)
- BiGRU: 'GRU units' (256)

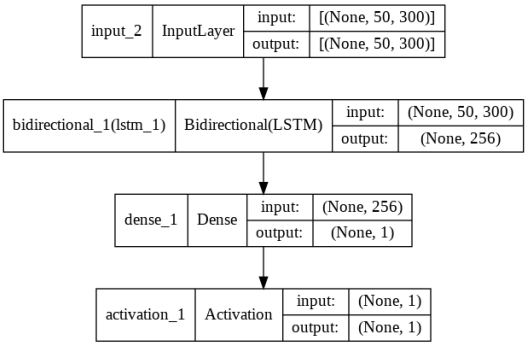

Figure 1: Optimised BiLSTM architecture

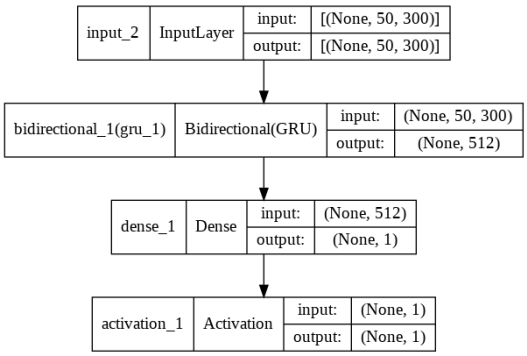

Figure 2: Optimised BiGRU architecture

Training

The pre-trained BERTweet model has been trained on the SemEval 2017 corpus (around 40k tweets) for semantic analysis of tweets in English [4]. The base model is a RoBERTa model trained on English tweets.

The optimised BiLSTM and BiGRU were retrained from scratch using 10 epochs, while the pre-trained BERTweet was fine-tuned using 10 epochs. All three neural network models used a batch size of 32 and early stopping for training.

The final trained models were then evaluated on the test set by majority voting on the tweet predictions for each author in order to classify them as either a hate speech spreader or not. The accuracy was taken into account for comparing the models. A confusion matrix was also plotted for each model to better visualise the predictions and errors.

4 Results

Method	Test Accuracy
TF-IDF + SVM	76.0
GLoVe + BiLSTM	64.0
GLoVe + BiGRU	67.0
BERTweet	78.0

Table 2: Test accuracy on the dataset.

Based on our achieved results, we note certain interesting findings. Firstly, we were surprised at how well a simple TF-IDF + SVM approach models the data. Our baseline approach achieves a performance comparable to models with a much higher model capacity such as a fine-tuned BERTweet with much less training time.

Secondly, we note how a pre-trained BERTweet on a similar task such as sentiment analysis of tweets, which is subsequently fine-tuned, seems to help in modelling the data well. Perhaps due to the similarity of the pre-trained task from a semantic understanding perspective (parsing sentiment from tweets), the model is able to quickly model the new data with few epochs.

Finally, we acknowledge how the same number of epochs given to the BiLSTM and BiGRU models as we did for BERTweet may have unfairly limited their ability to fully learn from the data. In our experiments, depending on the optimised architecture from the random search, doubling the number of epochs from 10 to 20 seems to produce better performance (approximately 0.7 or higher test accuracy) for both models.

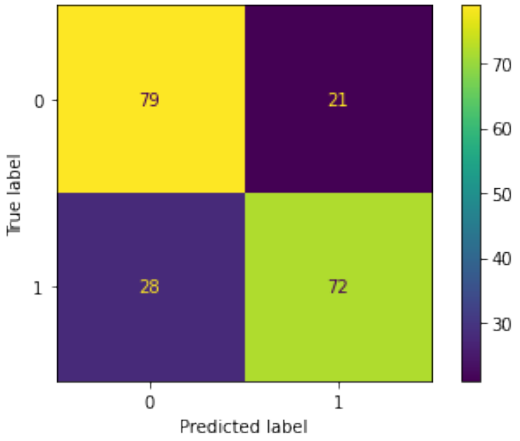

Figure 3: Confusion matrix of SVM

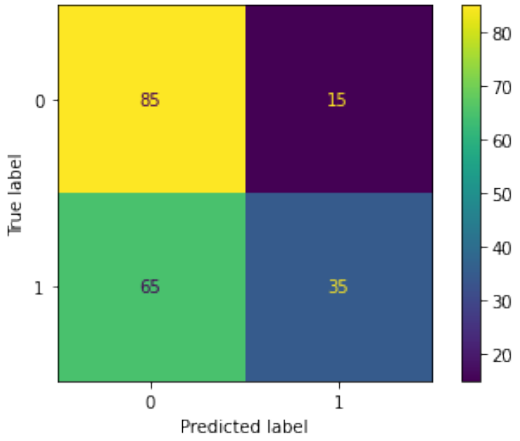

Figure 4: Confusion matrix of BiLSTM

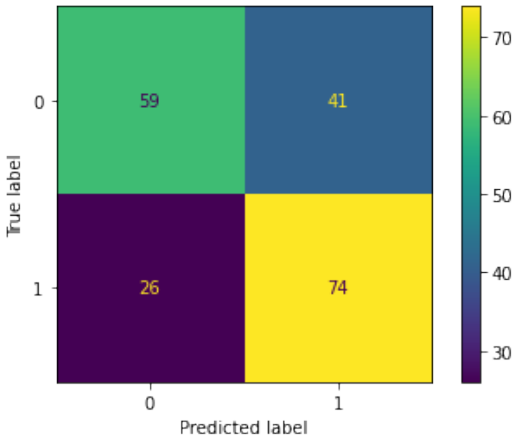

Figure 5: Confusion matrix of BiGRU

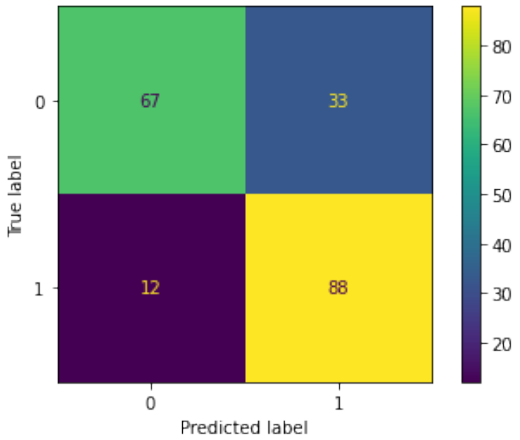

Figure 6: Confusion matrix of BERTweet

Given the confusion matrices, it can be seen that most of our models have lower precision on

detecting non-hate speech spreaders. One hypothesis for this is that due to our training of the models on a short tweet level (instead of concatenating the tweets into long texts), our models may be associating certain neutral statements from hate speech spreaders as hate speech tweets. This in turn causes the models to think that neutral statements from non-hate speech spreaders are in fact hate speech, thus biassing the final majority voted predictions towards hate speech spreaders.

5 Conclusion

Given our results, we can say that they are comparable to those obtained in others recent works. The accuracy on this task remains relatively low especially compared to the classic hate speech detection task. This certainly depends on the ambiguity of the task as classifying whether a tweet is hateful is much easier than classifying users as hate spreaders.

In turn, we propose some potential tweaks and approaches to try to improve the performance on the dataset:

- **N-grams**: using n-grams as features is a popular approach that could be tried.
- **CNNs**: training a Convolutional Neural Networks.
- **Combining BERT with SVM**: extracting the pretrained embeddings from BERT and using them as input to the SVM.
- **Exploiting relationships between users**: hate speech spreaders are more likely to be connected to other hate speech spreaders.
- **Modifying the majority voting threshold**: we classify users as hate spreader if more than 50% of their tweets are hateful tweets, trying other numbers could improve the results.

References

[1] Kumar Das et al. "Profiling Hate Speech Spreaders on Twitter-Notebook for PAN at CLEF 2021". In: Sept. 2021, pp. 21–24.

[2] Webis Group. *Profiling Hate Speech Spreaders on Twitter*. `https://pan.webis.de/clef21/pan21-web/author-profiling.html`. Accessed 26 January 2022. 2021.

[3] Marzieh Mozafari, Reza Farahbakhsh, and Noel Crespi. "A BERT-Based Transfer Learning Approach for Hate Speech Detection in Online Social Media". In: Dec. 2019, pp. 928–940.

[4] Juan Manuel Pérez, Juan Carlos Giudici, and Franco Luque. *pysentimiento: A Python Toolkit for Sentiment Analysis and SocialNLP tasks*. 2021. arXiv: `2106.09462 [cs.CL]`.

[5] Juan Pizarro. "Using N-grams to detect Fake News Spreaders on Twitter". In: *CLEF*. 2020, pp. 22–25.