Isolated Profile Style Representation

Carlos A. Rodríguez-Losada[1], Daniel Castro-Castro[2]

[1]Computer Science Department, University of Oriente "Antonio Maceo", Santiago de Cuba, Cuba
[2]Information Retrieval Lab, Computer Science Department, University of La Coruña, Spain

Abstract

This work shows the obtained results by the UO-UDC team at Profiling Irony and Stereotype Spreaders on Twitter shared task hosted by PAN22. We presented a hybrid model from BERT-like embeddings and the lexical representation of tweets. We exposed as experimental results the interactions and impact between combinations of representations in the final accuracy score. It is shown that it is not enough to represent a profile considering only independent features along with its corresponding class.

Keywords

irony profiling, stereotype, tweet's representation

1. Introduction

Currently, large flows of information are managed on the Internet. Content creation is accompanied by ethical questions regarding determining what is socially accepted. The irony is defined as a clever way of expressing an idea when in fact, another is being expressed. This, when accompanied by stereotyped ideas such as the sexual orientation of others, women's rights, or the LGBTQI+ community, can generate controversy on social networks and in some cases, hate speech [1, 2, 3].
Interesting approaches have been taken along this time by some authors and one of the most known environments this takes place happens at the PAN shared tasks[1].
Fersini et al. [4] proposed an approach based on stylometry, personality, emotions, and feed embedding to train a Support Vector Machine (SVM) classifier. Espinosa et al. [5] extracted features from text using N-grams of characters and words to train the SVM classifier as well. Duan et al. [6] address the binary classification problem to detect Fake News Spreader extracting linguistic and sentiment features from users tweet's feed using the `torch.nn`[2] library.
Carracedo et al. [7] proposed several emotion-prototypes to map user messages to an emotion space, to finally test every prototype with Naïve Bayes, K-nearest neighbors, SVM, Logistic Regression and Gradient Boosting, among others. Bagdon et al. [8] combine the results from a n-gram-based logistic regression classifier with a transformer model based on RoBERTa [9] via a SVM meta-classifier.

CLEF 2022: Conference and Labs of the Evaluation Forum, September 5–8, 2022, Bologna, Italy
carlosarl1999@gmail.com (C. A. Rodríguez-Losada); daniel.castro3@udc.es (D. Castro-Castro)
0000-0001-9102-7601 (D. Castro-Castro)

[1] https://pan.webis.de/shared-tasks.html
[2] https://pytorch.org/docs/master/generated/torch.nn.GRU.html

In the Profiling Irony and Stereotype Spreaders on Twitter (IROESTERO) competition [10] organized by PAN22 [11], the main goal is to classify Twitter profiles as irony spreaders or non-irony spreaders. The main difference concerning previous tasks is that this year's task proposes to develop systems capable of identifying Twitter profiles that spread irony and stereotypes given the history of tweets of that profile.

As usual, the task organizers provide the TIRA platform [12] to perform all the heavy computations by the competitor's models.

In our work we proposed a tweet profile representation using some techniques of deep learning and lexical analysis, due to the state-of-the-art of this model [13, 14].

The paper is structured as follows. In Section 2, we described the proposed system to competitions. In Section 3, an experimental description of developed experiments is provided. Finally, in Section 4, we explain the acquired conclusions from the work and we suggest a future investigation line related to this work.

2. Our proposal

2.1. Task's specifications

The task consists in given a Twitter user's feed written in English, composed of 200 tweets sharing Irony and Stereotyping content, and discriminating whether the given user should be labeled as an Irony and Stereotype Spreader (ISS) or not. As competition baselines, PAN22 will use Character/Words n-grams + SVM/Logistic Regression, etc.

2.2. Model overview

Given a user timeline, our model builds its representation based on a Semantic Representation (SR), Punctuation Marks Representation (PMR), and an Auxiliary Words Representation (AWR).

The first representation aims to distinguish what is spoken in a text document. To do so, we employ a fine-tuned model from sentence-transformers named r2d2/stsb-bertweet-base-v0 which maps sentences and paragraphs to a 768-dimensional dense vector space and can be used for tasks like clustering or semantic search. This model belongs to the Bidirectional Model from Transformers (BERT) models [15].

The second representation looks to capture the author's writing style based on his use of punctuation marks such as emojis, emphasis signs, and special characters.

The last representation is intended to represent the writing style relying on author discourse markers use.

2.3. Model Stages

2.3.1. Semantic, Punctuation and Auxiliary Words Representations

Figure 1 shows all the profile representations our model builds.

First of all, it is constructed a 10-vectors 768-dimensional vector to build the SR. Each of the 10-vectors holds an accumulated sum obtained by encoding 20 user tweets with r2d2/stsb-bertweet-base-v0 sentence-transformers embeddings. All these vectors were intended to acquire

the profile lexical and semantic style.
Subsequently, it is computed a Punctuation marks term frequency vector which holds the number of occurrences of a given punctuation mark in the author profile. The possible terms this vector could have are precomputed finding the dataset vocabulary punctuation marks. This vector gives the model the capability to quantify the punctuation writing style similarity between different authors and belongs to the PMR in our model.
Lastly, an auxiliary word term frequency vector is calculated in the same way the PMR builds its vector. This final vector holds the AWR.

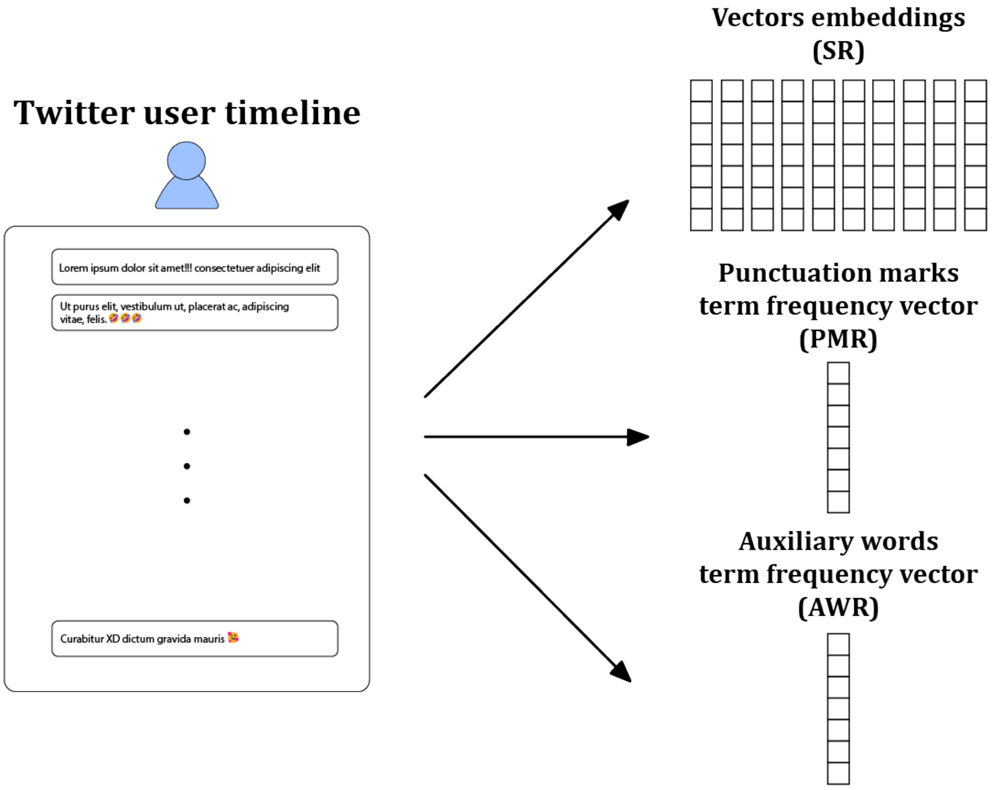
Twitter user timeline
Lorem ipsum dolor sit amet!!! consectetuer adipiscing elit
Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis.
Curabitur XD dictum gravida mauris
Vectors embeddings (SR)
Punctuation marks term frequency vector (PMR)
Auxiliary words term frequency vector (AWR)

Figure 1: Author profile representations

2.3.2. Similarity metric

In our proposal, we need a measure that quantifies how similar any two profiles are. We fix this measure like the mean of each representation's similarities.
The SR similarity is computed by taking the mean from the highest vector embedding pair similarities (v_i, v_j) such i and j are not associated, where i represents the i^{th} dense vector associated to a tweet from a profile A and j represents the j^{th} dense vector associated to a tweet from a profile B. This is illustrated in Figure 2.
We calculate the vector's similarities using the usual cosine similarity:

$$sim(v_1, v_2) = \cos(\theta) = \frac{v_1.v_2}{\|v_1\|\|v_2\|} \quad (1)$$

Being v_1 and v_2 vector embeddings associated to two tweets.

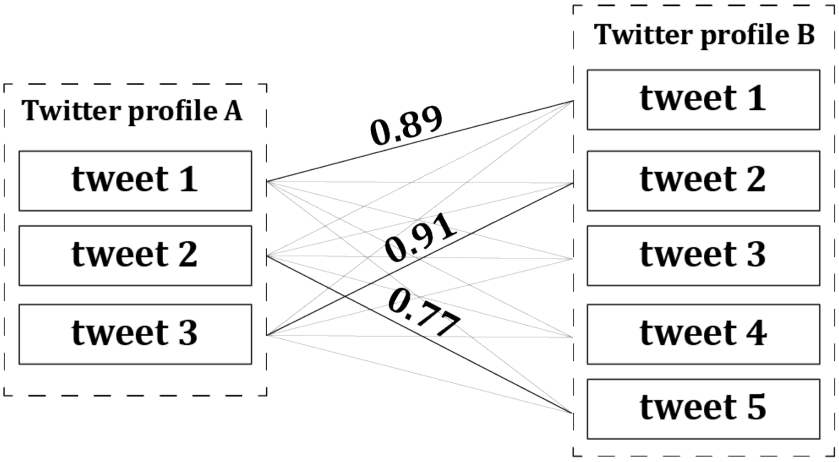

Figure 2: Semantic Representation Similarity Calculation Method.

In the previous graph, the numbers on the lines that connect two tweets represent the greatest cosine similarities between the embeddings of the encodings of the tweets of profile A and the tweets of profile B. The general similarity value of profiles A and B would be the average of these similarities (i.e 0.85).

To compute the semantic similarity between pairs of tweets, this same methodology is applied, considering the sentences of those tweets. This approach seeks to avoid the influence in the comparison of documents (paragraphs) with different amounts of sentences and to build the similarity based on the most similar tweets.

As the PMR and the AWR calculated the same length vectors for every profile pair, we take advantage of this fact and compute its corresponding similarity with the same cosine similarity shown in Equation 1

2.3.3. Core basis of profiling method

When the model attempts to classify a non-labeled profile into one of the two classes (ISS or non-ISS) we state that it should belong to the class where the accumulated sum of the most similar k profiles to the unknown profile is the biggest. This is illustrated in Figure 3 for $k = 1$ (i.e the unlabeled profile will belong to the class that contains the most similar profile to it).

For some $k > 1$ the model computes an accumulate sum on each class.

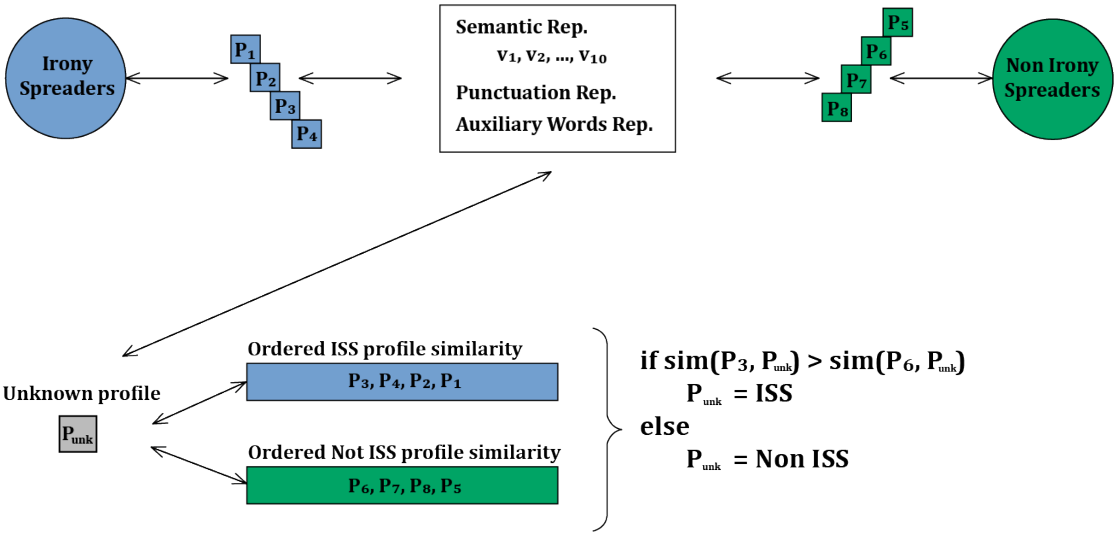

Figure 3: Irony spreader detection strategy 1-NN example classification

3. Experimental results

3.1. Parameter fitting

We fix the most similar k profiles, the unlabeled profiles require to make the accumulated sum as our model tunable parameter. Our model's parameter fitting is focused on finding the best k such that the accuracy of predictions increases. We also estimate how are the interactions between representations by testing some representations and dropping the remaining.

3.2. Dataset Specification

A training corpus was released to train and validate competitor's models[3]. This dataset is composed of 420 English Twitter user profiles timelines composed each one with 200 tweets, along with a ground truth file holding the given profiles classification. A test dataset was released for testing the models as well. The test corpus is composed of 180 Twitter timelines. As organizers state in the competition's description, the whole dataset is balanced (i.e the number of ISS profiles and non-ISS profiles released is the same). The quality metric proposed to evaluate the competitor's results in this task was accuracy.

3.3. Developed experiments

First of all, we represent the whole training dataset into the proposed representations: SR, PMR, and AWR. To estimate the best k and representation combination we run a 5-fold cross-validation over the training corpora, testing on each cross all possible k and representations

[3] https://pan.webis.de/clef22/pan22-web/author-profiling.html

Table 1
5-fold cross validation over training corpora

SR	PMR	AWR	Validation dataset mean accuracy							
			K=1	K=7	K=13	K=19	K=25	K=31	K=37	K=41
×			0,59	0,59	0,60	0,63	0,66	0,68	0,72	**0,79**
	×		0,53	0,55	0,56	0,59	0,60	0,60	0,60	0,64
		×	0,55	0,57	0,58	0,59	0,59	0,59	0,59	0,59
×	×		0,54	0,57	0,55	0,60	0,61	0,61	0,64	0,69
×		×	0,56	0,59	0,59	0,60	0,60	0,61	0,63	0,65
	×	×	0,52	0,57	0,58	0,58	0,58	0,58	0,58	0,59
×	×	×	0,55	0,57	0,58	0,59	0,60	0,60	0,61	0,64

Table 2
5-fold cross validation over training dataset with models: Single Representation and Majority vote

Model	Cross number					Statistics	
	Cross 1	Cross 2	Cross 3	Cross 4	Cross 5	Mean	Std. Dev.
Single Representation	0.50	0.67	0.84	0.83	0.83	**0.73**	0.14
Majority vote	0.51	0.60	0.73	0.75	0.75	0.67	**0.10**

permutations. We illustrate in Table 1 the best k obtained results and the respective accuracy of the model obtained using that parameter.

We also considered testing a slightly different approach where instead of having a single representation combination, have three different combinations (the ones with best results), each one emitting a vote to determine the unknown profile predicted class. In this version, an unknown profile belongs to the majority voted class from each of the representations.

3.4. Results analysis

From the developed experiments we elaborate two possibilities to test models: a single representation one and a majority vote model. We run a 5-fold cross-validation over training to compare the two models. Results are illustrated in Table 2.
In the test set, it was tested the Majority vote approach, getting an accuracy of 0.63, consistent with validation set tests.

4. Conclusion and future work

Making a system able to extract author feature as age, gender or political orientation is a challenging and interesting task in the research community. This work shows the results obtained by the UO-UDC team at the Profiling Irony and Stereotype Spreaders on Twitter shared task hosted by PAN. We strongly believe that adding some other lexical-semantic representations to our model could improve its accuracy. For future work, we considered adding a word phonetic representation that expresses some irony regarding words likely considering their pronunciation due to the Twitter tweet's informal nature. We also consider it appropriate to evaluate an approach in which each representation contributes in a different and weighted way to the general similarity measure.

Acknowledgments

This work was supported by projects PLEC2021-007662 (MCIN/AEI/10.13039/501100011033, Ministerio de Ciencia e Innovación, Agencia Estatal de Investigación, Plan de Recuperación, Transformación y Resiliencia, Unión Europea-Next Generation EU) and RTI2018-093336-B-C22 (Ministerio de Ciencia e Innovación, Agencia Estatal de Investigación). The second author also thanks the financial support supplied by the Consellería de Cultura, Educación e Universidade (GPC ED431B 2022/33)

References

[1] F. Balouchzahi, H. Shashirekha, Las for hasoc-learning approaches for hate speech and offensive content identification., in: FIRE (Working Notes), 2020, pp. 145–151.

[2] F. Rangel, G. L. De la Peña Sarracén, B. Chulvi, E. Fersini, P. Rosso, Profiling hate speech spreaders on twitter task at pan 2021., in: CLEF (Working Notes), 2021, pp. 1772–1789.

[3] A. Reyes, P. Rosso, D. Buscaldi, From humor recognition to irony detection: The figurative language of social media, Data & Knowledge Engineering 74 (2012) 1–12. URL: https://www.sciencedirect.com/science/article/pii/S0169023X12000237. doi:https://doi.org/10.1016/j.datak.2012.02.005, applications of Natural Language to Information Systems.

[4] E. Fersini, J. Armanini, M. D'Intorni, Profiling Fake News Spreaders: Stylometry, Personality, Emotions and Embeddings—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/.

[5] D. Espinosa, H. Gómez-Adorno, G. Sidorov, Profiling Fake News Spreaders using Characters and Words N-grams—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/.

[6] X. Duan, E. Naghizade, D. Spina, X. Zhang, RMIT at PAN-CLEF 2020: Profiling 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/.

[7] Á. Carracedo, R. J. Mondéjar, Profiling Hate Speech Spreaders on Twitter—Notebook for PAN at CLEF 2021, in: G. Faggioli, N. Ferro, A. Joly, M. Maistro, F. Piroi (Eds.), CLEF 2021 Labs and Workshops, Notebook Papers, CEUR-WS.org, 2021. URL: http://ceur-ws.org/Vol-2936/paper-152.pdf.

[8] C. Bagdon, Profiling Spreaders of Hate Speech with N-grams and RoBERTa—Notebook for PAN at CLEF 2021, in: G. Faggioli, N. Ferro, A. Joly, M. Maistro, F. Piroi (Eds.), CLEF

2021 Labs and Workshops, Notebook Papers, CEUR-WS.org, 2021. URL: http://ceur-ws.org/Vol-2936/paper-155.pdf.

[9] 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).

[10] O.-B. Reynier, C. Berta, R. Francisco, R. Paolo, F. Elisabetta, Profiling Irony and Stereotype Spreaders on Twitter (IROSTEREO) at PAN 2022, in: CLEF 2022 Labs and Workshops, Notebook Papers, CEUR-WS.org, 2022.

[11] J. Bevendorff, B. Chulvi, E. Fersini, A. Heini, M. Kestemont, K. Kredens, M. Mayerl, R. Ortega-Bueno, P. Pezik, M. Potthast, F. Rangel, P. Rosso, E. Stamatatos, B. Stein, M. Wiegmann, M. Wolska, E. Zangerle, Overview of PAN 2022: Authorship Verification, Profiling Irony and Stereotype Spreaders, and Style Change Detection, in: M. D. E. F. S. C. M. G. P. A. H. M. P. G. F. N. F. Alberto Barron-Cedeno, Giovanni Da San Martino (Ed.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Thirteenth International Conference of the CLEF Association (CLEF 2022), volume 13390 of *Lecture Notes in Computer Science*, Springer, 2022.

[12] 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.

[13] H. Wu, Y. Liu, J. Wang, Review of text classification methods on deep learning, Comput. Mater. Contin 63 (2020) 1309–1321.

[14] S. Hashida, K. Tamura, T. Sakai, Classifying tweets using convolutional neural networks with multi-channel distributed representation, IAENG International Journal of Computer Science 46 (2019) 68–75.

[15] 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).