JCTDHS at SemEval-2019 Task 5: Detection of Hate Speech in Tweets using Deep Learning Methods, Character N-gram Features, and Preprocessing Methods

Yaakov HaCohen-Kerner, Elyashiv Shayovitz,
Shalom Rochman, Eli Cahn, Gal Didi, and Ziv Ben-David
Department of Computer Science, Jerusalem College of Technology, Lev Academic Center
21 Havaad Haleumi St., P.O.B. 16031, 9116001 Jerusalem, Israel
kerner@jct.ac.il, elyashiv12@gmail.com,
shal.rochman@gmail.com, eli.cahn@gmail.com,
galdd8@gmail.com, and benda1237@gmail.com

Abstract

In this paper, we describe our submissions to SemEval-2019 contest. We tackled subtask A - "a binary classification where systems have to predict whether a tweet with a given target (women or immigrants) is hateful or not hateful", a part of task 5 "Multilingual detection of hate speech against immigrants and women in Twitter (HatEval)". Our system JCTDHS (Jerusalem College of Technology Detects Hate Speech) was developed for tweets written in English. We applied various supervised ML methods, various combinations of n-gram features using the TF-IDF scheme. In addition, we applied various combinations of eight basic preprocessing methods. Our best submission was a special bidirectional RNN, which was ranked at the 11th position out of 68 submissions.

1 Introduction

Hate Speech is usually defined as communication that contains contempt or hatred towards a person or a group of people on the basis of some characteristic e.g., color, ethnicity, gender, nationality, race, religion, and sexual orientation.

The phenomenon of hate speech in social media has grown in recent years (Eadicicco. 2014). A strong connection between hate speech and actual hate crimes has been shown in Watch (2014). In light of the huge amount of information in social media, early detection of people using hate speech could prevent them from carrying out their hate speech. Therefore, the detection of hate speech in social media has become an issue of increasing importance (Moulson. 2016).

In this paper, we describe our six models (each model with another team member as the first author) submitted to task 5-A for tweets written in English. The full description of this task is given in Basile et al. (2019).

The structure of the rest of the paper is as follows. Section 2 introduces a background concerning hate speech, tweet classification, and data preprocessing. Section 3 presents, in general, the description of Task 5. In Section 4, we describe the submitted models and their experimental results. Section 6 summarizes and suggests ideas for future research.

2 Background

2.1 Hate Speech

Waseem and Hovy (2016) introduced a list of criteria founded in critical race theory and used them to label a publicly available corpus of more than 16k tweets with tags about both racial and sexist offenses. Nobata et al. (2016) developed a machine learning based method to detect hate speech on online user comments from two domains. They also built a corpus of user comments annotated accordingly to three subcategories (hate speech, derogatory, profanity). Schmidt and Wiegand (2017) introduced a survey of the NLP methods that were developed in order to detect hate speech. Davidson et al. (2017) presented a multi-class classifier to distinguish between three categories: hate speech, offensive language, and none of these two. The analysis of the predictions and the errors shows when they can reliably separate hate speech from other types of offensive language (e.g., tweets with the highest predicted probabilities of being hate speech tend to contain multiple racial

Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019), pages 426–430
Minneapolis, Minnesota, USA, June 6–7, 2019.

or homophobic slurs) and when this differentiation is more difficult (e.g., many tweets misclassified as hate speech contain terms that can be considered racist and sexist; however it is apparent that many Twitter users use this type of language in their everyday communications). Anzovino et al. (2018) built a labeled corpus containing 2,227 misogynous (hate speech against women) tweets and no-misogynous tweets and explored various NLP features and ML models for detecting and classifying misogynistic language.

2.2 Tweet Classification

Tweet classification is the task to automatically classify a tweet into one of a set of predefined classes. This research domain has been growing rapidly in recent years. Twitter as one of the leading social networks presents challenges to the researchers since tweets are informal, short, and contain various misspellings, shortenings, and slang words (HaCohen-Kerner et al., 2017).

2.3 Data preprocessing

Data preprocessing is an important step in data mining (DM) and ML processes. In data files, it is common to find typos, emojis, slang, HTML tags, spelling mistakes, irrelevant and redundant information. Analyzing data that has not been carefully cleaned or pre-processed might lead to misleading results.

Not all of the preprocessing types are considered effective by all text classification (TC) researchers. For instance, Forman (2003), in his study on feature selection metrics for TC, claimed that stop words frequently occur and are ambiguous and therefore should be removed, However, HaCohen-Kerner et al. (2008) demonstrated that the use of word unigrams including stop words lead to improved TC results compared to the results obtained using word unigrams excluding stop words in the domain of Hebrew-Aramaic Jewish law documents.

In our system, we applied various combinations of eight basic preprocessing types: C - spelling Correction (The spelling correction is performed using an autocorrect library, written by McCallum (2014)[1]), H – HTML Tags Removal, L – converting to Lowercase letters, P – Punctuation removal, S – Stopwords Removal, R – Repeated characters removal (repeated characters were removed and only one character was left), T – sTemming, and M - leMmatizion) in order to check whether they improve TC or not.

3 Task Description

Task 5 deals with two tasks related to hate speech detection in Twitter with two specific targets, women and immigrants, for tweets in English and Spanish. We participated only in Task 5-A for tweets written in English, i.e., a two-class classification task where we have to predict whether a tweet with a given target (women or immigrants) is hateful or not hateful.

The datasets of Task 5-A consists of Train, Dev and Test datasets. The Train dataset contains 9,000 categorized tweets: 3,783 HS (hateful speech) tweets and 5,217 NHS (not hateful speech) tweets. The Dev dataset first published without labels, and they were added only in the final evaluation phase. The Dev set contains 1,000 tweets: 427 HS tweets, and 572 NHS tweets. The Test dataset contains 3,000 uncategorized Tweets.

4 The Submitted Models and Experimental Results

We have submitted six models (one for each author) to task 5-A for tweets written in English. The general TC algorithm is as follows.

1. Using a TF-IDF scheme, find the optimal number of word n-grams and combination of pre-processing types.
2. Apply various supervised ML methods including RNN models and others to find the best accuracy results.

Table 1 presents the main characteristics and results of our six submitted models in descending order according to their F1-Macro (F-M) score on the test set. Figure 1 presents our final RNN model using n-gram features in layer N.

[1] https://github.com/phatpiglet/autocorrect/ Last access date: 19-MAR-19.

The first name of the model authors	Pre-processing	Model			Score		
		RNN Architecture	N-gram Features	Its Fully Connected Layer uses	CV Score (F- M)	Test Score (F- M)	Rank
galdd8@ gmail.com	CHLPRS	Bidirectional RNN with 4 hidden layers. Each layer contains128 LSTM units and Dropout layer (0.4). GloVe of 200d used for embedding	100 char trigrams, no skips	Logistic Regression	0.737	0.5	11 \ 68
elyashiv12 @ gmail.com	None	RNN contains 128 LSTM units, and Dropout layer (0.3). GloVe of 200d used for embedding	None	None	0.751	0.49	15 \ 68
kerner@ jct.ac.il	None	Bidirectional RNN with 4 hidden layers. Each layer contains128 LSTM units and Dropout layer (0.4). GloVe of 200d used for embedding	None	None	0.754	0.48	21 \ 68
ShalomRo chman	CHLPRS	Bidirectional RNN with 4 hidden layers. Each layer contains128 LSTM units and Dropout layer (0.4). GloVe of 200d used for embedding	200 char bigrams, no skips	SVM (SGD Variant)	0.749	0.42	42 \ 68
benda1237 @ gmail.com	CHLPRS	Bidirectional RNN with 4 hidden layers. Each layer contains128 LSTM units and Dropout layer (0.4). GloVe of 200d used for embedding	200 word unigram, no skips	SVM (SGD Variant)	0.713	0.41	41 \ 68
ecahn	CHLPRS	Bidirectional RNN with 2 hidden layers. Each layer contains128 LSTM units and Dropout layer (0.4). GloVe of 200d used for embedding	300 char bigrams, no skips	SVM (SVC Variant)	0.759	0.38	54 \ 68

Table 1: Results of our 6 models in task-A.

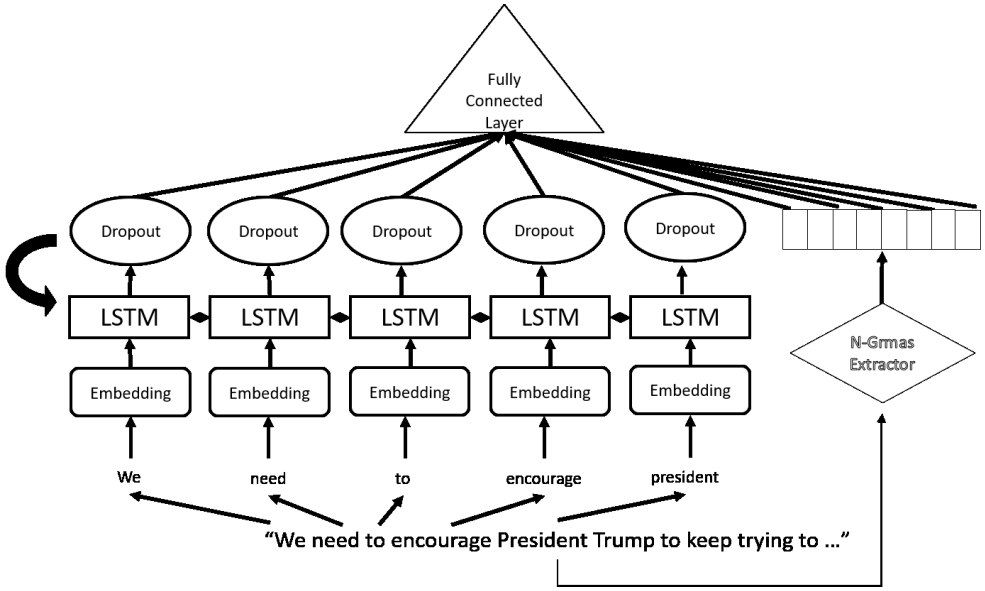

Figure 1: Final RNN model with n-gram features in layer N.

Analysis of the results presented in Table 1 shows that our best Macro F-measure score (as opposed to F1 of hate speech alone) for the test set (0.5) was obtained by a bidirectional RNN with 4 hidden layers, 128 LSTMs, 0.4 Dropout and a GloVe (Pennington et al., 2014) of 200d special for Twitter as an embedded layer. The best combination of pre-processing types was found to be CHLPRS, which means to activate all pre-processing types.

In our experiments, we used the following framework Python 3.6 with Keras[2] and Scikit-Learn in PyCharm IDE (Pedregosa et al., 2011) using the TF-IDF scheme called TfidfTransformer[3]). The accuracy of each ML model was estimated by a 5-fold cross-validation testing. The vocabulary words were used as zero-vectors during the word-to-embedding conversion. The Fully connected layer (FC) is the last layer in RNN models. It performs the final classification. The activation function of the FC layer is the sigmoid function. We used the RMSProp optimizer and 30 epochs for each model.

5 Conclusions and Future Research

In this paper, we describe our submissions to Task 5-A of SemEval-2019 contest. Our system JCTDHS (Jerusalem College of Technology Detects Hate Speech) was developed for tweets written in English. We used a TF-IDF scheme and we performed various combinations of six pre-processing methods to improve the performance.

Our best submission for Task 5-A was a bidirectional RNN with 4 hidden layers while each layer contains128 LSTM units, a dropout layer (0.4), and a GloVe (Global Vectors for Word Representation) of 200d that was used for embedding. GloVe was developed by the Stanford NLP Group (Pennington et al., 2014). It applies a co-occurrence matrix and by using matrix factorization.

This submission was ranked at the 11th out of 68 submissions for tweets written in English.

Future research proposals are as follows. It is known that many tweets include acronyms (abbreviations) that are presented in different forms. Acronym disambiguation (HaCohen-Kerner et al., 2010A), i.e., selecting the correct long form of the acronym depending on its context will enrich the tweet's text and will enable better TC.

More ideas that may contribute to better classification are implementing TC using (1) additional feature sets such as stylistic feature sets (HaCohen-Kerner et al., 2010B) and key phrases that can be extracted from the text files (HaCohen-Kerner et al., 2007) and (2) additional deep learning models.

[2] https://github.com/gucci-j/vae.

[3] https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfTransformer.html#sklearn.feature_extraction.text.TfidfTransformer

Acknowledgments

This research was partially funded by the Jerusalem College of Technology, Lev Academic Center.

References

Maria Anzovino, Elisabetta Fersini, and Paolo Rosso. 2018. Automatic Identification and Classification of Misogynistic Language on Twitter. International Conference on Applications of Natural Language to Information Systems. Springer, Cham.

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Rangel, Paolo Rosso, and Manuela Sanguinetti. 2019. SemEval-2019 Task 5: Multilingual Detection of Hate Speech Against Immigrants and Women on Twitter. Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019), Association for Computational Linguistics, Minneapolis, Minnesota, USA.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In: Proceedings of the 12th International AAAI Conference on Web and Social Media.

Lisa Eadicicco. 2014. This female game developer was harassed so severely on twitter she had to leave her home. Business Insider, 12(10).

George Forman. 2003. An extensive empirical study of feature selection metrics for text classification. Journal of machine learning research, 3(Mar), 1289-1305.

Yaakov HaCohen-Kerner, Ittay Stern, David Korkus, and Erick Fredj. 2007. Automatic machine learning of keyphrase extraction from short HTML documents written in Hebrew. *Cybernetics and Systems: An International Journal*, *38*(1), 1-21.

Yaakov HaCohen-Kerner, Dror Mughaz, Hananya Beck, and Elchai Yehudai 2008. Words as classifiers of documents according to their historical period and the ethnic origin of their authors. Cybernetics and Systems: An International Journal, 39(3), 213-228.

Yaakov HaCohen-Kerner, Ariel Kass, and Ariel Peretz. 2010A. HAADS: A Hebrew Aramaic abbreviation disambiguation system. Journal of the American Society for Information Science and Technology, 61(9), 1923-1932.

Yaakov HaCohen-Kerner, Hananya Beck, Elchai Yehudai, and Dror Mughaz. 2010B. Stylistic feature sets as classifiers of documents according to their historical period and ethnic origin. Applied Artificial Intelligence, 24(9), 847-862.

Yaakov HaCohen-Kerner, Ziv Ido, and Ronen Ya'akobov. 2017. Stance classification of tweets using skip char Ngrams. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases (pp. 266-278). Springer, Cham.

Hate Speech Watch. 2014. Hate crimes: Consequences of hate speech. http: //www.nohatespeechmovement. org/hate-speech-watch/focus/ consequences-of-hate-speech, June. Seen on 23rd Jan. 2016.

Jonas McCallum. 2014. Python 3 Spelling Corrector. https://pypi.python.org/pypi/autocorrect/0.1.0.

Geir Moulson. 2016. Zuckerberg in Germany: No place for hate speech on Facebook. http://abcnews.go.com/Technology/ wireStoryZuckerberg-place-hatespeech-facebook-37217309. Accessed 10/03/2016.

Chikashi Nobata, Joel Tetreault, Achint Thomas, Yashar Mehdad, and Yi Chang. 2016. Abusive language detection in online user content. In: Proceedings of the 25th International Conference on World Wide Web, pp. 145–153. International World Wide Web Conferences Steering Committee.

Fabian Pedregosa, Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R. and Dubourg, V. and Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. 2011. Scikit-learn: Machine learning in Python. Journal of machine learning research, 12(Oct), 2825-2830.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation.

Anna, Schmidt, and Michael Wiegand. 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. Association for Computational Linguistics, Valencia, Spain, pp. 1–10.

Zeerak Waseem, and Dirk Hov. 2016. Hateful symbols or hateful people? predictive features for hate speech detection on Twitter. In: SRW@ HLT-NAACL, pp. 88–93.