Hate Speech Detection on Twitter

Jainul N. Vaghasia
Department of Computer Science
University of Washington
Seattle, WA 98105
jnv3@cs.washington.edu

Abstract

We explore a neural network based method for classifying the presence of hate speech in a tweet. The network is based on word embeddings and mean pooling, and produces state-of-the art results on offline model. The method when combined with resampling of subset of past training data performs well under class imbalance in online setting. It is also speculated to be robust against concept drift in the stream.

1 Introduction

Cyberbullying has become a rather common incident in that Duggan and Smith [2013] found that, as of 2013, 73% of people had witnessed harassment online, and a full 40% of people had experienced harassment directly. Such online harassment often includes posts with sexually violent language, threats, hate speech and degrading racist terms. Being able to identify cases of toxic tweets can therefore help in tackling this problem at its core. To that end, we frame a binary classification problem: given a tweet, we wish to determine whether a tweet is offensive or not. There is no agreed upon definition of "offensive" owing to the ambiguities in human communication, but we use the following specific definition provided by Davidson et al. [2017]: "language that is used to express hatred towards a targeted group or is intended to be derogatory, to humiliate, or to insult the members of the group."

We then approach this problem with two different perspectives. First, we handle the tweets in an offline manner in which tweets are provided in a batch beforehand. Then we improve upon our model for online learning where tweets arrive at different time steps. In doing so, we address the class imbalance arising from the fact that offensive tweets are expected to be relatively fewer than non-offensive tweets (note the difference between witnessing an offensive tweet mentioned above and the presence of such tweet), and the concept drift that occurs when the distribution of such tweets changes based on socio-political events.

All code was written on basic libraries and is available here: `https://github.com/JainulV/Hate-Detection-Twitter`

2 Datasets

In this paper, we deal with two datasets to train and evaluate our classifier. First dataset is Davidson et al. [2017]'s publicaly available HATE dataset compiled by searching for tweets on `www.Hatebase.org`. Tweets are binarized as "hate speech", or "not". The other dataset is Golbeck et al. [2017]'s HAR harassment dataset, which identifies tweets as "harassing" or "not".[1] Retweets were removed from both datasets. Table 1 provides a numerical summary of both datasets.

[1] Available to researchers by emailing jgolbeck@umd.edu.

32nd Conference on Neural Information Processing Systems (NIPS 2018), Montréal, Canada.

Table 1: Dataset Summary

Name	Labels and Counts		Total
HATE	Hate Speech	Not Hate Speech	
	$1,430$	$23,353$	$24,783$
HAR	Harassing	Not Harassing	
	$5,285$	$15,075$	$20,360$

We also use 300 dimensional GloVe Common Crawl vector embeddings for translating words to vectors that are fed into our model (Pennington et al. [2014]).

3 Methods

3.1 Offline

Let us have $\{(x^t, y^t)\}_{t=1}^n$ denote the training set where x^t is some tweet and y^t is the corresponding class label. We modify the TWEM method (Kshirsagar et al. [2018]) to use fewer parameters in exchange of minimal drop in accuracy. First, we create 300 dimenional embeddings for each word in a given tweet x^t of word length T, so each word $\{w_i\}_{i=1}^T$ in x^t is mapped to $\{m_i\}_{i=1}^T \in \mathbb{R}^{300}$. Following this, we use mean pooling operation on $\{m_i\}_{i=1}^T$, which allows us to capture the overall context of the tweet. Denote the output from this operation as a. This representation a is then fed into a 50 node 2-layer MLP followed by ReLU activation to allow for nonlinear representation learning. This is the penultimate layer which passes to a fully connected softmax layer whose output is the probability distribution over the class labels.

3.2 Online

In order to adapt the above approach to online stream of tweets, we consider two queues of size L each, one holds the positive examples while the other holds the negative examples (Malialis et al. [2018]). Queue-based resampling stores the most recent example plus $2L - 1$ old ones. We will refer to the proposed algorithm as $Queue_L$. The union of the two queues is then taken at each time step to form the new training set for the classifier. The cost function is given by

$$J = \frac{1}{|q^t|} \sum_{i=1}^{|q^t|} l(y_i, h(x_i)), \tag{1}$$

where q^t is the union queue with $|q^t| \leq 2L$ and $(x_i, y_i) \in q^t$. At each time step, the classifier is updated once according to 1.

The effectiveness of this approach can be attributed to the following highly intuitive reasoning. Having two queues keeps track of examples from both classes. It allows the classifier to "remember" old data from both classes, which can be seen as a form of oversampling to counteract the imbalance since the queue with less frequent examples will have same examples resampled more times than their counterparts. At the same time, we also note that since these queues are of fixed length L, it allows the classifier to "forget" relatively old data and therefore act like a sliding window over the data stream. This forgetfulness is speculated to handle the concept drift present in the stream.

4 Experiments and Results

4.1 Offline

To train the neural network, we perform minimal preprocessing which includes tokenizing the data using Spacy and applying GloVe Common Crawl vector embeddings. Training is performed using gradient descent with ℓ_2 regularization to reduce overfitting. The regularization parameter was chosen to be 0.001 by cross validation on 80–20 training-validation split.

For comparing our model on HATE dataset, we borrow features engineered by Davidson et al. [2017] (which include part-of-speech ngrams, sentiment analysis, and Twitter specific features) and apply

Table 2: Offline F1 Results

Method	HATE	HAR
Logisitic Regression (Kshirsagar et al. [2018])	-	0.68
Logisitic Regression (Davidson et al. [2017])	0.93	-
Naive Bayes	0.82	-
GRU Text + Metadata (Founta et al. [2018])	0.89	-
Ours	0.94	0.70

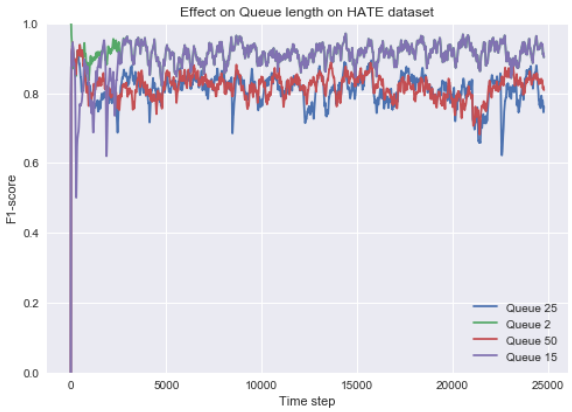

Figure 1: Prequential F1-Score against time step for different queue lengths

logistic regression. In addition, we use the statistics for Naive Bayes baseline and GRU model that uses metadata like popularity, network reciprocity and subscribed lists provided by Founta et al. [2018]. For HAR dataset, Kshirsagar et al. [2018] offer performance for a baseline model trained using logistic regression with character ngrams, word unigrams and TF*IDF.

The results are compiled in Table 2. It is evident that our model requires relatively fewer preprocessing steps and depends only on tweet text, which eliminates dependence on retrieving features like social graphs, sentiment readibility, etc. while performing better than these more complex approaches.

4.2 Online

We begin by defining the prequential F1-score as our metric as suggested in Gama et al. [2012] with a fading factor of $\alpha = 0.99$. At all time steps, we compute the prequential F1-score averaged over the most recent 30 runs.

We inspect the performance of our algorithm $Queue_L$ for $L = \{2, 15, 25, 50\}$ as shown in Figure 1. $Queue_2$ is the quickest to learn and performs the best followed by $Queue_{15}$ which eventually matches the performance of $Queue_2$ at about 3000th time step. We speculate a trade-off here that deals with concept drift. It is also seen that $Queue_{25}$ and $Queue_{50}$ lead to excessive oversampling which forces the classifier to "remember" old data for a long time, thereby affecting the decision boundary. Consequently, they are not able to perform as good as the former algorithms.

5 Planned work

As seen above, we achieve state-of-the-art results on offline model and perform well on online model with imbalanced data. Before the final paper submission, we expect to produce experimental evidence for the trade-off between concept drift and learning speed, or show that the given method fails under concept drift. In addition, the online model is currently trained and evaluated only on the HATE dataset. Time permitting, we will also train and evaluate online model on the HAR dataset.

References

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the 11th International AAAI Conference on Web and Social Media*, ICWSM '17, pages 512–515, 2017.

Maeve Duggan and Aaron Smith. Social media update 2013. *Pew Internet and American Life project*, 2013.

Antigoni-Maria Founta, Despoina Chatzakou, Nicolas Kourtellis, Jeremy Blackburn, Athena Vakali, and Ilias Leontiadis. A unified deep learning architecture for abuse detection, 2018.

João Gama, Raquel Sebastião, and Pedro Pereira Rodrigues. On evaluating stream learning algorithms. *Machine Learning*, 90:317–346, 2012.

Jennifer Golbeck, Zahra Ashktorab, Rashad O. Banjo, Alexandra Berlinger, Siddharth Bhagwan, Cody Buntain, Paul Cheakalos, Alicia A. Geller, Quint Gergory, Rajesh Kumar Gnanasekaran, Raja Rajan Gunasekaran, Kelly M. Hoffman, Jenny Hottle, Vichita Jienjitlert, Shivika Khare, Ryan Lau, Marianna J. Martindale, Shalmali Naik, Heather L. Nixon, Piyush Ramachandran, Kristine M. Rogers, Lisa Rogers, Meghna Sardana Sarin, Gaurav Shahane, Jayanee Thanki, Priyanka Vengataraman, Zijian Wan, and Derek Michael Wu. A large labeled corpus for online harassment research. In *Proceedings of the 2017 ACM on Web Science Conference*, WebSci '17, pages 229–233, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4896-6. doi: 10.1145/3091478.3091509. URL `http://doi.acm.org/10.1145/3091478.3091509`.

Rohan Kshirsagar, Tyus Cukuvac, Kathleen McKeown, and Susan McGregor. Predictive embeddings for hate speech detection on twitter. *arXiv preprint arXiv:1809.10644*, 2018.

Kleanthis Malialis, Christoforos Panayiotou, and Marios M. Polycarpou. Queue-based resampling for online class imbalance learning. In *ICANN*, 2018.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014. URL `http://www.aclweb.org/anthology/D14-1162`.