A Comparative Study of Embeddings Methods for Hate Speech Detection from Tweets

Shashank Gupta
IIIT-Hyderabad, India
shashank.gupta@research.iiit.ac.in

Zeerak Waseem
University of Sheffield, UK
zeerak.w@gmail.com

In this work, we review popular representation learning methods for the task of hate speech detection on Twitter data-. Methods in representation learning have been successfully applied to a variety of NLP tasks, including sentiment analysis (Tang et al., 2016), sarcasm detection (Joshi et al., 2016), and text similarity (Kenter et al., 2016). Specifically, for the task of hate speech detection, representation learning methods have shown promising results in comparison to traditional feature-based methods (Badjatiya et al., 2017; Nobata et al., 2016; Djuric et al., 2015). However, there has been no comprehensive study comparing the utility of embedding methods for hate speech detection. To fill this gap in literature, we study the effect of word and sentence embeddings methods for hate speech detection on publicly available data sets.

We use the following embedding methods:
Word Embeddings: 1) Word2vec pre-trained on Google News and Twitter corpus (Godin et al., 2015) respectively. **2)** GloVe , pre-trained on Gigaword corpus and tweet corpus respectively. **3)** Word2vec, trained on domain (hate) specific tweets[1]. **4)** Pretrained embeddings using Fasttext (Grave et al., 2017).
Sentence Embeddings: 5) Doc2vec, trained on hate speech corpus as described previously. **6)** Skip-thought, pre-trained on BookCorpus dataset. **7)** Tweet2vec (Dhingra et al., 2016), pre-trained on a large tweets corpus.
Datasets: We compare performance on three data sets collected on Twitter, that are annotated for hate speech (Waseem and Hovy, 2016; Waseem, 2016; Davidson et al., 2017) (see Table 2). For each data set, we collapse the annotations into a positive and negative classes, where all items annotated as 'hate speech' retain their annotation while all other labels are collapsed into "Neither"

0.1 Experimental Setup

We apply a Logistic Regression model and train each data set with an 80%/20% train and test split, respectively. We evaluate our model using AUC-score, F1-score, Precision, and Recall scores. In-domain word2vec embeddings are trained using gensim on a corpus of size 1 billion documents.

0.2 Results

In our experiments (see Table 1), we find that sentence level embeddings outperform sentence level methods on (Waseem and Hovy, 2016; Waseem, 2016) while domain specific word-level embeddings perform best on (Davidson et al., 2017).

Considering the pre-trained word-embeddings, we find that GloVe embeddings are outperformed by word2vec embeddings trained on Google News data sets and on Twitter. Of the pre-trained embeddings, FastText has the worst performance as it is based on simple bag-of-words model.

Further, we find that word embeddings trained on in-domain data outperform all other word-level embedding types as they can capture fine-grained characteristics of the domain. In contrast, the domain agnostic embeddings tend to capture more semantic properties. Concatenating domain-specific and domain-agnostic embeddings outperforms other embedding methods for 2 data sets, as these embeddings capture both domain specific characteristics and semantic properties.

Amongst sentence embedding methods, doc2vec performs poorly across all datasets. This is consistent with previous results (Nobata et al., 2016; Djuric et al., 2015) Skip-thought embeddings outperform all methods on the data set with the smallest class imbalance, suggesting that the model is well equipped to deal with balanced data sets of hate speech. In contrast, Tweet2vec outperforms all methods on the data set with the largest class

[1] Tweets were collected using the keywords as described by Davidson et. al (Davidson et al., 2017)

Methods	**Waseem-EMNLP (Waseem, 2016)**				**Waseem-NAACL (Waseem and Hovy, 2016)**				**Davidson-ICWSM (Davidson et al., 2017)**			
	AUC	**F1**	**R**	**P**	**AUC**	**F1**	**R**	**P**	**AUC**	**F1**	**R**	**P**
W2V-Google (300)	0.595	0.838	0.870	0.855	0.699	0.768	0.782	0.774	0.746	0.885	0.893	0.885
W2V-Twitter (400)	0.576	0.830	0.871	0.873	0.659	0.741	0.768	0.766	0.705	0.873	0.888	0.880
W2V-Hate (25)	0.567	0.821	0.859	0.830	0.627	0.709	0.738	0.723	0.758	0.882	0.887	0.880
W2V-Hate (50)	0.587	0.829	0.860	0.831	0.644	0.725	0.751	0.739	0.783	0.892	0.895	0.890
W2V-Hate (100)	0.618	0.841	0.864	0.840	0.668	0.744	0.764	0.754	0.811	0.902	0.904	0.901
W2V-Hate (200)	0.629	0.843	0.862	0.840	0.682	0.755	0.771	0.762	0.820	0.904	0.904	0.903
W2V-Hate (300)	0.638	0.844	0.860	0.839	0.679	0.750	0.766	0.755	**0.840**	**0.912**	**0.912**	**0.912**
Glove-Twitter (25)	0.546	0.809	0.852	0.811	0.622	0.704	0.732	0.714	0.720	0.870	0.879	0.868
Glove-Twitter (50)	0.554	0.812	0.851	0.811	0.630	0.710	0.736	0.719	0.750	0.880	0.886	0.878
Glove-Twitter (100)	0.561	0.816	0.853	0.816	0.639	0.717	0.739	0.723	0.772	0.887	0.891	0.885
Glove-Twitter (200)	0.581	0.824	0.856	0.824	0.656	0.731	0.751	0.737	0.789	0.893	0.895	0.891
Glove-Gigaword (300)	0.535	0.801	0.846	0.795	0.666	0.742	0.762	0.752	0.749	0.879	0.885	0.877
Fasttext (400)	0.584	0.829	0.862	0.835	0.663	0.739	0.759	0.748	0.747	0.884	0.892	0.884
W2V-Hate + W2v-Twitter (700)	**0.653**	**0.852**	**0.867**	**0.848**	0.686	0.755	0.770	0.760	0.799	0.899	0.902	0.898
W2V-Hate + W2v-Google (700)	0.648	0.852	0.868	0.849	**0.705**	**0.770**	**0.782**	**0.773**	0.793	0.899	0.902	0.897
Doc2vec (100)	0.502	0.784	0.851	0.799	0.498	0.564	0.686	0.530	0.500	0.759	0.834	0.862
Doc2vec (300)	0.502	0.784	0.852	0.874	0.501	0.568	0.688	0.584	0.500	0.758	0.834	0.695
Skip-thought (4600)	0.640	0.858	0.881	0.868	**0.731**	**0.792**	**0.802**	**0.797**	**0.756**	**0.892**	**0.900**	**0.893**
Tweet2vec (150)	**0.727**	**0.882**	**0.889**	0.880	0.503	0.688	0.778	0.689	0.500	0.748	0.826	0.683

Table 1: Comparison of embedding methods by the best embedding dimension for each embedding type.

Dataset	**#HS**	**#Not-HS**
Waseem-EMNLP (Waseem, 2016)	1059	5850
Waseem-NAACL (Waseem and Hovy, 2016)	5406	11501
Davidson-ICWSM (Davidson et al., 2017)	4163	20620

Table 2: Statistics of the datasets

imbalance and performs poorly on more balanced datasets, suggesting that it is well equipped to deal with highly imbalanced datasets for hate speech.

In this work, we present a comprehensive study of different representation learning methods on the task of hate speech detection from Twitter. We find that domain-agnostic word-embeddings perform slightly worse compared to domain-specific, though domain-specific are apt at dealing with class embeddings. Further, it is apparent that using domain-specific knowledge, whether it is stylistic knowledge in the form of embeddings learned on entire tweets or word embeddings capturing semantic use of words in tweets is apt for dealing with class imbalances.

References

Pinkesh Badjatiya, Shashank Gupta, Manish Gupta, and Vasudeva Varma. 2017. Deep learning for hate speech detection in tweets. In *WWW*, pages 759–760.

Thomas Davidson, Dana Warmsley, Michael W. Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *ICWSM*, pages 512–515.

Bhuwan Dhingra, Zhong Zhou, Dylan Fitzpatrick, Michael Muehl, and William W. Cohen. 2016. Tweet2vec: Character-based distributed representations for social media. In *ACL*.

Nemanja Djuric, Jing Zhou, Robin Morris, Mihajlo Grbovic, Vladan Radosavljevic, and Narayan Bhamidipati. 2015. Hate speech detection with comment embeddings. In *WWW*, pages 29–30.

Fréderic Godin, Baptist Vandersmissen, Wesley De Neve, and Rik Van de Walle. 2015. Multimedia lab@ acl w-nut ner shared task: named entity recognition for twitter microposts using distributed word representations. *ACL-IJCNLP*, 2015:146–153.

Edouard Grave, Tomas Mikolov, Armand Joulin, and Piotr Bojanowski. 2017. Bag of tricks for efficient text classification. In *EACL*, pages 427–431.

Aditya Joshi, Vaibhav Tripathi, Kevin Patel, Pushpak Bhattacharyya, and Mark James Carman. 2016. Are word embedding-based features useful for sarcasm detection? In *EMNLP*, pages 1006–1011.

Tom Kenter, Alexey Borisov, and Maarten de Rijke. 2016. Siamese CBOW: optimizing word embeddings for sentence representations. In *ACL*.

Chikashi Nobata, Joel Tetreault, Achint Thomas, Yashar Mehdad, and Yi Chang. 2016. Abusive language detection in online user content. In *WWW*, pages 145–153.

Duyu Tang, Furu Wei, Bing Qin, Nan Yang, Ting Liu, and Ming Zhou. 2016. Sentiment embeddings with applications to sentiment analysis. *TKDE*, 28(2):496–509.

Zeerak Waseem. 2016. Are you a racist or am i seeing things? annotator influence on hate speech detection on twitter. In *1st Workshop on Natural Language Processing and Computational Social Science, EMNLP*, pages 138–142.

Zeerak Waseem and Dirk Hovy. 2016. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *SRW@ HLT-NAACL*, pages 88–93.

A Supplemental Material