Tw-StAR at SemEval-2019 Task 5: N-gram embeddings for Hate Speech Detection in Multilingual Tweets

Hala Mulki*, Chedi Bechikh Ali, Hatem Haddad†§ and Ismail Babaoğlu***

*Department of Computer Engineering, Selcuk University, Turkey
**LISI Laboratory, INSAT, Carthage University, Tunisia
†RIADI Laboratory, National School of Computer Sciences, University of Manouba, Tunisia
§iCompass Consulting, Tunisia
halamulki@selcuk.edu.tr,chedi.bechikh@gmail.com
haddad.Hatem@gmail.com,ibabaoglu@selcuk.edu.tr

Abstract

In this paper, we describe our contribution in SemEval-2019: subtask A of task 5 "Multilingual detection of hate speech against immigrants and women in Twitter (HatEval)". We developed two hate speech detection model variants through Tw-StAR framework. While the first model adopted one-hot encoding n-grams to train an NB classifier, the second generated and learned n-gram embeddings within a feedforward neural network. For both models, specific terms, selected via MWT patterns, were tagged in the input data. With two feature types employed, we could investigate the ability of n-gram embeddings to rival one-hot n-grams. Our results showed that in English, n-gram embeddings outperformed one-hot n-grams. However, representing Spanish tweets by one-hot n-grams yielded a slightly better performance compared to that of n-gram embeddings. The official ranking indicated that Tw-StAR ranked 9th for English and 20th for Spanish.

1 Introduction

Under the guise of free speech, social media systems have been misused by some users who embed hatred, offensive, racist or negative stereotyping contents within their shared posts. Unfortunately, online Hate Speech (HS) is spreading widely, forming a serious problem that can lead to actual hate crimes (Matsuda, 2018). Many countries adopted laws prohibiting HS where people convicted of using HS can face large fines and even imprisonment. Although Twitter has its anti HS policy*, the increasing size of the daily-shared tweets in addition to multilingualism and informal writing issues evoke the necessity for automatic HS detection in tweets.

Hate speech detection problem has been addressed as a machine learning classification task. Recent studies proposed multiple HS detection models with different characteristic in terms of features, classification algorithms and implementation architectures. While some HS models employed hand-crafted features generated by NLP tools and external semantic resources, other models adopted text embedding features that are automatically learned from the corpus itself. Both feature types were fed to train either traditional classifiers such as Support Vector Machines (SVM), Naive Bayes (NB) and so forth, or more complicated deep learning-based classifiers such as Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM) and Recurrent Neural Network (RNN) (Schmidt and Wiegand, 2017). The variety of hand-crafted features enabled obtaining reliable performances. However, generating such features based on morphological NLP tools or semantic resources remains laborious. In contrast, embedding features are easier to generate and can yield good HS classification results when used within deep learning architectures (Yuan et al., 2016). Nevertheless, producing good performances via deep neural systems requires providing large-sized labeled training data, tuning many hyper parameters and high computation/time cost. In line with Tw-StAR framework (Mulki et al., 2017, 2018a), we propose, here, an HS model based on the hypothesis that, pairing between n-gram embeddings and less-complicated architectures i.e. feedforward neural network can lead to an efficient HS detection with least complexity.

2 Hate Speech Detection Models

According to the used features, HS detection models can be classified into hand-crafted-based and text embeddings-based.

*support.twitter.com/articles/20175050

Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019), pages 503–507
Minneapolis, Minnesota, USA, June 6–7, 2019.

2.1 Hand-Crafted-based Models

Being a user-generated content, HS terms tend to have variant writing shapes. (Waseem and Hovy, 2016) handled this issue by using char-grams to train an LR classifier. Combining char-grams with extra linguistic features such as word n-grams and user's gender improved the performance.

Additional user-related features were studied in (Unsvåg and Gambäck, 2018) within a multilingual HS detection task. Single and combined features were fed into an LR classifier. The study showed that specific user features favorably impact the performance.

The winning system (Pamungkas et al., 2018) in misogyny detection contest (Fersini et al., 2018) examined several sets of hand-crafted features including stylistic, lexical and structural. The features were formulated within one-hot/sparse encoding vectors and fed into an SVM classifier. It was noted that using features from HurtLex lexicon (Bassignana et al., 2018) enriched the lexical features set and enhanced the performance.

2.2 Text Embeddings-based Models

In these models, the input text is represented using dense, low-dimentional and real-valued vectors. In (Nobata et al., 2016), a comprehensive comparison was conducted among three embedding feature types: doc2vec (Le and Mikolov, 2014), word2vec and pretrained word embeddings against hand-crafted features. Using a regression model trained with the previous features, it could be noted that while doc2vec embeddings outperformed the other embedding features, combining them with all other features could further enhance the HS content recognition.

(Badjatiya et al., 2017) explored CNN, LSTM and FastText models to learn embedding features needed to classify HS contents. These models were trained by embedding features and evaluated against each other and towards SVM, LR and GBDT classifiers trained with hand-crafted features. Moreover, the authors explored training an GDBT classifier with word embeddings learned via various deep models. While CNN was the best-performing deep model, using the word embeddings learned via LSTM to train the simple less-complicated GDBT classifier improved the results.

In (Gambäck and Sikdar, 2017), context-aware word embeddings learned by word2vec, char 4-grams and a combination of both were used to train a CNN-based classifier. The proposed model was compared with an LR classifier trained via n-gram features (Waseem and Hovy, 2016). The results showed that regardless of the used embeddings type, CNN model outperformed the baseline model. Moreover, word2vec embeddings were of the best classification performance among the other embedding features.

3 Tw-StAR HS Detection Model

To detect HS in English and Spanish datasets provided by (Basile et al., 2019), Tw-StAR (see Figure 1) was applied through the following steps:

3.1 Preprocessing

- Initial preprocessing: includes removing the non-sentimental content such as URLs, usernames, digits, hashtag symbols and punctuation from both datasets (Mulki et al., 2018b).

- Stopwords removal: for English and Spanish, we removed stopwords using 1,012 English stopwords and 731 Spanish stopwrods derived from Terrier package[†] and snowball[‡], respectively.

- Lemmatization: we adopted Treetagger lemmatizer (Schmid, 1999); as it was used successfully for English and Spanish in (Mulki et al., 2018a). TreeTagger forms a language-independent tool to annotate texts with part-of-speech and lemma information.

- Hate indicatives tagging: Multi-word terms (MWT) are meaning indicators of a sentence/document (Henry et al., 2018; Bechikh-Ali et al., 2019). In our case, they can represent the entities discussed within a tweet. As our objective is to infer HS in tweets, we believe that recognizing MWT can assist in identifying the important entities related to hate speech or victims of hate speech. This has been practically noticed among the MWT extracted from the training set as we can mention: african_migrant, Iraqi_refugee_terrorist, Muslim_refugee, immigration_negative_effect. It should be noted that, MWT were extracted from hate tweets contained in the training set. Later, the extracted MWT were replaced in both training

[†] https://bitbucket.org/kganes2/
[‡] http://snowball.tartarus.org/

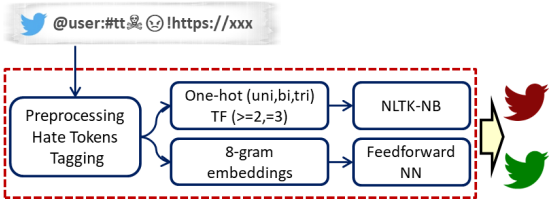

Figure 1: Tw-StAR framework

and dev/test sets with the tag "HateWord". MWT identification process was performed through two steps: (a) Shallow syntactic parsing where each word was tagged with its syntactic category using Treetagger that supports English and Spanish, and (b) MWT extraction conducted based on specific syntactic patterns of noun and adjective combinations using this schema:

$$\text{MWT}=(\text{Adj}|\text{N})^*(\text{N}|\text{NP})(\text{N}|\text{Adj}|\text{NP})^*$$

where * denotes a list of 0 or more elements, the MWT length varies between 2 and 4 words. Adj, N and NP refer to adjective, noun and proper noun, respectively.

3.2 Feature Extraction

Two types of features were generated to train both model variants of Tw-StAR.

- One-hot n-grams: are generated by subjecting the preprocessed tweets to tokenization. Three N-grams schemes including unigrams, bigrams and trigrams were adopted. For a certain n-grams scheme, a tweet's feature vector is constructed via examining the presence/absence of this scheme among the tweet's tokens. Thus, the feature vectors are formulated as one-hot encoding vectors with binary values "1" (presence) or "0" (absence). Term frequency (TF) property was employed to reduce the features size according to predefined frequency thresholds.

- N-gram embeddings: Based on word embeddings initialized randomly at the embedding layer of Tw-StAR Feedforward neural model, n-gram embeddings are produced by applying a composition function over a specific number of word embedding vectors. In our experiments, we used the additive composition function, known as Sum Of Word Embeddings (SOWE). While composing an n-gram embedding vector, by performing an element-wise sum over word embedding vectors, SOWE considers the co-occurrence information of the n-gram words and totally ignores the local word order.

3.3 Hate Speech Classification

Using the generated features a Naive Bayes (NB) classifier and a feedforward neural network model were trained:

- Naive Bayes model: with one-hot n-gram features, we used an NB classifier implemented as a multinomial NB decision rule together with binary-valued features.

- Feedforward neural network : this model was developed with the following layers:
  - Embeddings layer receives the n-grams generated for each input tweet and map their constituent words into their corresponding word dense representations. N-grams are produced by going through the tweet using a sliding window of a fixed size (N) such that each word of the tweet is considered. All the resulting n-grams (shingles) are then fed to the model with supervision information included where each n-gram is associated with 2-dimension labels HS [1,0] or NOT [0,1] that represent the polarity of the tweet from which the n-gram is derived.
  - Lambda layer composes n-gram embeddings by applying SOWE over the word embeddings resulting from the embedding layer.
  - Hidden layer introduces non-linear discriminating features to the model with Relu activation function.
  - Output layer is equipped with a softmax function to induce the estimated probabilities of each n-gram output label (HS/NOT). Considering the whole tweet, HS scores and NOT scores predicted for all n-grams of the tweet are summed, then each of which is divided by the number of n-grams, contained in a tweet, yielding two values for the potential HS and NOT scores of the tweet. The label of the tweet is, thus, decided according to the greater among these two values.

Lang.	Features	R.	F1	Acc.
English	uni+bi	0.85	0.87	0.89
	8-gram emb.	**0.98**	**0.94**	**0.95**
Spanish	uni+bi	**0.77**	**0.77**	**0.78**
	8-gram emb.	0.72	0.72	0.72

Table 1: Unigrams+bigrams (TF threshold=2) and 8-gram embeddings results of NB/neural models for train/dev sets.

L.	Team (F1 rank)	P.	R.	F1	Acc.
Eng.	saradhix (1)	**0.69**	**0.68**	**0.65**	**0.65**
	Panaetius (2)	0.59	0.59	0.57	0.57
	YunxiaDing (3)	0.64	0.603	0.55	0.56
	Tw-StAR (9)	0.54	0.53	0.5	0.54
Sp.	luiso.vega (1)	0.73	0.74	**0.73**	**0.73**
	francolq2 (2)	0.73	0.74	**0.73**	**0.73**
	gertner (3)	**0.75**	**0.75**	**0.73**	**0.73**
	Tw-StAR (20)	0.70	0.71	0.70	0.70

Table 2: Tw-StAR official Codalab ranking.

4 Results and Discussion

Having the data preprocessed and hate indicatives specified and tagged in both training and dev/test sets, two HS models were used.

The first model is an NB classifier from NLTK[§] trained with one-hot n-gram features. We generated three n-gram schemes: unigrams (uni), unigrams+bigrams (uni+bi) and unigrams+bigrams+trigrams (uni+bi+tri). NB was first trained using all n-gram features, then by a reduced number of features obtained via term frequency (TF) with two threshold: 2 and 3. Among several runs with various n-gram schemes and TF values, we adopted the best-performing scheme: uni+bi and TF threshold: 2.

The second model combines n-gram embeddings within a feedforward neural network. The window size 8 was, empirically, selected to produce 8-gram embeddings. Similarly, the embeddings dimension value was set to 100. For training, backpropagation algorithm and "Adam" optimizer (Kingma and Ba, 2014) were used.

Table 1 lists the results obtained using Train and Dev sets of English and Spanish tweets where the language, embeddings, average recall, average f-measure and accuracy are referred to as (Lang.), (emb.), (R.), (F1) and (Acc.), respectively.

Considering Table 1, both feature types performed well for HS detection in English. However, n-gram embeddings were better with an F1 of 94% against 87% scored by one-hot n-grams. We can explain that by the ability of n-gram embeddings to capture the semantic word regularities regardless of the local word order; which is appropriate to handle the informal English used on Twitter; where varying word orders can infer the same semantics (Iyyer et al., 2015).

Regarding the Spanish dataset, while the HS classification performances produced by both feature types were quite comparable, one-hot n-grams achieved slightly better results with an F1 77% and accuracy of 78% compared to 72% and 72% scored by n-gram embeddings, respectively. This could be attributed to the differences in vocabulary introduced by the different spoken varieties of Spanish found in the tweets (Maier and Gómez-Rodríguez, 2014). Hence, SOWE may miss the synonymous and semantic relations among such different words having same/close semantics which, in turn, leads to less expressive n-gram embeddings.

Having the best-performing features identified for English and Spanish, we adopted one-hot n-grams for Spanish and n-gram embeddings for English in the official submission. Table 2 lists the official results of Tw-StAR against the top three ranking systems where (L.), (Acc.), (Eng.), (Sp.), (R.) and (F1) refer to language, accuracy, English, Spanish, recall and f-measure, respectively.

Considering Table 1 and Table 2, we observe that Tw-StAR exhibit a robust performance for the Spanish dataset, while the evaluation measures degraded for the English dataset. We believe that, this could be attributed to the lack of homogeneity between the train/dev and test sets of English data.

5 Conclusion

We developed two HS detection models for multilingual tweets. With two feature types used, we investigated how likely n-gram embeddings can rival one-hot n-grams in HS detection. Upon training NB and a feedforward neural net with one-hot n-grams and n-gram embeddings, respectively, n-gram embeddings exhibited a better performance in English while the vocabulary differences in Spanish made n-gram embeddings less expressive. For future work, we aim to target HS in underrepresented languages such as Arabic and Turkish.

[§] https://www.nltk.org

References

Pinkesh Badjatiya, Shashank Gupta, Manish Gupta, and Vasudeva Varma. 2017. Deep learning for hate speech detection in tweets. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 759–760. International World Wide Web Conferences Steering Committee.

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 in twitter. In *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*. Association for Computational Linguistics.

Elisa Bassignana, Valerio Basile, and Viviana Patti. 2018. Hurtlex: A multilingual lexicon of words to hurt. In *5th Italian Conference on Computational Linguistics, CLiC-it 2018*, volume 2253, pages 1–6. CEUR-WS.

Chedi Bechikh-Ali, Hatem Haddad, and Yahya Slimani. 2019. Empirical evaluation of compounds indexing for turkish texts. *Computer Speech and Language*, 56(1):95–106.

Elisabetta Fersini, Paolo Rosso, and Maria Anzovino. 2018. Overview of the task on automatic misogyny identification at ibereval 2018.

Björn Gambäck and Utpal Kumar Sikdar. 2017. Using convolutional neural networks to classify hate-speech. In *Proceedings of the First Workshop on Abusive Language Online*, pages 85–90.

Sam Henry, Clint Cuffy, and Bridget T. McInnes. 2018. Vector representations of multi-word terms for semantic relatedness. *Journal of Biomedical Informatics*, 77:111 – 119.

Mohit Iyyer, Varun Manjunatha, Jordan Boyd-Graber, and Hal Daumé III. 2015. Deep unordered composition rivals syntactic methods for text classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 1681–1691.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *International Conference on Machine Learning*, pages 1188–1196.

Wolfgang Maier and Carlos Gómez-Rodríguez. 2014. Language variety identification in spanish tweets. In *Proceedings of the EMNLP'2014 Workshop on Language Technology for Closely Related Languages and Language Variants*, pages 25–35.

Mari J Matsuda. 2018. Public response to racist speech: Considering the victim's story. In *Words that wound*, pages 17–51. Routledge.

Hala Mulki, Chedi Bechikh Ali, Hatem Haddad, and Ismail Babaoglu. 2018a. Tw-star at semeval-2018 task 1: Preprocessing impact on multi-label emotion classification. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 167–171.

Hala Mulki, Hatem Haddad, Chedi Bechikh Ali, and Ismail Babaoğlu. 2018b. Tunisian dialect sentiment analysis: A natural language processing-based approach. *Computación y Sistemas*, 22(4).

Hala Mulki, Hatem Haddad, Mourad Gridach, and Ismail Babaoğlu. 2017. Tw-star at semeval-2017 task 4: Sentiment classification of arabic tweets. In *Proceedings of the 11th international workshop on semantic evaluation (SEMEVAL-2017)*, pages 664–669.

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*, pages 145–153. International World Wide Web Conferences Steering Committee.

Endang Wahyu Pamungkas, Alessandra Teresa Cignarella, Valerio Basile, Viviana Patti, et al. 2018. 14-exlab@ unito for ami at ibereval2018: Exploiting lexical knowledge for detecting misogyny in english and spanish tweets. In *3rd Workshop on Evaluation of Human Language Technologies for Iberian Languages, IberEval 2018*, volume 2150, pages 234–241. CEUR-WS.

Helmut Schmid. 1999. Improvements in part-of-speech tagging with an application to german. In *Natural language processing using very large corpora*, pages 13–25. Springer.

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*, pages 1–10, Valencia, Spain.

Elise Fehn Unsvåg and Björn Gambäck. 2018. The effects of user features on twitter hate speech detection. In *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, pages 75–85.

Zeerak Waseem and Dirk Hovy. 2016. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of the NAACL student research workshop*, pages 88–93.

Shuhan Yuan, Xintao Wu, and Yang Xiang. 2016. A two phase deep learning model for identifying discrimination from tweets. In *EDBT*, pages 696–697.