

Profiling Hate Speech Spreaders on Twitter: Exploiting Textual Analysis of Tweets and Combinations of Multiple Textual Representations

Claudio Moisés Valiense de Andrade¹, Marcos André Gonçalves¹

¹Federal University of Minas Gerais
Belo Horizonte, Minas Gerais, Brazil

Abstract

In this paper, we describe our solution for the task “Profiling Hate Speech Spreaders on Twitter” promoted by PAN CLEF 2021. The task is to identify user profiles that promote hate speech on social media Twitter. Data for 200 users has been made available – each user has a set of 200 posts and the corresponding label (hate speech propagator or not). Our solution consists of exploiting several text representations and classifiers available in the literature. Based on the predictions of the classifiers estimated with nested cross-validation in the training set, we reach a consensus between the best results to obtain a final prediction. Our solution reached an accuracy of 63.0% for English language and 79% for Spanish. To guarantee the reproducibility of our solution, all the documentation and code is available on github ¹.

Keywords

Author Profiling, Hate Speech, Supervised Algorithm

1. Introduction

In this paper we describe our participation in PAN @ CLEF 2021 [1]. This edition looks for solutions to identify profiles of hate propagators [2], where hate is defined as any communication that depreciates a person or a group. From the identification of the profiles, it is possible to avoid the spread of hate speech, keeping the social network healthier.

We treat this problem as a *binary text classification task*. In this context, a user profile is represented by the set of textual posts she has issued in the social network altogether. For the sake of classification, such textual data can be represented in several ways (e.g. TF-IDF, word embeddings, text graph), each representation capturing or focusing on a different aspect of the classification task. For instance, the traditional TF-IDF representation captures statistical aspects of the text and the specificity of certain words in the collection (IDF component). Word embeddings are vectorial representations of words, sentences and whole documents aimed at capturing word co-occurrence patterns and contextual information.

Our solution is simple yet very effective. We rely on a majority voting of several classifiers exploiting these different representation based on the assumption that those combinations


¹https://github.com/claudiovaliense/hate_speech_twitter

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

✉ claudio.valiense@dcc.ufmg.br (C. M. V. d. Andrade); mgoncalv@dcc.ufmg.br (M. A. Gonçalves)



© 2021 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

(representation/classifier) are complementary and can be effectively combined for improving the classification task. We exploit 4 representations (word tfidf, char tfidf, vader, and roberta's word embeddings) along with two classifiers (svm and random forests), which are still very strong text classifiers, even when compared to recent deep neural networks, including recent Transformer architectures, as shown in recent work [3, 4]. Lastly, we rely on a majority voting (most frequent decision) for the final outcome.

Our experimental results show that solution can reach an accuracy of 63% for English language and 79% for Spanish, where the majority voting is always better or at least tied to the best single combination of (representation/classifier). This best combination, in both cases, English and Spanish, was obtained when using the char_tfidf representation along with Random Forests, showing the importance of dealing with noise in the data, especially social network textual data.

2. Related Work

BERT proposed by Devlin et al. [5] is a framework that exploits several techniques existing in the literature. Among those techniques we highlight: 1) dynamic embeddings, where the word embedding is adjusted according to other words found in the sentence (that is, it seeks to capture a word's context); 2) term masking, consisting of a percentage of words that are exchanged for a fixed sequence of words aiming at finding out what the original terms are from these masked terms – the goal is to learn relationships between words; and 3) a Multi-head attention mechanism. In our solution we will use RoBERTa, a technique that uses BERT as a base but proposes several improvements, for example, using a longer training phase exploiting longer word sequences.

Several works [6, 7] propose stacking classifiers in order to assess the final impact. Different from traditional stacking, this article combines representations in order to benefit from the complementary information containing in each one of them.

Our work is more aligned with [8], which also proposed to combine representations for the sake of citation classification. Differently from that work, in which the representations are simply concatenated before serving as input for a classifier, here we exploit a different combination rule, based on a majority voting of outputs of the application of different representations to different classifiers.

3. Data Representation Methodology

In this section we describe the methodology used to represent the data in our solution. Subsection 3.1 presents how we treat the dataset while subsections 3.2, 3.3, 3.4 and 3.5 describe the specific representations uses to represent the textual data, aimed at capturing different aspects of the task.

3.1. Dataset Representation

A dataset of 200 users was made available. For each user, there are 200 posts issued on the social network Twitter. We concatenated all these 200 messages together into a single document to represent a user profile. For each user, their label was made available (hate spreader or not). The data presented in Table 1 refers to the training and validation sets used in our solution.

Table 1 presents the two datasets made available for PAN@2021, one in English and another in Spanish. The respective columns describe: amount of documents ($|D|$), average amount of words per document ($Avg(T)$), amount of unique words in the collection ($U(T)$), average number of times a word appears in the dataset's documents (*aka*, rarity) ($O(T)$), number of classes ($|C|$) and number of documents in the largest ($Max(C)$) and smallest class ($Min(C)$).

As it can be seen in Table 1, the data is balanced as the two classes have the same number of documents. Spanish has a larger vocabulary and the Spanish words appears in average in less documents, potentially making them more discriminative. This may be one of the reasons that classification results in Spanish were a bit better than in English, as we shall see.

Table 1
Dataset statistics.

Name	$ D $	$Avg(T)$	$U(T)$	$O(T)$	$ C $	$Max(C)$	$Min(C)$
Pan21 English	200	2454.8	57556	8.4	2	100	100
Pan21 Spanish	200	2587.7	75038	6.9	2	100	100

3.2. Word TF-IDF

The TF-IDF representation [9] consists of two parts, the *tf* that quantifies the frequency of terms, and the *idf*, the inverse document frequency, that gives higher weight to words that occur less frequently in documents. The final score of a word *t* is obtained by $tf * idf$, which gives more importance to rare words considering the set of documents.

3.3. Char TF-IDF

We made a modification in *tf-idf*, which we call *char tf-idf*, by defining sequences of characters to be analyzed. Those sequences become a potential new "word" present in the vocabulary. This modification aims to deal with typos, misspellings and other types of grammar and syntactical mistakes commonly found in social media text. For instance, "love" and "lovee" are considered the same word based on a 4-character string. We limit the creation of a sequence from 2 to 6 characters.

3.4. Vader

The Vader representation [10] is based on a lexicon dictionary where each word has three polarity scores: positive (0-1), neutral (0-1) or negative (0-1). A document is represented as a 3-dimensional vector, each vector position representing the proportion of words of each type of polarity, where the higher the value, the more the word is related to polarity.

3.5. Word Embedding

Word embeddings are vector representations of words, phrases and entire documents aimed at capturing patterns of co-occurrence of words and contextual information. In this work, we exploited RoBERTa [11] embeddings, a pre-trained learning algorithm to obtain vector representations for words. In RoBERTa's representation, each word has a 12-layer representation, where each layer has 768 dimensions. To represent a word t , we concatenate the last 4 layers, resulting in a vector of 3072 positions. To represent a document d , we average the vectors of all words in the document. The document's representation is the input for the classification algorithms.

4. Combining Representations and Classifiers – Majority Voting

After creating the representations for a text, we used them as input for each classifier. Each representation and classifier pair generates a prediction for a document d . Given n predictions for a document d , the “frequency” algorithm selects the predictions of the 4 models that obtained the best result in the validation, verifies the most frequent one and sets it as the final prediction for the document.

5. Experiment

5.1. Tuning Hyperparameter

After creating the dataset representation used as input to a classifier, training and actual classification of documents is carried out. For each dataset we apply a nested cross validation procedure as suggested by Varma and Simon [12], using a 5x3 setting (5-fold cross-validation in the outer loop and 3-fold cross-validation in the inner loop). The internal cross validation is responsible for finding the best parameters for a given fold, used to build the model utilized to make predictions in the test set. We evaluated the parameters of the following classifiers:

- Linear SVM, we varied the C parameter considering $[10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1, 10^2, 10^3, 10^4]$
- Random Forest, we evaluated the n estimator parameter considering the following values: $[10, 50, 100, 200, 500, 1000, 2000]$

5.2. Results

We report in Table 2 the average accuracy results of the 5 test folds along with confidence intervals (with 95% confidence level) for both, the English and Spanish datasets, while the last line is the result of the test set submitted through the TIRA [13] system. Each line in the Table corresponds to a combination (representation, classifier) in a given language. For instance, line 1 of Table 2 presents the accuracy results of using the roberta representation as input for the SVM classifier for the English dataset. The penultimate line of the table for both, English and Spanish represent our Majority Voting heuristics results. And finally, the last line corresponds

to the results in the Test set made available by the CLEF organizers for which we did not have labels at development time¹.

Table 2

Result of average accuracy in validation data and 95% confidence interval (IC).

Name	Accuracy \pm IC
pan21 English roberta_concat_classifier_svm	47.50 \pm 22.17
pan21 English vader_classifier_random_forest	54.00 \pm 8.39
pan21 English word_tfidf_classifier_random_forest	57.50 \pm 9.57
pan21 English vader_classifier_svm	58.00 \pm 16.04
pan21 English word_tfidf_classifier_svm	62.50 \pm 7.60
pan21 English roberta_concat_classifier_random_forest	63.50 \pm 11.74
pan21 English char_tfidf_classifier_svm	65.50 \pm 8.89
pan21 English char_tfidf_classifier_random_forest	67.50 \pm 12.22
Majority voting English	69.00 \pm 10.43
Test Set	63.00
pan21 Spanish vader_classifier_svm	42.50 \pm 8.78
pan21 Spanish vader_classifier_random_forest	58.50 \pm 6.44
pan21 Spanish word_tfidf_bigram_classifier_svm	66.00 \pm 20.31
pan21 Spanish roberta_concat_classifier_svm	69.50 \pm 13.78
pan21 Spanish roberta_concat_classifier_random_forest	72.00 \pm 5.55
pan21 Spanish char_tfidf_classifier_svm	72.50 \pm 6.21
pan21 Spanish word_tfidf_classifier_random_forest	75.00 \pm 3.80
pan21 Spanish char_tfidf_classifier_random_forest	76.50 \pm 9.21
Majority voting Spanish	76.00 \pm 7.48
Test Set	79.00

As it can be seen, our experimental results show that our solution can reach an accuracy of 63% for English language and 79% for Spanish. The majority voting is always better or at least tied to the best single combination of (representation/classifier). In both cases, English and Spanish, the best individual results were produced by combining the char_tfidf representation with Random Forests.

The best results of the char_tfidf in both languages confirm our hypotheses that it is important to cope with noise and errors when dealing with social media textual data, especially Twitter data. The better results of the majority voting provides evidence for our hypothesis of complementarity among the representations and classifiers. And finally, the slightly better results for Spanish when compared to English may be due to larger Spanish vocabulary and presence of more discriminative words, but this hypothesis requires a more thorough investigation.

6. Conclusion and Future Work

In this paper we described our participation in the task “Profiling Hate Speech Spreaders on Twitter” organized by PAN @ CLEF 2021. We use four complementary text representations along

¹The Test Set results do not contain confidence intervals as it was not part of our 5-fold cross-validation experiments.

with two strong text classifiers available in the literature. Our simple majority voting heuristics that combines the decision of pairs (representation/classifier) achieves accuracy as high as 63% for English and 79% for Spanish, outperforming or tying the best individual combinations. This provides evidence for the complementarity of the strategies. Dealing with noise in social media network is also important to boost results. And results in Spanish demonstrated to be better than in English, requiring an additional investigation for determining the exact reasons for this behavior.

As future work we intend to perform an ablation or full factorial analysis to understand which elements of our solution contributed more to the final results, include other representations and classifiers, study other ways of combining them and, finally, take advantage of attention mechanisms to reinforce cross-representation common elements (e.g., words) that are discriminative in more than one representation.

Acknowledgments

This work was partially supported by CNPq, CAPES, Fapemig, NVIDIA and Google.

References

- [1] 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.
- [2] 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.
- [3] W. Cunha, S. D. Canuto, F. Viegas, T. Salles, C. Gomes, V. Mangaravite, E. Resende, T. Rosa, M. A. Gonçalves, L. C. da Rocha, Extended pre-processing pipeline for text classification: On the role of meta-feature representations, sparsification and selective sampling, *Inf. Process. Manag.* 57 (2020) 102263.
- [4] W. Cunha, V. Mangaravite, C. Gomes, S. D. Canuto, E. Resende, C. Nascimento, F. Viegas, C. França, W. S. Martins, J. M. Almeida, T. Rosa, L. C. da Rocha, M. A. Gonçalves, On the cost-effectiveness of neural and non-neural approaches and representations for text classification: A comprehensive comparative study, *Inf. Process. Manag.* 58 (2021) 102481. doi:10.1016/j.ipm.2020.102481.
- [5] 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).
- [6] N. F. da Silva, E. R. Hruschka, E. R. Hruschka, Tweet sentiment analysis with classifier ensembles, *Decision Support Systems* 66 (2014) 170 – 179.
- [7] C. R. F. Gomes, M. A. Gonçalves, On the cost-effectiveness of stacking of neural and non-neural methods for text classification: Scenarios and performance prediction, *Association for Computational Linguistics, Bangkok, Thailand, 2021.*

- [8] C. M. V. de Andrade, M. A. Gonçalves, Combining representations for effective citation classification, in: Proceedings of the 8th International Workshop on Mining Scientific Publications, Association for Computational Linguistics, Wuhan, China, 2020, pp. 54–58.
- [9] K. S. Jones, A statistical interpretation of term specificity and its application in retrieval, *Journal of documentation* (1972).
- [10] C. Hutto, E. Gilbert, Vader: A parsimonious rule-based model for sentiment analysis of social media text, in: Proceedings of the International AAAI Conference on Web and Social Media, volume 8, 2014.
- [11] 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, *arXiv preprint arXiv:1907.11692* (2019).
- [12] S. Varma, R. Simon, Bias in error estimation when using cross-validation for model selection, *BMC bioinformatics* 7 (2006) 1–8.
- [13] 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.