Profiling Spreaders of Hate Speech with N-grams and RoBERTa

Notebook for PAN at CLEF 2021

Christopher Bagdon

Eberhard Karls Universität Tübingen, Fachschaft Sprachwissenschaft Wilhelmstr. 19 72074 Tübingen, Germany

Abstract

This paper outlines our approach to the 2021 CLEF Conference Shared Task, Profiling Hate Speech Spreaders on Twitter. Our approach uses the probability output of a logistic regression classifier and a RoBERTa based classifier as features for a linear support vector classifier. During a final cross validation analysis the Spanish meta-classifier performed better than any other single classifier. For English the meta-classifier performed slightly worse than the RoBERTa classifier. On the test set our system performed moderately well in comparison to other submissions, with 81% accuracy for Spanish and 67% for English. Overall our system placed 15th of 66 entries.

Keywords

N-grams, RoBERTa, SVM, TF-IDF, Transformer-model

1. Introduction

Social media has taken a firm place in people's lives around the world. The number of people which use social media continues to grow year over year. While this has many possible benefits, such as allowing people to express themselves and connect with others, it also comes with drawbacks, such as the proliferation of hate speech. The combination of anonymity, echo chambers, and ease of access helps to circulate hate speech on different platforms [1]. These platforms have a need to be able to automatically detect hate speech and profile its spreaders.

This paper details our submission to the 2021 PAN Author Profiling Shared Task, Profiling Hate Speech Spreaders on Twitter. The task is to classify Twitter users as a spreader of hate speech or not, given a sample of 200 tweets per user. In previous Author Profiling shared tasks Support Vector Machines (SVM) and n-grams have proven very successful across different tasks, while transformer based approaches have only seen moderate success [2], despite showing strong results in direct fake news [3] and hate speech detection [4].

Our approach attempts to combine the results from a n-gram-based logistic regression classifier with a transformer model based on RoBERTa via a SVM meta-classifier. The paper is structured as follows: Section 2 reviews related research and works. Section 3 dives into our approach by going through the preprocessing, the logistic regression and RoBERTa models, and

CLEF 2021 – Conference and Labs of the Evaluation Forum, September 21–24, 2021, Bucharest, Romania

Christopher.Bagdon@student.uni-tuebingen.de (C. Bagdon)
0000-0002-0877-7063 (C. Bagdon)

CEUR Workshop Proceedings (CEUR-WS.org)

finally the meta-classifier. Section 4 covers the results from training and of the test set. Finally section 6 shares our conclusions.

2. Related Work

Hate speech detection has been a popular topic among Natural Language Processing researchers in recent years. Academic events such as IberEval 2018 [5] and SemEval-19 [6], to name a couple, show how strong interest in the topic is. The tasks from these events provide insight on successful approaches and common challenges when detecting hate speech. In the IberEval 2018 task, Automatic Misogyny Identification, the most successful approach used an SVM with combinations of stylistic, structural and lexical features, while other strong approaches used SVMs with n-gram features. Deep learning approaches were not as successful [5]. The majority of submissions to SemEval-19's task, Multilingual Detection of Hate Speech Against Immigrants and Women in Twitter, used some form of Deep Learning Model, including Recurrent Neural Networks (RNN) and large language models. Despite this, the highest performing systems across both sub tasks for English and Spanish datasets employed more traditional machine learning models, mostly SVMs [6]. Interestingly, the top three systems labeled the same 14.6% of texts incorrectly on the Spanish dataset and 19.1% of texts for English [6]. This could be caused by a common difficulty in identifying hate speech; when slurs or words commonly associated with hate speech are used in a humorous context [6], or by targeted communities reclaiming the words used to target them [7]. Machine learning systems often lack the context to determine if these words are being used in a manner that constitutes hate speech [8]. There are numerous other possible pitfalls for machine learning to fall into due to lack of contextual understanding, such as authors quoting historical texts or referring to specific instances of hate speech [9]. While a tweet or text might contain hate speech, that does not guarantee that the text as a whole is hate speech.

In the 2020 PAN shared task, Profiling Fake News Spreaders on Twitter, there was a variety of methods used for classification, preprocessing, and feature selection [2]. The best performing approaches used word and/or character n-grams with SVM and/or Logistic Regression classifiers. This saw upwards of 0.75 and 0.82 accuracy scores on English and Spanish data sets respectively. These approaches were also effective in the 2019 task, seeing results as high as 0.95 accuracy on bot detection and 0.82 accuracy on gender profiling [10]. The top three approaches from 2020 directly showed to be effective on this year's task with only small losses in performance, though without any tuning [11].

Recently NLP tasks have seen success using transformer based large language models and transfer learning [12][13]. Researchers have been successful in using models such as Google's BERT in classification tasks such as detecting hate speech [4], fake news detection [3], and authorship attribution [14]. Over the last couple years variations of of these models have been made available, such as RoBERTa [13] and DistilBERT[15], which have shown improvements in both performance and accessibility. As researchers continue to pour resources into building larger language models, their ability to perform down stream tasks via transfer learning continues to grow [16].

3. Methodology

The system is composed of three major parts. After preprocessing, the data it is sent to a logistic regression classifier and to a RoBERTa classifier separately. The probability outputs from each are then used as features for the final meta classifier, a linear SVM.

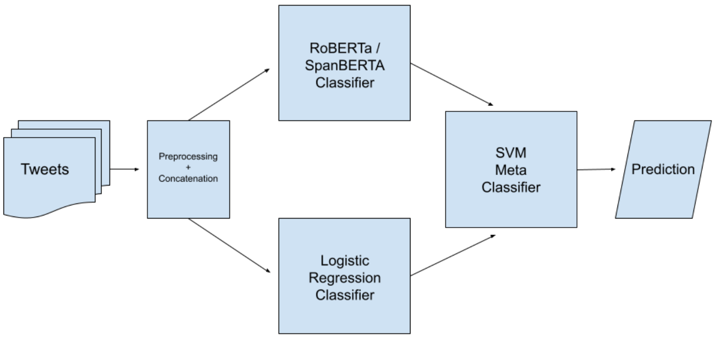

Figure 1: System architecture

3.1. Preprocessing

First each author's tweets are concatenated into a single string per author, as this was found to be more effective than classifying each tweet separately in previous work[17]. Then the text is set to lowercase and repeated characters are removed. For data going to the RoBERTa models, emojis are replaced with #EMOJI# (following the dataset's format for replacement tokens).

3.2. SVM and Logistic Regression

To serve as a baseline, a Linear SVM is used, based on the success found in previous shared tasks [11][18]. The classifier takes two features; a Term Frequency-Inverse Document Frequency (TF-IDF) sparse matrix of character n-grams and a TF-IDF for word n-grams. The word TF-IDF uses a range of (1, 2) n-grams, limited to a minimum of 0.05 frequency and a max of 0.85. The character TF-IDF uses a range of (1, 6) n-grams with 0.001 minimum and no maximum. The model was optimized with a repeated stratified K-fold grid search, using 10 splits repeated 3 times.

In order to provide probabilities rather than predictions for the meta-classifier, a logistic regression classifier is used in place of the SVM. It was optimized with the same methods as the SVM. The hyperparameters for both can be seen in Table 1.

3.3. RoBERTa

The RoBERTa models are built using the Simple Transformers[1] library. The English model is built with the Roberta-base pretrained model and the Spanish model with the Spanberta-base-

[1] https://simpletransformers.ai

Table 1
Hyperparameters found via Grid Search

Model	Language	C	Tol	Class Weight	Intercept Scaling	Loss
SVM (baseline)	EN	22000	0.1	Balanced	0.877	Hinge
Logistic Regression	EN	100000	1e-05	Balanced	0.1	–
Meta-Classifier SVM	EN	0.015	0.5	None	5	Hinge
SVM (baseline)	ES	22000	0.1	Balanced	0.01	Hinge
Logistic Regression	ES	100000	1.53e-04	Balanced	0.1	–
Meta-Classifier SVM	ES	1	5	Balanced	5	Squared Hinge

Table 2
Hyperparameters for RoBERTa and SpanBERTa models.

Model	Language	LR	Epochs	Batch Size	Eval-Batch Size	Weight Decay	Special Tokens
RoBERTa	EN	2.84E-05	3	8	4	0.1	[#EMOJI#, #HASHTAG#, #USER#, #URL#]
SpanBERTa	ES	2.86E-05	1	8	4	0.1	[#EMOJI#, #HASHTAG#, #USER#, #URL#]

cased[2] pretrained model. The models consist of 12 hidden layers, 12 attention heads, a single dense layer classifier, and uses Adam optimizer. Hyper parameters were found using the Sweeps function of Wandb[3] and can be seen in Table 2. Each model was trained on an 80/20 split of their respective data-set.

The max token length is set to 128, due to a lack of computational power, so a sliding window is used to break up the long concatenated strings. Each window uses a 20% overlap. The final classification output is a list containing a probability for each class for each window, per author. Each list is reduced to a single probability per class by taking the median value of all windows. Summing the values and averaging the values was also tested, but the median values showed marginally better results.

3.4. Meta-Classifier

A Linear SVM is used as the meta-classifier. Four features are used as input; one probability from the RoBERTA classifier and from the logistic regression classifier each, per class. The meta-classifier was trained using the same 80/20 training splits as the RoBERTa models. Hyper parameter optimization was done using grid searches and the chosen parameters can be found in Table 1.

4. Results

To analyze the effectiveness of the system each working part was put through a 10 fold cross validation using only the training set. The Spanish model performed far better than its English counterpart. The logistic regression and SpanBERTa models each performed better than the baseline SVM, and the SVM meta-classifier out performed all of them. Unfortunately the English

[2] https://skimai.com/roberta-language-model-for-spanish

[3] https://docs.wandb.ai/guides/sweeps

Table 3
Cross Validation Results

Model	Language	Accuracy
SVM (baseline)	EN	0.66
Logistic Regression	EN	0.640
RoBERTa	EN	0.695
Meta-Classifier	EN	0.682
SVM (baseline)	ES	0.796
Logistic Regression	ES	0.825
SpanBERTa	ES	0.817
Meta-Classifier	ES	0.845

Table 4
Final test set results

Model	Language	Accuracy
Meta-Classifer	EN	67%
	ES	81%

models did not fare as well. Logistic regression saw a slight loss compared to the baseline. The RoBERTa model was the strongest, scoring a point higher than the meta-classifier, but failing to break into 0.70 accuracy.

On the test set the system performed very well compared to the training set. The meta-classifier only lost 1% accuracy on the English dataset and still outperformed the SVM and logistic regression parts. For Spanish the meta-classifier lost 3% but maintained a better score than the SVM and roughly the same score as the SpanBERTa model.

Our system ranked 15$^{\text{th}}$ of 66 submissions. It did especially well on the Spanish dataset, coming in just 4% below the top ranked submission. English was not far behind, scoring 7% under the leading system.

Overall it seems there could be a benefit to combining the results of a transformer based model and simpler models such as logistic regression. In the future it will be interesting to try this again with different datasets and different transformer models, such as OpenAI's ginormous GPT-3.

References

[1] M. Mondal, L. A. Silva, F. Benevenuto, A measurement study of hate speech in social media, in: Proceedings of the 28th ACM Conference on Hypertext and Social Media, HT '17, Association for Computing Machinery, New York, NY, USA, 2017, p. 85–94. URL: https://doi.org/10.1145/3078714.3078723. doi:10.1145/3078714.3078723.

[2] 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.

[3] H. Jwa, D. Oh, K. Park, J. M. Kang, H. Lim, exbake: Automatic fake news detection model

based on bidirectional encoder representations from transformers (bert), Applied Sciences 9 (2019). URL: https://www.mdpi.com/2076-3417/9/19/4062. doi:10.3390/app9194062.

[4] M. Mozafari, R. Farahbakhsh, N. Crespi, A bert-based transfer learning approach for hate speech detection in online social media, CoRR abs/1910.12574 (2019). URL: http://arxiv.org/abs/1910.12574. arXiv:1910.12574.

[5] E. Fersini, P. Rosso, M. Anzovino, Overview of the task on automatic misogyny identification at ibereval 2018, in: IberEval@SEPLN, 2018.

[6] V. Basile, C. Bosco, E. Fersini, D. Nozza, V. Patti, F. M. Rangel Pardo, P. Rosso, M. Sanguinetti, SemEval-2019 task 5: Multilingual detection of hate speech against immigrants and women in Twitter, in: Proceedings of the 13th International Workshop on Semantic Evaluation, Association for Computational Linguistics, Minneapolis, Minnesota, USA, 2019, pp. 54–63. URL: https://www.aclweb.org/anthology/S19-2007. doi:10.18653/v1/S19-2007.

[7] C. Bianchi, Slurs and appropriation: An echoic account, Journal of Pragmatics 66 (2014) 35–44. doi:https://doi.org/10.1016/j.pragma.2014.02.009.

[8] T. Davidson, D. Warmsley, M. W. Macy, I. Weber, Automated hate speech detection and the problem of offensive language, CoRR abs/1703.04009 (2017). URL: http://arxiv.org/abs/1703.04009. arXiv:1703.04009.

[9] S. MacAvaney, H.-R. Yao, E. Yang, K. Russell, N. Goharian, O. Frieder, Hate speech detection: Challenges and solutions, PLOS ONE 14 (2019) 1–16. doi:10.1371/journal.pone.0221152.

[10] F. Rangel, P. Rosso, Overview of the 7th Author Profiling Task at PAN 2019: Bots and Gender Profiling, in: L. Cappellato, N. Ferro, D. Losada, H. Müller (Eds.), CLEF 2019 Labs and Workshops, Notebook Papers, CEUR-WS.org, 2019. URL: http://ceur-ws.org/Vol-2380/.

[11] C. Bagdon, S. Grässel, Examining hate speech spreaders and fake news spreaders through pan shared tasks (2021). URL: https://www.researchgate.net/publication/351881197_Examining_Hate_Speech_Spreaders_\and_Fake_News_Spreaders_Through_PAN_Shared_Tasks?channel=doi&linkId=60ae7e01a\6fdcc647ede8894&showFulltext=true. doi:10.13140/RG.2.2.12308.22404.

[12] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, CoRR abs/1810.04805 (2018). URL: http://arxiv.org/abs/1810.04805. arXiv:1810.04805.

[13] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized BERT pretraining approach, CoRR abs/1907.11692 (2019). URL: http://arxiv.org/abs/1907.11692. arXiv:1907.11692.

[14] M. Fabien, E. VILLATORO-O, P. Motlicek, S. Parida, Bertaa: Bert fine-tuning for authorship attribution, Proceedings of the 17th International Conference on Natural Language Processing (2020). URL: http://infoscience.epfl.ch/record/285045.

[15] V. Sanh, L. Debut, J. Chaumond, T. Wolf, Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter, CoRR abs/1910.01108 (2019). URL: http://arxiv.org/abs/1910.01108. arXiv:1910.01108.

[16] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei,

Language models are few-shot learners, CoRR abs/2005.14165 (2020). URL: https://arxiv.org/abs/2005.14165. arXiv:2005.14165.

[17] A. Baruah, K. Das, F. Barbhuiya, K. Dey, Automatic Detection of Fake News Spreaders Using BERT—Notebook for PAN at CLEF 2020, in: L. Cappellato, C. Eickhoff, N. Ferro, A. Névéol (Eds.), CLEF 2020 Labs and Workshops, Notebook Papers, CEUR-WS.org, 2020. URL: http://ceur-ws.org/Vol-2696/.

[18] J. Pizarro, Using N-grams to detect Fake News Spreaders on Twitter—Notebook for PAN at CLEF 2020, in: L. Cappellato, C. Eickhoff, N. Ferro, A. Névéol (Eds.), CLEF 2020 Labs and Workshops, Notebook Papers, CEUR-WS.org, 2020. URL: http://ceur-ws.org/Vol-2696/.