Literature Review: Hate Speech Detection

Shuo Zhang
janezhangshuo@hotmail.com

Zhexuan Xu
zhexuanx@gmail.com

1 General Problem Definition

With the fast growth in social media and the increasing propagation of abusive language, hate speech detection has drawn significant attention from governments, companies, and researchers. However, it faces a few challenges. First, diversity of hate speech makes it difficult to catch features for different types, such as racism, sexism, and profanity. Second, users proactively create variations of hate speech (e.g. "@$$") to thwart existing detection systems, which may require systems to update overtime, or algorithms to be able to learn subword information and make inference on a new word. Lastly, some hate speeches may not even use offensive or profane words and further require cultural and background knowledge, which are sometimes confused with sarcasm. These all push hate speech detection algorithms to develop from spotting bad words to deeper natural language understanding.

2 Concise Summaries of the Articles

2.1 Hateful Symbols or Hateful People? Predictive Features for Hate Speech Detection on Twitter (Waseem and Hovy, 2016)

First, the authors provided a dataset of 16k tweets annotated for hate speech, which was used by many later works on hate speech detection. The dataset contains totally 16,914 tweets, 3,383 of them were labeled as sexist, 1,972 as racist, and 11,559 for neither. While gender distribution in this dataset is heavily skewed towards men (men 50.08%, women 2.26%), majority of users cannot be identified (47.64%), which may impairs use of gender information as features.

Second, the final model given by this article was then referred by later papers as baseline model, which is character n-grams (of lengths up to 4) + logistic regression. Character n-grams outperforms word n-grams, and character n-gram matrices is far less sparse than the word n-gram matrices. Adding gender feature cannot significantly improve the performance. And using location or length feature is detrimental to the f1 scores.

2.2 Abusive Language Detection in Online User Content (Nobata et al., 2016)

The authors of this article developed and made public a new abusive language detection dataset from Yahoo! Finance and News. The abusive language in this article includes hate speech, profanity and derogatory, and it's a binary classification on abusive/non-abusive. For Finance dataset, there are 759,402 in total, and 53,516 of them are labeled as abusive. For News dataset, there are 1,390,774 in total, and 228,119 are abusive. For model, they applied a supervised classification technique (Vowpal Wabbit's regression model) with features coming from four types: (1) n-grams, both character level and token level; (2) linguistic features, such as number of hate blacklist words; (3) syntactic features to capture long-range dependencies between words; (4) distributional semantic features, both word embedding and comment embedding. The authors performed analysis on how each individual type of feature contributes to the detection, by running classification using only one type of feature, and the character n-grams has the highest f-scores. The authors also did analyses of how previously trained model performs over time, across different domains.

2.3 Automated Hate Speech Detection and the Problem of Offensive Language (Davidson et al., 2017)

In this paper, the authors separated abusive language into hate speech and only offensive language, and trained multi-class classifier to distinguish hate speech, only offensive language, and neither. They

defined hate speech as language that is used to expresses hatred towards a targeted group or is intended to be derogatory, to humiliate, or to insult the members of the group.

Under such settings, bag-of-words approaches tend to have high recall but lead to high false positives of hate speech. To better distinguish the categories, the authors tried different classification models on features including stemmed token n-grams, POS n-grams, as well as sentiment scores. Logistic regression with L1 regularization was applied for dimension reduction. It turned out Logistic Regression and Linear SVM perform better. However, there is still 31% of hate speeches were misclassified as offensive language. Positive sentiment and higher readability scores are more likely to belong to non-abusive class.

2.4 A Survey on Hate Speech Detection using Natural Language Processing (Schmidt and Wiegand, 2017)

In this paper, the authors presented a survey on the automatic detection of hate speech, focusing on features that have been explored in articles until 2017. For each type of features, the authors compared previous works and made instructive conclusions:

Fairly generic features, such as bag of words or embeddings, systematically yield reasonable classification performance. Character-level n-grams works better than token-level n-grams. Lexical resources, such as list of slurs, may help classification, but usually only in combination with other types of features. Linguistic features such as typed-dependency relationships, have also been shown to be effective. Information derived from text may not be the only cues. It may be complemented by meta-information (history about the speaker) or information from other modalities (e.g. images attached to messages). Knowledge-based features could also help with domain-specific hate speech.

2.5 Deep Learning for Hate Speech Detection in Tweets (Badjatiya et al., 2017)

In this paper, the hate speech detection is a three-class classification: sexist, racist, and neither, from 16K annotated tweets dataset. The authors experiment with multiple classifiers, which can be overall described as three types: (1) Baselines: embedding as features + classical machine learning classifier; (2) DNN using GloVe/random embedding; (3) Use the learned embedding from trained (fine-tuned) DNN as features + classical machine learning classifier. Figure 1 shows the comparison of various methods. Standard deviation for all methods varies from 0.01 to 0.025.

	Method	Prec	Recall	F1
Part A: Baselines	Char n-gram+Logistic Regression [6]	0.729	0.778	0.753
	TF-IDF+Balanced SVM	0.816	0.816	0.816
	TF-IDF+GBDT	0.819	0.807	0.813
	BoWV+Balanced SVM	0.791	0.788	0.789
	BoWV+GBDT	0.800	0.802	0.801
Part B: DNNs Only	CNN+Random Embedding	0.813	0.816	0.814
	CNN+GloVe	0.839	0.840	0.839
	FastText+Random Embedding	0.824	0.827	0.825
	FastText+GloVe	0.828	0.831	0.829
	LSTM+Random Embedding	0.805	0.804	0.804
	LSTM+GLoVe	0.807	0.809	0.808
Part C: DNNs + GBDT Classifier	CNN+GloVe+GBDT	0.864	0.864	0.864
	CNN+Random Embedding+GBDT	0.864	0.864	0.864
	FastText+GloVe+GBDT	0.853	0.854	0.853
	FastText+Random Embedding+GBDT	0.886	0.887	0.886
	LSTM+GloVe+GBDT	0.849	0.848	0.848
	LSTM+Random Embedding+GBDT	**0.930**	**0.930**	**0.930**

Figure 1: Comparison between methods

Using the embedding learnt from DNN + Gradient Boosted Decision Tree gives the best results. It's interesting that initialization with random embeddings is slightly better than initialization with GloVe embeddings when used along with GBDT. To verify that task-specific learned embedding outperforms general embeddings in hate speech detection, the author chose few words and showed top similar words from GloVe and DNN learned embeddings separately, and the latter clearly shows the "racist" or "sexist" bias for the chosen words.

2.6 Using Convolutional Neural Networks to Classify Hate-Speech (Gambäck and Sikdar, 2017)

In this paper, the classification has four target labels: racism, sexism, both (racism and sexism), and non-hate-speech. The authors applied CNN with four feature embeddings separately: (1) random vector; (2) word2vec; (3) character n-gram vector; (4) concatenating word2vec and character n-gram vector. The CNN model is shallow, and filters of different region sizes were utilized to extract information from windows of different lengths, as shown in Figure 2.

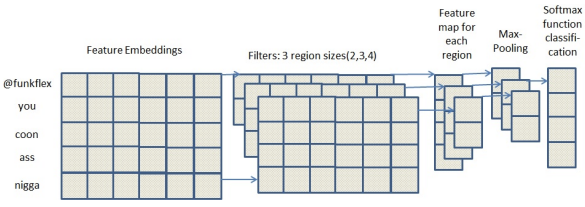

Figure 2: Illustration of CNN classifier

The word2vec embeddings performed the best. From the error analysis, the classifier did perform

well on 'racism' category and 'both' category, partly because the small amount of data in these two categories. And around 20% of 'sexism' category was misclassified as non-hate-speech. Overall, the model did beat the baseline model a lot, which is the character n-gram + logistic regression. This is probably because the CNN used here is too shallow to extract enough useful information.

2.7 Detecting Hate Speech on Twitter using a Convolution-GRU Based Deep Neural Network (Zhang et al., 2018)

In this paper, the authors introduced a deep neural network model combining CNN, GRU with drop-out and global max pooling layers to regularize learning for better performance. Intuitively, the CNN layer learns features similar to n-gram sequences while the later GRU layer learns sequence orders and context information. Figure 3 shows the model architecture. The drop-out layer right after embedding layer forces the classification not to rely on any individual words, and the global max pooling extracts features from the GRU layer. These two regularization layers were found to improve classification accuracy.

In terms of dataset and baselines, the authors conducted comparative evaluation of the largest collection of public datasets, and shown their model outperforms baselines as well as state-of-art for most datasets. They also created and published another hate speech dataset, focusing on Muslim (religion) and refugees.

For classical methods that depend on pre-engineered features, they also proved that there is often no need for sophisticated feature engineering, but using automatic feature selection techniques on generic features such as n-grams can produce better results.

2.8 Neural Word Decomposition Models for Abusive Language Detection (Bodapati et al., 2019)

This paper dived into the importance of using finer units such as character or subword units to learn robust representations for detecting abusive language. The author compared the effectiveness of end-to-end character-based models, word + character embedding models, byte pair encoding (BPE), subword models, with pure word-based models. Two main conclusions that could be helpful for future works are : (1) End-to-end character models (using very deep CNN) are not as effective as subword

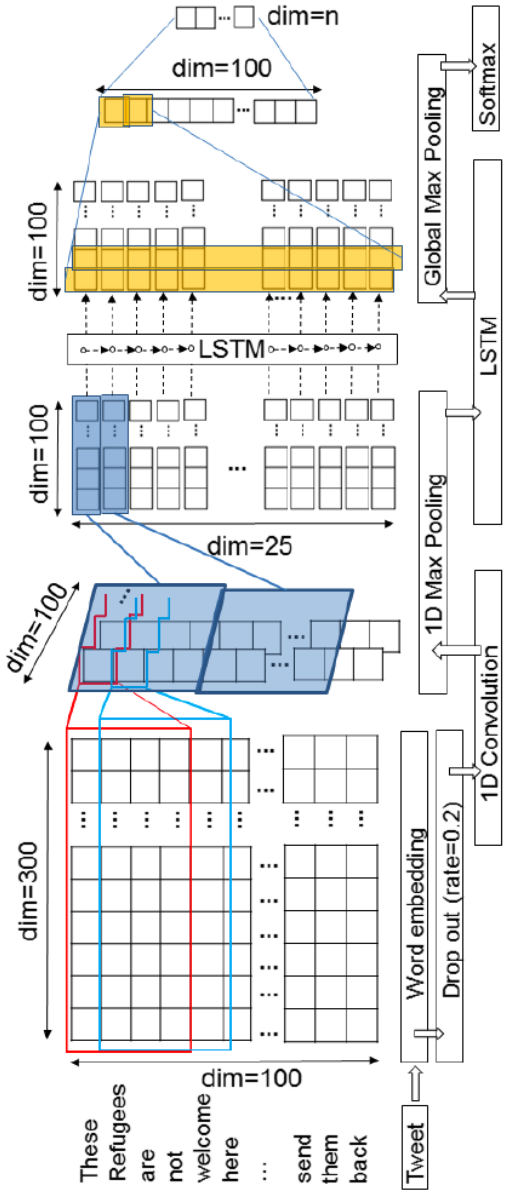

Figure 3: CNN + LSTM architecture

or word + character models, which demonstrates the basic fact that the knowledge of word leads to a powerful representation, and word boundary information is still informative in noisy settings; (2) Pretraining a BERT BPE model on a large general corpus like Wikipedia, and using this for encoding input text by splitting words has shown significant improvements for all the word based models.

3 Compare and Contrast

3.1 Terminology

The definition of "hate speech" is not exactly the same in all literature, and some may use similar terms like "offensive", "abusive" languages. The term "hate speech" was formally 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 characteristic" (Nockleby, 2000). Nobata *et al.* divided abusive language into three categories: hate speech as formally defined, derogatory as "language with attacks an individual or a group not based on above characteristic", and profanity as "language which contains sexual remarks or profanity" (Nobata et al., 2016). Davidson *et al.* divided the data into three categories as hate speech, only offensive language, and neither (Davidson et al., 2017). For articles using 16k annotated tweets dataset (Waseem and Hovy, 2016), they were specifically detecting racism and sexism, as their data labels are racism, sexism, and neither.

3.2 Datasets

There isn't a benchmark dataset for hate speech detection, and the large majority of existing works were evaluated on privately collected datasets, often for different problems. Figure 4 lists some publicly available hate speech datasets provided in above literatures, including papers that used them in "consumer" column.

Publisher	Source	Classes	Consumers
Waseem et al. 2016	Twitter	racism (1,972) sexism (3,383) neither (11,559)	Waseem et al. 2016 Badjatiya et al. 2017 Zhang et al. 2018 Bodapati et al. 2019
Waseem. 2016	Twitter	racism (91) sexism (946) both (18) neither (5,600)	Waseem. 2016 Gamback et al. 2017 Zhang et al. 2018
Nobata et al. 2016	Yahoo! Finance	abusive (53,516) non-abusive (705,886)	Nobata et al. 2016 Zhang et al. 2018
Nobata et al. 2016	Yahoo! News	abusive (228,119) non-abusive (1,162,655)	Nobata et al. 2016 Zhang et al. 2018
Zhang et al. 2018	Twitter	hate (414) Non-hate (2,021)	Zhang et al. 2018

Figure 4: Public datasets

Note that second dataset (Waseem, 2016) has different versions and the number keeps changing because of deletion.

3.3 Methods

In general, the methods can be divided into classic methods and deep learning based methods depending on whether there is an automated feature learning process.

3.3.1 Classic Methods

Features of classic methods are manually encoded into feature vectors. There are several types of features from above literature. (1) *Simple surface features* such as bag of words, character and word n-grams. Character n-grams has been proven to be more effective than word n-grams (Nobata et al., 2016) (Schmidt and Wiegand, 2017). (2) *Syntactic features* such as POS n-grams to capture long-range dependencies dependency (Nobata et al., 2016) (Davidson et al., 2017). (3) *Distributional semantic features*, such as word embedding and comment embedding, that learn low-dimensional dense feature vectors of word/sentence meaning, like GloVe and word2vec (Nobata et al., 2016) (Badjatiya et al., 2017). (4) *Lexical resources* containing lists of negative words. Early methods were heavily based on spotting bad words. (5) *Sentiment analysis* provides polarity information. Positive sentiment and higher readability scores are more likely to belong to non-abusive class (Davidson et al., 2017). (6) *Multimodal information* from images (Yang et al., 2019). (7) *Knowledge-Based features* to bring stereotypical concepts, which could help detecting domain-specific hate speech (Schmidt and Wiegand, 2017). (8) *Metadata* such as gender and history of users.

For classifier, Logistic Regression, SVM and Gradient Boosted Decision Tree often give the best performance. And a usual baseline model for many papers is character n-gram (up to length of 4) + Logistic Regression from Waseem's work in 2016 (Waseem and Hovy, 2016). Besides, feature selection such as Logistic Regression + L1 regularization can be a very powerful technique to improve performance of classic methods (Zhang et al., 2018).

3.3.2 Deep Learning Based Methods

Deep learning based methods use Neural Networks to learn abstract feature representations from input data to boost classification. Note some works use DNN to learn word or text embeddings as features and feed into another classifier like GBDT for classification. This is proven to be an effective way to train a task-specific embedding and get low-dimensional dense features (Badjatiya et al., 2017).

CNN is well known as an effective network for feature extraction, and when applied to NLP, it extracts character n-gram features or special phrases. RNN, especially Gated-RNN is powerful in modelling sequence input by learning long-range word or character dependencies. While some works applied CNN or LSTM independently (Badjatiya et al., 2017) (Gambäck and Sikdar, 2017), Zhang *et al.* started to combine CNN and GRU to utilize

the power of these two (Zhang et al., 2018). CNN layers was first applied to extract n-gram features, then the feature vectors will be max-pooled and fed into GRU while still keeping sequence order. Regularization techniques such as adding a drop-out layer right after the embedding layer, and applying global max pooling layer, were shown to improve the performance.

While most DNN methods focus on the structure of classifier, Bodapati *et al.* dived into the importance of using finer units of word to learn more robust representations in neural networks (Bodapati et al., 2019). They compared the effectiveness of end-to-end character-based models, word + character embedding models, byte pair encoding (BPE), subword models, with pure word-based models. It is shown that pretraining a BERT BPE model on a large general corpus then using this for encoding input text can improve significantly for all word-based models.

4 Future Work

Since previous work showed the effectiveness of combining CNN and RNN, it's natural to think about the powerful Transformer model, as Transformer was proven to be strong in both feature extractions and dealing with long term sequential information. It would be interesting to see whether Transformer can beat CNN + RNN models, or bi-directional RNN with multi-head attention. Also, due to the small size of dataset, we could try to use BERT to pretrain embeddings on a large corpus related to the dataset.

It would be very helpful to collect an even larger dataset for hate speech detection, including metadata like user information. A large benchmark dataset enables fare comparisons between algorithms.

Most works have so far focused on hate speech detection in English. Due to different culture background and probably different need from government, it would be a good application to detect hate speech for other languages.

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.

Sravan Babu Bodapati, Spandana Gella, Kasturi Bhattacharjee, and Yaser Al-Onaizan. 2019. Neural word decomposition models for abusive language detection. *arXiv preprint arXiv:1910.01043*.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Eleventh international aaai conference on web and social media*.

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.

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.

John T Nockleby. 2000. Hate speech. *Encyclopedia of the American constitution*, 3(2):1277–1279.

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*, pages 1–10.

Zeerak Waseem. 2016. Are you a racist or am i seeing things? annotator influence on hate speech detection on twitter. In *Proceedings of the first workshop on NLP and computational social science*, pages 138–142.

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.

Fan Yang, Xiaochang Peng, Gargi Ghosh, Reshef Shilon, Hao Ma, Eider Moore, and Goran Predovic. 2019. Exploring deep multimodal fusion of text and photo for hate speech classification. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 11–18.

Ziqi Zhang, David Robinson, and Jonathan Tepper. 2018. Detecting hate speech on twitter using a convolution-gru based deep neural network. In *European semantic web conference*, pages 745–760. Springer.