Hate speech spreader detection using contextualized word embeddings

Notebook for PAN at CLEF 2021

Evgeny Finogeev[1], Mariam Kaprielova[1,2], Artem Chashchin[1,3], Kirill Grashchenkov[1,3,4], George Gorbachev[1] and Oleg Bakhteev[1,2,5]

[1]Antiplagiat, Moscow, Russia
[2]Moscow Institute of Physics and Technology (MIPT), Moscow, Russia
[3]Skolkovo Institute of Science and Technology (Skoltech), Moscow, Russia
[4]Institute of Oceanology (IO RAS), Moscow, Russia
[5]Dorodnicyn Computing Center RAS, Moscow, Russia

Abstract

he paper presents a method of hate speech spreaders recognition developed for the task of "Profiling Hate Speech Spreaders on Twitter" at the PAN@CLEF Conference 2021. Hate speech is an increasing problem nowadays. Due to the spread of the internet and the rise of social media, people with hostile views towards certain groups post hateful messages on social media resources. In this paper, we present a model to detect hate speech spreaders based on their Twitter posts. We aggregate contextualized embeddings of single tweets to form a vector representation for every user and employ classification methods to find users spreading hate speech. We analyze different embedding models based on BERT architecture for the problem. The submitted model achieves 67% in terms of accuracy for the English part of the dataset and 83% for the Spanish part of the dataset.

Keywords

BERT, contextualized word embeddings, deep learning,

1. Introduction

Hate speech detection in social networks is a significant social issue nowadays. A large number of works is devoted to hate speech detection in the social media domain [1, 2, 3]. There are variety of methods of hate speech detection. Most of them consider the problem as a supervised document classification task [1]. The approaches can be divided into two categories: either based on manual feature engineering [2, 3] and using classic methods or deep learning-based models which employ neural networks to automatically learn abstract features from raw data [4, 5].

Hate Speech Spreaders profiling task considers the problem of hate speech detection in social networks [6, 7]: given a set of tweets written by a user, one should establish, whether the user can potentially spread hate speech or not. The tweets are written in two languages: English

CLEF 2021 – Conference and Labs of the Evaluation Forum, September 21–24, 2021, Bucharest, Romania
finogeev@ap-team.ru (E. Finogeev); kaprielova@ap-team.ru (M. Kaprielova); chashchin@ap-team.ru (A. Chashchin); grashchenkov@ap-team.ru (K. Grashchenkov); gorbachev@ap-team.ru (G. Gorbachev); bakhteev@ap-team.ru (O. Bakhteev)

and Spanish. The training dataset contains 200 sets of tweets for each language. There are 100 sets for users that can spread hate speech and 100 sets for ordinary users. The performance metric for this task is accuracy.

The main challenge of the current task is that the analysed object is not a single tweet, but a collection of messages posted by a user. Even if the developed method detects a content without hate speech in some of the user's posts, the same user still can be classified as a potential hate speech spreader. In this way, hate speech detection is similar to the previous fake news detection task [8]. The key idea for many approaches to such tasks is to aggregate information contained in all the texts written by the author. We employ a neural network-based approach for such aggregation.

Recent studies show that approaches, based on embedding generation and their further classification can be quite effective in hate speech detection. For example, in [9] authors present a solution involving hybrid embeddings based on TF-IDF for word-level feature extraction, LSTM for sentence-level feature extraction and also naïve Bayes to extract topics from tweets. Those embeddings are then used as an input for improved cuckoo search neural network. In [10] authors proposed an approach which first used LSTM to perform word embeddings and then CNN was used for hate speech classification. One of the most impressive deep learning-based architectures to obtain semantic information stored in short texts is BERT [11]. The embeddings obtained from BERT are used to classify whether a user can spread hate speech or not. In this paper, we employ different BERT-based architectures to obtain embeddings for every tweet in the collection. We also analyze model performance after fine-tuning on the external corpus of tweets [12].

2. Methodology

The following section describes the proposed method of hate speech spreader detection.

Similar to the previous year's task [13], we consider the current task as a classification problem, where the classified object is a collection of tweets.

There is a given dataset $\mathfrak{D} = (x_i, y_i)$:

$$x_i = \{x_i^1, \ldots, x_i^m\}, \quad x_i^j \in \mathbb{W}^+, \quad j \in \{1, \ldots, m\}, \quad y_i \in \{0, 1\},$$

where $\mathbb{W}^+$ corresponds to all the possible strings written in the given language. The label $y_i = 1$ corresponds to users that are likely to spread hate speech, $y_i = 0$ corresponds to ordinary users.

Formally, the task is to find the binary classifier that minimizes an empirical risk on the dataset $\mathfrak{D}$:

$$f = \arg\min_{f \in \mathfrak{F}} \sum_{x_i, y_i \in \mathfrak{D}} [f(x_i) \neq y_i],$$

where $\mathfrak{F}$ is a set of all considered classification models.

For the proposed problem solution we employ a deep learning-based approach: each tweet is vectorized using a deep learning model. After that, we aggregate the obtained vectors and use the resulting averaged vector as a feature set for the classifier.

The proposed method is illustrated in Figure 1. The method consists of 3 main components:

Table 1
Overall results for the experiments.

English part of the dataset		
Model	**Cross-validation accuracy**	**Test accuracy**
Character n-grams, $d = 20$, $n = 3$	0.68	-
BERT-multilingual, SVM (kernel — RBF)	0.69	0.58
BERT-multilingual, Logistic regression	0.64	-
LaBSE, SVM (kernel — RBF)	0.63	-
BERT-base, SVM (kernel — RBF)	0.70	-
BERT-base, Logistic regression	0.68	-
BERT-base with fine-tuning, SVM (kernel — RBF)	**0.72**	**0.67**
BERT-base with fine-tuning, Logistic regression	0.68	-
Spanish part of the dataset		
Model	**Cross-validation accuracy**	**Test accuracy**
Character n-grams, $d = 50$, $n = 6$	0.79	-
BERT-multilingual, SVM (kernel — RBF)	**0.80**	**0.83**
BERT-multilingual, Logistic regression	0.79	-
LaBSE, SVM (kernel — RBF)	0.8	-

1. BERT model that extracts embedding vector $\mathbf{e}_i^j$ from every tweet x_i^j.
2. Aggregation operation that averages the information across all the tweets of the author:

$$\mathbf{e}_i^j = \frac{1}{m} \sum_{j=1}^{m} \mathbf{e}_i.$$

3. The resulting classifier that uses averaged tweet vector $\mathbf{e}_i$ as an input feature set.

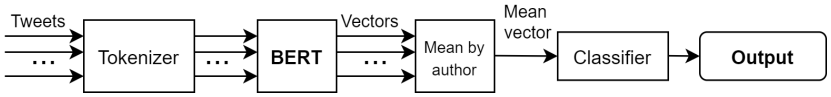

Figure 1: The scheme of the proposed approach.

3. Experiment Details

To validate our models we conducted a computational experiment. Since we use a standard BERT tokenizer, we did not use any special preprocessing for our method. For all the analyzed models we used k-fold cross-validation with $k = 10$. The results are presented in Table 1. The test accuracy was evaluated using TIRA environment [14].

Inspired by the author profiling task from the previous year [8] we used a char n-gram based model as a baseline. We used principal component analysis [15] with component number d to reduce the feature set and make the n-gram feature space denser. The n-gram order n and the component number d was selected using cross-validation.

Since the dataset contains texts in two languages, we initially decided to use a pretrained multilingual BERT model [11] for the vectorization. We analyzed its performance with two classifiers: support vector machine [16] and logistic regression. For each classification algorithm, the optimal parameters for cross-validation were selected using GridSearch. Based on the results in Table 1, it is clear that this approach works well for Spanish, but shows poor performance on English data. We decided to use Multilingual BERT for the processing of Spanish and use a different model for English. In order to improve our quality in the English part of the dataset we analyzed the following models:

- LaBSE [17], a multilingual model with BERT architecture. The main feature of this model is that all the languages supported by LaBSE share a common hidden space that can potentially improve our model generalization.
- BERT-Base, the pretrained model that was trained only on the English texts;
- BERT-base tuned on the external corpus of the tweets. The main idea of this approach is to fine-tune the lexical and semantic properties of the model on the tweet corpus in order to make the model more receptive to the specific Twitter lexicon.

Note that since the training dataset has rather small information for BERT model tuning, we did not consider the idea of fine-tuning straightforwardly on the current training dataset. Therefore we only tuned the model on the external tweet collection in an unsupervised manner it was originally trained. Some tokens in the sentences were masked, and the model tried to predict them. The predicted tokens were fed into a softmax layer to get the output words. Such procedure does not require additional labeling of external data and allows BERT to focus more on the Twitter posts than the English and Spanish texts in general.

LaBSE vectorization. Although the LaBSE model shares the common hidden space for all the languages including Spanish and English, we did not find any quality improvement in comparison to the basic architecture and other methods.

BERT-base. The pretrained BERT-base model was trained only on English texts, so we suggest that using it as a vectorizer for English tweets might improve accuracy. Our assumption that BERT-base single-language model shows better performance than the multilingual model on an English-only task is confirmed by cross-validation results. When using this model, we did not make any changes to the architecture.

BERT-based model fine-tuned on the external tweet corpus. Given that BERT model base was trained on a large collection of English texts from Wikipedia , we assume that it needs fine-tuning on the data related to our problem, since tweets often use specific slang, syntax, emoji etc. In order to improve the quality of vector representation we took a large corpus of tweets [12], which contains 1,600,000 tweets in English, and fine-tuned the model on this dataset in masked language model task regime. For all 1,600,000 tweets from the additional dataset, we replaced links, hashtags and user mentions with tokens from the training dataset (#URL#, #HASHTAG#, #USER# respectively). We also added these tokens to the BERT tokenizer dictionary. The use of additional training improved the accuracy.

As we can see, the resulting model shows a competitive performance in comparison to the baseline model. The cross-validation accuracy has increased from 0.68 to 0.72 on English part of the dataset and from 0.79 to 0.8 on its Spanish part. We got a rather significant improvement on English data by tuning the model in an unsupervised manner on the external corpus of tweets. We also observe a slight increase in performance on Spanish tweets.

The submitted English model achieves 67% in terms of accuracy on the test set, while the Spanish model shows the accuracy 83%.

For further analysis, we made a t-SNE transformation [18] for each averaged user vector $\mathbf{e}_i$ in two dimensional space and visualized them according to their class. The result is shown in Figure 2,3 for English and Spanish, respectively.

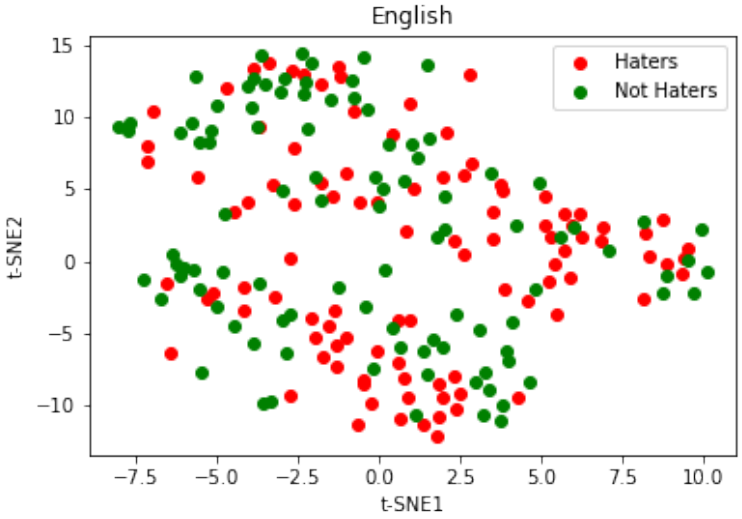

Figure 2: The t-SNE projection for the English part of the dataset.

As we can see, the obtained projection poorly separates averaged user vectors, which indicates that this method needs to be potentially improved.

We also built an LDA topic model [19] for the English part of the dataset in order to analyze the topic distributions across the tweets. For the topic model, we used 10 topics. The resulting t-SNE projection is shown in Figure 3. We analyzed the words mostly corresponding to each topic. Although we could not find a total interpretability of the obtained topics we noticed that some of them correspond to the US politics (“Biden”, “Trump”, “President”, “Wall”, “American” in the top words of the topic), US elections (“Biden”, “Trump”, “Election”, “Voters”, “Democrats”, “Republicans”, “Court” in the top words of the topic), COVID-19 (“Covid”, “19”, “Virus” in the top words of the topic). This makes us believe that clustering tweets using some topic modeling approach and employing information about topic distribution across the user tweets for the decision about hate speech spreading.

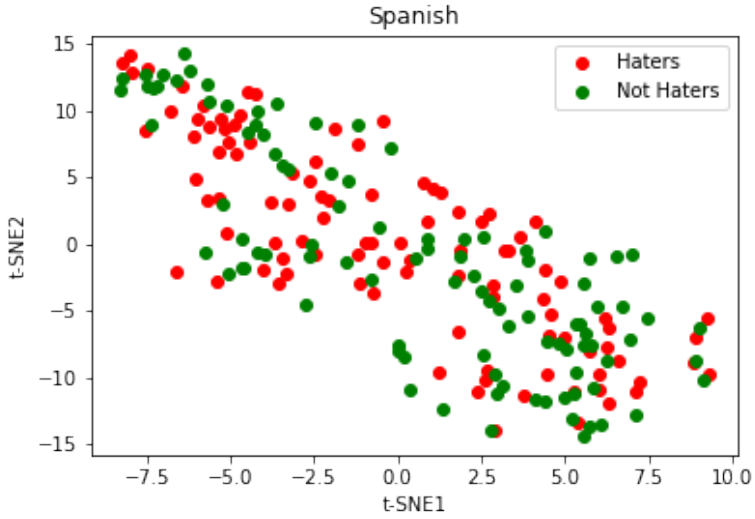

Figure 3: The t-SNE projection for the Spanish part of the dataset.

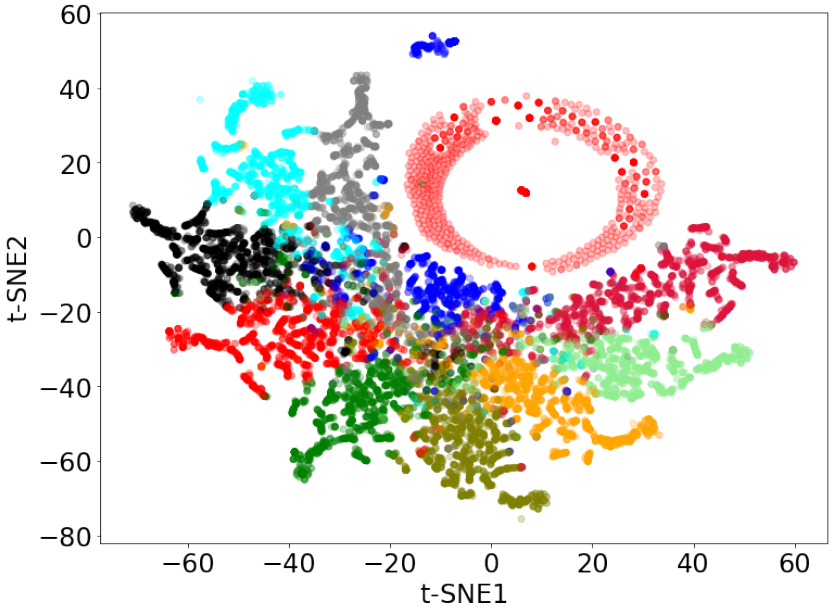

Figure 4: The t-SNE projection for topics obtained from the English part of the dataset. The light-green points correspond to the tweets about US politics, the black points correspond to the tweets about the US presidential elections, the orange points correspond to the tweets about COVID-19.

4. Conclusion

The paper describes an approach to the problem of hate speech spreaders detection. We proposed a model to detect hate speech spreaders using contextualized embeddings of single tweets. We aggregate the obtained vectors and use their average vector as a feature set for the further classification. We also provide an analysis of different vectorization models based on the BERT architecture. The resulting model shows an accuracy of about 67% for the English test dataset and 83% for the Spanish test dataset. The future work includes a detailed analysis of topic distribution across the tweets and usage of this information in the final decision rule.

Acknowledgments

This research is funded by RFBR, grant 19-29-14100.

References

[1] Z. Zhang, L. Luo, Hate speech detection: A solved problem? the challenging case of long tail on twitter, Semantic Web 10 (2019) 925–945.

[2] P. Burnap, M. L. Williams, Cyber hate speech on twitter: An application of machine classification and statistical modeling for policy and decision making, Policy & internet 7 (2015) 223–242.

[3] T. Davidson, D. Warmsley, M. Macy, I. Weber, Automated hate speech detection and the problem of offensive language, in: Proceedings of the International AAAI Conference on Web and Social Media, volume 11, 2017.

[4] B. Gambäck, U. K. Sikdar, Using convolutional neural networks to classify hate-speech, in: Proceedings of the first workshop on abusive language online, 2017, pp. 85–90.

[5] K. Miok, B. Škrlj, D. Zaharie, M. Robnik-Šikonja, To ban or not to ban: Bayesian attention networks for reliable hate speech detection, Cognitive Computation (2021) 1–19.

[6] J. Bevendorff, B. Chulvi, G. L. D. L. P. Sarracén, M. Kestemont, E. Manjavacas, I. Markov, M. Mayerl, M. Potthast, F. Rangel, P. Rosso, E. Stamatatos, B. Stein, M. Wiegmann, M. Wolska, , E. Zangerle, Overview of PAN 2021: Authorship Verification,Profiling Hate Speech Spreaders on Twitter,and Style Change Detection, in: 12th International Conference of the CLEF Association (CLEF 2021), Springer, 2021.

[7] F. Rangel, G. L. D. L. P. Sarracén, B. Chulvi, E. Fersini, P. Rosso, Profiling Hate Speech Spreaders on Twitter Task at PAN 2021, in: CLEF 2021 Labs and Workshops, Notebook Papers, CEUR-WS.org, 2021.

[8] F. Rangel, A. Giachanou, B. Ghanem, P. Rosso, Overview of the 8th author profiling task at pan 2020: Profiling fake news spreaders on twitter, in: CLEF, 2020.

[9] F. E. Ayo, O. Folorunso, F. T. Ibharalu, I. A. Osinuga, Hate speech detection in twitter using hybrid embeddings and improved cuckoo search-based neural networks, International Journal of Intelligent Computing and Cybernetics (2020).

[10] H. Faris., I. Aljarah., M. Habib., P. Castillo., Hate speech detection using word embedding and deep learning in the arabic language context, in: Proceedings of the 9th Interna-

tional Conference on Pattern Recognition Applications and Methods - ICPRAM„ INSTICC, SciTePress, 2020, pp. 453–460. doi:10.5220/0008954004530460.
[11] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).
[12] A. Go, R. Bhayani, L. Huang, Twitter sentiment classification using distant supervision, CS224N project report, Stanford 1 (2009) 2009.
[13] O. Bakhteev, A. Ogaltsov, P. Ostroukhov, Fake news spreader detection using neural tweet aggregation, in: CLEF, 2020.
[14] M. Potthast, T. Gollub, M. Wiegmann, B. Stein, TIRA Integrated Research Architecture, in: N. Ferro, C. Peters (Eds.), Information Retrieval Evaluation in a Changing World, The Information Retrieval Series, Springer, Berlin Heidelberg New York, 2019. doi:10.1007/978-3-030-22948-1_5.
[15] C. M. Bishop, Pattern recognition and machine learning, springer, 2006.
[16] C. Cortes, V. Vapnik, Support vector machine, Machine learning 20 (1995) 273–297.
[17] F. Feng, Y. Yang, D. Cer, N. Arivazhagan, W. Wang, Language-agnostic bert sentence embedding, arXiv preprint arXiv:2007.01852 (2020).
[18] L. Van der Maaten, G. Hinton, Visualizing data using t-sne., Journal of machine learning research 9 (2008).
[19] K. Vorontsov, O. Frei, M. Apishev, P. Romov, M. Dudarenko, Bigartm: Open source library for regularized multimodal topic modeling of large collections, in: International Conference on Analysis of Images, Social Networks and Texts, Springer, 2015, pp. 370–381.