TuEval at SemEval-2019 Task 5: LSTM Approach to Hate Speech Detection in English and Spanish

Mihai Manolescu
University of Tübingen
mihai.manolescu*

Denise Löfflad
University of Tübingen
denise.loefflad*

Adham Nasser Mohamed Saber
University of Tübingen
adham-nasser.mohamed-saber*

Masoumeh Moradipour Tari
University of Tübingen
masoumeh.moradipour-tari*

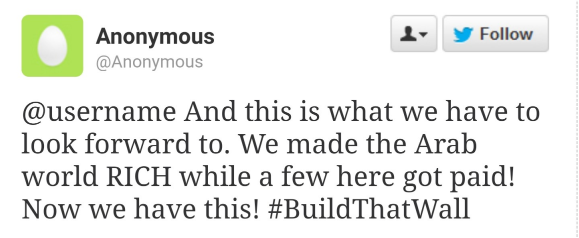
Abstract

The detection of hate speech, especially in online platforms and forums, is quickly becoming a hot topic as anti-hate speech legislation begins to be applied to public discourse online. The HatEval shared task was created with this in mind; participants were expected to develop a model capable of determining whether or not input (in this case, Twitter posts in English and Spanish) could be considered hate speech (designated as Subtask A), if they were aggressive, and whether the tweet was targeting an individual, or speaking generally (Subtask B). We approached this Subtask by creating a LSTM model with an embedding layer. We found that our model performed considerably better on English language input when compared to Spanish language input. In English, we achieved an F1-Score of 0.466 for Subtask A and 0.462 for Subtask B; In Spanish, we achieved scores of 0.617 and 0.612 on Subtask A and Subtask B, respectively.

1 Introduction

Social media plays an important role nowadays and dominates everyday life. Social networks like Facebook, Twitter and Instagram are platforms where people express thoughts, feelings and emotions regarding themselves or others. This can lead to different opinions colliding and creating conflicts. Often, feelings are not expressed objectively and can be offensive to other users. In order to make social media more comfortable, so called hate speech needs to be detected and removed. Hate speech is here defined as: Any communication that disparages a person or a group on the basis of some characteristic such as race, color, ethnicity, gender, sexual orientation, nationality, religion, or other characteristics (Basile et al., 2019). To assure there is no spread of illegal hate speech, the EU has created a code of conduct for social media platforms (European Union, 2018) that needs to be followed. According to these EU regulations, social media platforms must regulate hateful speech. In addition, social media occupies an increasingly larger portion of public discourse; even without these EU regulations, it seems that these platforms should have some methods for controlling violent discourse.

For these reasons, the HatEval shared task (Basile et al., 2019) was created. The task is divided into two subtasks; Subtask A is hate speech detection against immigrants and women, a binary classification problem where a tweet is classified as either hateful or not hateful. Subtask B is determining whether a given tweet is aggressive, and whether it is targeting an individual, or not referring to any particular person. Further, each of these Subtasks is evaluated on English tweets and using Spanish tweets. We were provided a 9000-tweet English training set, and a 5000 tweet Spanish training set. The training sets were manually tagged as hateful or not hateful, aggressive or not aggressive, and targeted or not targeted - examples of tweets marked as hateful can be seen in Figures 1 and 2 below.



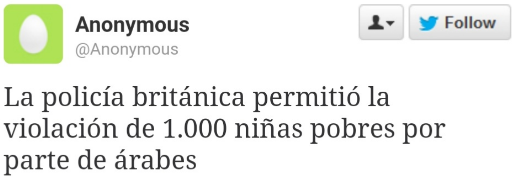
Anonymous
@Anonymous

@username And this is what we have to look forward to. We made the Arab world RICH while a few here got paid! Now we have this! #BuildThatWall

Figure 1: Example of a hateful English tweet

In this paper, we detail our methods for approaching these problems. We will first cover related works before detailing our specific so-

*@student.uni-tuebingen.de



Anonymous
@Anonymous

La policía británica permitió la violación de 1.000 niñas pobres por parte de árabes

Figure 2: Example of a hateful Spanish tweet

lutions for Subtask A and Subtask B; we will then cover our model (and previously attempted models) and present our results. Hate speech detection is naturally a far reaching topic, and in conclusion we will discuss the implications of our work for the field in general.

2 Related Work

To begin, we attempted to take a brief survey of previous work in the field of hate speech detection. Since this is, at heart, a binary classification task, we saw that there were many established approaches to solving this problem - various machine learning techniques, according to our research, were shown to be valid, such as Recurrent and Convolutional Neural Networks (RNN and CNNs) (Stammbach et al., 2018), Support Vector Machines (SVMs) (Malmasi and Zampieri, 2017), Long Short Term Memory models (LSTMs) (Zhang et al., 2015; Risch et al., 2018), as well as simpler linear regression approaches (Kent, 2018). In our estimation, we determined that LSTM approaches were most successful (Golem et al., 2018; Del Vigna12 et al., 2017), and took such an approach in the creation of our model. Some other approaches were too computationally expensive; in addition, we felt that, due to the nebulous nature of hate speech determination, the additional information captured by an LSTM model would be worthwhile in these tasks. We also determined that, for such a task, the use of non-word features would be superfluous, as previous work had shown them to decrease performance (Stammbach et al., 2018), and this was supported by other works on simple classification tasks, even when LSTMs or RNNs were not used (Malmasi and Zampieri, 2017). Research showed that various features, including emoticons, sentiment analysis, and number of characters tended to hurt performance (Kent, 2018).

Predictably, most work done on this topic has focused on English language data; we found only a few papers on Spanish language hate speech detection (Álvarez-Carmona et al., 2018; Fersini et al., 2018), which we attempted to use to ensure our model would function across language boundaries.

3 Model

At the outset, we employed a simple unidirectional, 1-layer LSTM model. As we saw preliminary results we altered our model accordingly. We also attempted to use a 2-layer LSTM model, and settled on a 1-layer LSTM model with a simple embedding layer, using mainly the *Keras* (Chollet et al., 2015) library.

3.1 Pre-Processing

Based on our research, we saw that limited pre-processing of the data set could improve performance; to that end, the following pre-processing steps were taken:

- replace usernames with username markers
- remove punctuation and special characters (@ / , ; . : ? ¿ ¡ ! $)
- lowercase

We made the decision not to omit hashtags; while usernames do not necessarily convey information pertinent to the tweet itself, it was determined that hashtags are frequently used for meaningful purposes and must be considered when attempting to classify Twitter data. We attempted to expand our pre-processing efforts when dealing with Spanish language data after seeing early results (replacing characters such as ‘ñ´ with ‘n´ for example), but without success; such efforts hurt our model more than they helped.

3.2 Recurrent Neural Network

Our model used character based representations of all data. We used an embedding layer with input dimension of 5000 and an output of 28; input length was determined by finding the length of the longest item in the data set, and padding all representations to this length. Additionally, we used an LSTM layer with 64 units, with a dropout rate of 0.1 (determined after simple trial and error tests), and our model employed a sigmoid activation function and a binary cross entropy loss function. Our model was trained for 50 epochs on the English language dataset, and 20 epochs on the Spanish language dataset.

4 Evaluation

Models	F1-score
1-Layer LSTM	0.31
2-Layer LSTM	0.42
1-Layer LSTM w/ Embedding	0.69

Table 1: Development set F1-scores for preliminary testing of models

We first evaluated our preliminary models using the development data set, specifically using results of Subtask A in English to determine which of our beginning approaches was most successful. After this determination, we expanded upon our best working model (the simple LSTM model with embedding layer), and proceeded to use this approach to handle all tasks in both languages. We calculated the F1-score for each of our models, and used this for our evaluations. As shown in Table 1, the 1-Layer LSTM model with Embedding outperformed our other two models significantly and achieved an F1-Score of 0.69 on the development set.

tasks	Accuracy	Precision	Recall	F1-score	EMR
Subtask A (En.)	0.488	0.548	0.533	0.466	N/A
Subtask B (En.)	0.565	0.497	0.482	0.462	0.173
Subtask A (Sp.)	0.630	0.618	0.617	0.617	N/A
Subtask B (Sp.)	0.680	0.629	0.608	0.612	0.428

Table 2: Results for Tasks A and B in English and Spanish

The average results for each metric are shown in Table 2. The final ranking for Subtask A for English, as well as Spanish, was based on the F1-score. Our F1-score was 0.466 for English, which ranked us 27$^{\text{th}}$ out of 69 teams that submitted a result for this Task. Since we had some problems with the Spanish data set, we could only submit one solution for Subtask A, which placed us 36$^{\text{th}}$ out of 39 teams.

Evaluation for Subtask B was based on two criteria - partial match and exact match. For partial match, each dimension that needs to be predicted, is being looked at independently and therefore the usual evaluation metrics are being used (Precision, Accuracy, Recall and F1-Score). For the exact match all the dimensions to be predicted are jointly considered. Ranking was solely based on the score of the *Exact Match Ratio* (EMR). For English we achieved an EMR score of 0.173, which ranked us second to last, even if our average F1-score was higher than other systems'. Since we had the same problems as in Subtask A, we again were only able to submit one file in Subtask B for Spanish, where we achieved an EMR of 0.428 and an average F1-Score of 0.612. The significant difference for the F1-Score between English and Spanish comes from the fact that there was less training data for Spanish compared to English.

5 Conclusion & Future Work

We created a simple LSTM model and applied it to all tasks - detecting hate speech, determining aggression, and determining targeted or general speech, achieving F1-scores of 0.466 and 0.462 for Subtask A and B in English, and scores of 0.617 and 0.612 for Tasks A and B in Spanish. In our work, we saw that our model performed considerably better on English language data when compared to Spanish language data. We were not able to reduce this discrepancy with additional pre-processing of Spanish language data. The difference in performance may be explained by the nature of Spanish language discourse online - perhaps there is greater accent- or dialect-based difference in Spanish when compared to English (Çöltekin and Rama, 2018), which could confound attempts to train a model off of a Spanish language corpus that does not specifically control for dialect.

There is still much work to be done in the field of hate speech evaluation. It is possible that a large improvement in performance would be seen if word representations were used instead of character representations; much of the vocabulary of online communication and discourse involves the use of colloquialisms, informal speech, and metaphorical language, which word based representations could perhaps better capture. Further, contextual information from the rest of a particular tweet could also help in determining whether or not a given word is being used in a malicious way; this information could have been captured through the use of n-gram models or contextual word representation methods. Using meta-information about a particular user, topic, or hashtag could have also improve performance (Schmidt and Wiegand, 2017); such methods go outside the scope of the shared task, but it is conceivable that a platform such as Twitter could consider previous tweets of a given user, or perhaps topic modelling methods, in a commercial hate speech detection model (for example, it seems rational to consider a tweet with a topic such as 'right wing politics' more likely to be hate speech than a tweet with the topic 'gardening'). The use of lexical resources like lists of slurs have also shown to be effective in combination with other features (Schmidt and Wiegand, 2017; Davidson et al., 2017). Work could also be done in hate speech detection in long form documents; it goes without saying that a model that can effectively detect hate speech in short, one- to three-sentence tweets will not necessarily perform as well on longer corpora, such as articles. In these cases, context-based word representations, n-gram models, etc. could become even more valuable.

References

Álvarez-Carmona, M. Á., Guzmán-Falcón, E., Montes-y Gómez, M., Escalante, H. J., Villasenor-Pineda, L., Reyes-Meza, V., and Rico-Sulayes, A. (2018). Overview of MEX-A3T at IberEval 2018: Authorship and aggressiveness analysis in Mexican Spanish tweets. In *Notebook Papers of 3rd SEPLN Workshop on Evaluation of Human Language Technologies for Iberian Languages (IBEREVAL), Seville, Spain*, volume 6.

Basile, V., Bosco, C., Fersini, E., Nozza, D., Patti, V., Rangel, F., Rosso, P., and Sanguinetti, M. (2019). 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 (SemEval-2019)*. Association for Computational Linguistics.

Chollet, F. et al. (2015). Keras. https://keras.io.

Çöltekin, Ç. and Rama, T. (2018). Tübingen-Oslo at Semeval-2018 task 2: SVMs perform better than RNNs in Emoji Prediction. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 34–38.

Davidson, T., Warmsley, D., Macy, M., and Weber, I. (2017). Automated hate speech detection and the problem of offensive language. In *Eleventh International AAAI Conference on Web and Social Media*.

Del Vigna12, F., Cimino23, A., Dell'Orletta, F., Petrocchi, M., and Tesconi, M. (2017). Hate me, hate me not: Hate speech detection on Facebook.

European Union (2018). Code of conduct on countering illegal hate speech online. https://ec.europa.eu/info/sites/info/files/code_of_conduct_on_countering_illegal_hate_speech_online_en.pdf. [Online; accessed 16-February-2019].

Fersini, E., Rosso, P., and Anzovino, M. (2018). Overview of the task on automatic misogyny identification at Ibereval 2018.

Golem, V., Karan, M., and Šnajder, J. (2018). Combining Shallow and Deep Learning for Aggressive Text Detection. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018)*, pages 188–198.

Kent, S. (2018). German Hate Speech Detection on Twitter. *Proceedings of the GermEval 2018 Workshop*.

Malmasi, S. and Zampieri, M. (2017). Detecting Hate Speech in Social Media. *arXiv preprint arXiv:1712.06427*.

Risch, J., Krebs, E., Löser, A., Riese, A., and Krestel, R. (2018). Fine-Grained Classification of Offensive Language. *Proceedings of the GermEval 2018 Workshop*.

Schmidt, A. and Wiegand, M. (2017). A survey on hate speech detection using natural language processing. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*, pages 1–10.

Stammbach, D., Zahraei, A., Stadnikova, P., and Klakow, D. (2018). Offensive Language Detection with Neural Networks for Germeval Task 2018. *Proceedings of the GermEval 2018 Workshop*.

Zhang, X., Zhao, J., and LeCun, Y. (2015). Character-level Convolutional Networks for Text Classification. In *Advances in neural information processing systems*, pages 649–657.