Efficient Large-scale Stance Detection in Tweets

Yilin Yan[1], Jonathan Chen[2], Mei-Ling Shyu[1,*]
[1]Department of Electrical and Computer Engineering
University of Miami, Coral Gables, FL 33124, USA
[2]Miami Palmetto Senior High School, Miami, FL 33156, USA
Emails: y.yan4@umiami.edu, jchen7760@gmail.com, shyu@miami.edu

ABSTRACT

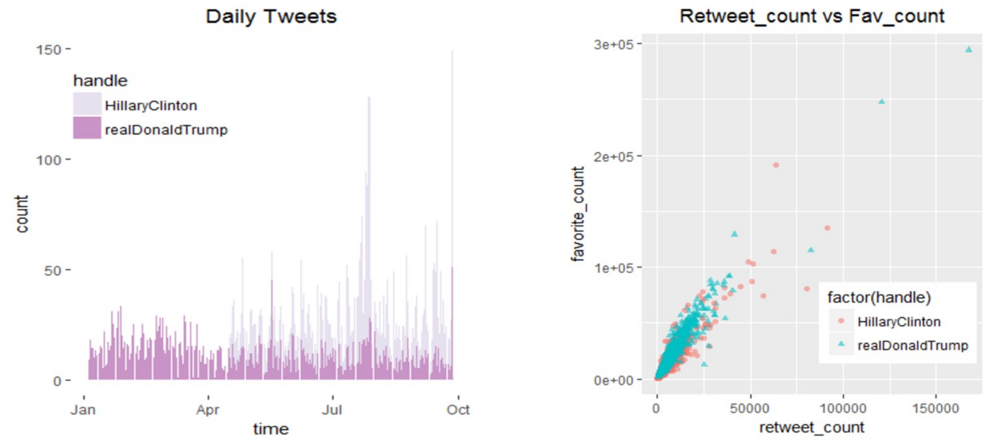
Stance detection is an important research direction which attempts to automatically determine the attitude (positive, negative, or neutral) of the author of a text (such as tweets) towards a target. Nowadays, a number of frameworks have been proposed using the deep learning techniques that show promising results in application domains such as automatic speech recognition, computer vision, as well as natural language processing (NLP). This paper shows a novel deep learning-based fast stance detection framework in bipolar affinities on Twitter. It is noted that millions of tweets regarding Clinton and Trump were produced per day on Twitter in the 2016 United States presidential election campaign, and thus it is used as a test use case because of its significant and unique counter-factual properties. In addition, stance detection can be utilized to imply the political tendency of the general public. Experimental results show that the proposed framework achieves high accuracy results when compared to several existing stance detection methods.

Keywords: Stance detection, opinion mining, deep learning, imbalanced data

Introduction

There has been a major growth in the use of multimedia data on the Internet in the past decade, especially the microblogging platforms. Microblogs, such as Twitter, Tumblr, Weibo, and Facebook, allow users to exchange small contents including images, short videos, as well as comments. Some previous research efforts were paid on these kinds of multimedia data (Yan et al., 2016; Chen, Hsieh, Yan, & Chen, 2015; Meng et al., 2014; Yan et al., 2014; Chen, Zhu, Lin, & Shyu, 2013; Lin, Chen, Shyu, & Chen, 2011; Zhu, Lin, Shyu, & Chen, 2011; Lin & Shyu, 2010; Chen, Rubin, Shyu, & Zhang, 2006; Chen, Shyu, & Kashyap, 2000). Twitter is one of the most widely used microblog platforms nowadays. As with traditional blogging, Twitter users ranging from regular users to politicians, celebrities, and company representatives post and interact with messages ranging from simple to the thematic post (known as “tweets”). As a result, it is possible to collect tweets of users from different social and interested groups for commercial or academic proposes. In 2016, the battle on Twitter was an integral part of a prearranged effort to disturb the past U.S. presidential election. To visualize the overall picture, Figure 1(a) presents the daily tweet counts between Hillary Clinton and Donald Trump for the election time period in 2016; while the popularity measure between the two candidates is shown in Figure 1(b) by mapping the retweet and favorite counts of the two candidates.

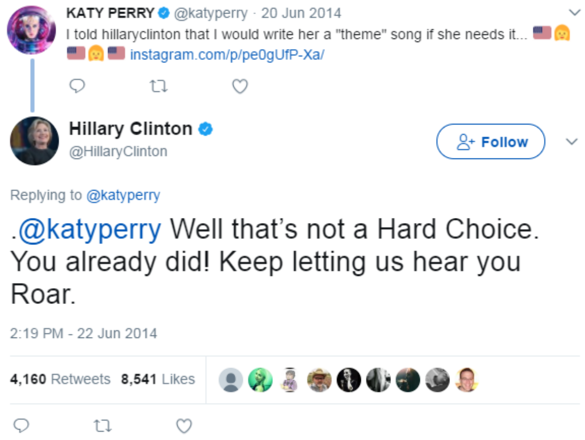
Generally speaking, stance detection can be formulated in different ways. In the scope of this paper, it is defined as automatically determining whether a tweet or a Twitter user tends to endorse the candidate of Democratic (Clinton) or Republican (Trump) Party. For instance, considering the candidate-tweet pair in Figure 2, the tweet from “Katy Perry” was in favor of Clinton obviously. Stance detection could be viewed as a subtask of opinion mining and it stands next to the sentiment analysis; while one significant difference is that the models determine whether a piece of a text is positive, negative, or neutral in sentiment analysis (Krejzl, Hourová, & Steinberger, 2017). However, our work is to determine a tweet or a Twitter user's favorability towards either candidate even when the candidate's name is not explicitly mentioned in the text. It is notable that the lack of evidence for 'favor' or 'against' does not necessarily imply that a tweet or a Twitter user is neutral towards the target, and it just means that we cannot deduce stance sometimes (Sobhani, Mohammad, & Kiritchenko, 2016).



(a) Daily tweets of the two candidates

(b) Popularity vs following measure of all supporters grouped by their candidates

Figure 1. Daily tweets versus Retweet and favorite counts



KATY PERRY @katyperry · 20 Jun 2014

I told hillaryclinton that I would write her a "theme" song if she needs it... 🇺🇸👱‍♀️🇺🇸👱‍♀️🇺🇸 instagram.com/p/pe0gUfP-Xa/

Hillary Clinton
@HillaryClinton

Replying to @katyperry

.@katyperry Well that's not a Hard Choice. You already did! Keep letting us hear you Roar.

2:19 PM - 22 Jun 2014

4,160 Retweets 8,541 Likes

Figure 2. Katy Perry shows her support for Clinton to the public on Twitter

One important research objective in stance detection is to imply the political tendency of the general public. By tweets from Twitter accounts, researchers can deduce whether a user is either in favor of or against a candidate. Thus, another target of this paper is to automatically infer the stances of Twitter users to see whether a user is more likely a Clinton or Trump supporter. While most election predictions reply on polls, automated stance detection can be applied to a much larger number of samples and bring complementary information to predict the election results.

The remaining of this paper is organized in the following manner. The next section discusses the related work in stance detection, followed by some recent progresses in deep learning on natural language processing (NLP). Next, the special presidential election Twitter dataset is introduced in details. Our affinity-based and deep learning-based stance detection frameworks are discussed in two separate sections, with their performance evaluated using the experimental results on tweets. The last section concludes the findings and develops the directions for future research.

Previous Work on Stance Detection

Sentiment analysis and stance detection are more and more popular nowadays with the recent rise of social media. Twitter is especially popular for research due to its scale, representativeness, variety of topics discussed, as well as ease of public access to content. SemEval has a special task for sentiment analysis in Twitter each year, and once specifically focused on detecting stance in tweets (Mohammad, Kiritchenko, Sobhani, Zhu, & Cherry, 2016). The task had two independent subtasks, i.e., supervised and weakly supervised. The supervised task comes with nearly three thousands of labeled training tweets towards the targets including “Atheism”, “Climate”, “Change is a Real Concern”, “Feminist Movement”, “Hillary Clinton”, and “Legalization of Abortion”. On the other hand, there were no labeled training data in the weakly supervised task. Participants were provided with around 70-thousand tweets related to “Donald Trump”, and the goal was to classify tweets into three classes, namely “favor”, “against”, and “none”. Traditional models such as n-grams and term frequency–inverse document frequency (TF-IDF) have been applied to this kind of tasks (Wang, Zhang, Liu, Lv, & Wang, 2014). A maximum entropy classifier based system that supports initial unigrams to initial trigrams with surface-level, sentiment, and domain-specific features is proposed (Krejzl, Hourová, & Steinberger, 2017). A stance detection system was proposed using a linear-kernel SVM classifier which relies on features drawn from the training samples, as well as those obtained from additional resources, e.g., sentiment features from lexicons and word-embedding features from external unlabeled data (Sobhani, Mohammad, & Kiritchenko, 2016).

Some earlier work ran experiments that used Twitter hashtags and emoticons such as #bestfeeling, #epicfail, and #news to identify positive, negative, and neutral tweets to train and analyze the sentiment of a tweet (Kouloumpis, Wilson, & Moore, 2011). The sentiments were identified as a powerful predictor in differentiating the behaviors of various accounts. Agarwal et al. (Agarwal, Xie, Vovsha, Rambow, & Passonneau, 2011) proposed a 3-way task of separating tweets into positive, negative, and neutral, and then used 3 models: unigram, feature-based, and tree kernel-based models to split the data. It was proposed in (Bollen, Mao, & Pepe, 2011) to use a psychometric instrument to classify six mood states including tension, depression, anger, vigor, fatigue, and confusion. The authors used aggregated Twitter content to compute a six-dimensional mood vector for each day in the timeline. One challenge in Twitter analysis is to identify and collect the right corpus that corresponds well to the domain and context of the tweets. This was attempted in (Pak & Paroubek, 2010) to focus and improve the corpus by an automatic collection and by using TreeTagger for POS-tagging. The wide scale effects of the socioeconomic events on the overall general mood of tweets were explored in (Bollen, Mao, & Pepe, 2011) over the longer periods of time. This provides a useful yardstick to track the sentiments, but this method does not solve the problem of context invariance.

While there are several existing frameworks on sentiment analysis and stance detection, they haven’t been used for election prediction based on our best knowledge. A hypothesis was proposed in (Robinson, 2017) that every non-hyperbolic tweet was from Trump’s staff while every hyperbolic tweet was from Trump himself. The researchers collected Trump’s tweets from Trump’s account including the “source” information and found out that most tweets were from either iPhones or Android phones. Their analysis showed that the iPhone and Android tweets were clearly from different people since tweets from them used different hashtags, retweeted in distinct ways, and were posted during different times. They also found that the iPhone tweets were less angry and more positive with benign announcements, while the Android tweets tended to be more negative with angry words. In (Stecanella, 2017), machine learning techniques were utilized to do sentiment analysis on candidates' Twitter mentions. They collected millions of tweets posted by users who discussed U.S. politics for Americans and non-Americans worldwide, and classified them based on their sentiments. Each posted tweet related to Clinton or Trump was labelled with either positive, neutral, or negative. The authors concluded that there were much more negative tweets about both candidates than positive tweets, while there were fewer tweets that mentioned Clinton than Trump. In (Jia, 2017), two groups of hashtags were defined arbitrarily, where each group was assumed to support Clinton or Trump, respectively. After that, the author used descriptive statistics methods and concluded that Trump's campaign knew more about how to use Twitter chat bots.

Recent Work on Deep Learning for Text Data

While there are several existing frameworks for multimedia data mining (Chiang et al., 2015; Zhu, Lin, Shyu, & Chen, 2010; Shyu, Xie, Chen, & Chen, 2008; Chen, Zhang, Chen, & Chen, 2005; Shyu, Haruechaiyasak, & Chen, 2003; Li, Chen, Shyu, & Furht, 2002; Chen, Shyu, Zhang, & Kashyap, 2001), text data analysis remains as a classical and important research area in NLP and machine learning with many applications. Currently, the most advanced and efficient way to handle this challenging situation would be to employ solutions from other domains of machine learning such as deep learning. With famous successful stories such as the AlphaGo Zero platform (DeepMind, 2017), deep learning algorithms have been applied across a wide variety of research domains including robotics, transportation prediction, autonomous driving, data compression, information retrieval, outlier detection, biomedicine, healthcare, disaster management, etc. (Yan, Chen, Sadiq, & Shyu, 2017; Yan & Shyu, 2016).

Specifically, deep learning-based frameworks have shown promising results in visual data processing (computer vision, multimedia data analysis, etc.), speech and audio processing (enhancement, recognition, etc.), NLP (sentence classification, translation, etc.), and social network analysis (Yan, Chen, Shyu, & Chen, 2015; Yan, Shyu, & Zhu, 2016). Deep learning was originally evolved from the concept of artificial neural networks (ANNs) but is capable of achieving much higher performance metrics than some existing competing machine learning methods. Several deep learning methods have sprouted from the initial ANN concepts such as the convolutional neural network (CNN), deep belief network (DBN), restricted Boltzman machine (RBM), deep neural network (DNN), and enhanced models (Pouyanfar et al., 2018; Swietojanski, Ghoshal, & Renals, 2014).

Recurrent neural networks (RNNs) have shown effective performance on processing sequential data, including text data on social media like Twitter. Therefore, RNNs are also important to solve the problems of sentiment analysis and stance detection. The gated RNN was proposed and applied to sentiment classification for texts (Tang, Qin, & Liu, 2015). The gated RNN takes the sentence vectors generated by either a CNN or long-short term memory (LSTM) network and combines them to form a document-level vector, which is used for classification. Augenstein et al. developed a neural network architecture based on conditional encoding in SemEval 2016 (Augenstein, Rocktäschel, Vlachos, & Bontcheva, 2016). They used an LSTM network to encode the target, followed by a second LSTM that encoded the tweet using the encoding of the target as its initial state. Similarly, another two-phase approach using attention embedding at each phase and encoding using the LSTM was proposed by Dey et al. for the same task (Dey, Shrivastava, & Kaushik, 2018). Their work was based on the observation that messages with neutral stances are usually non-subjective, while the ones with favor and against stances are usually subjective. Mohtarami et al. addressed the problem of automatic stance detection using a novel model based on end-to-end memory networks, which incorporated CNN, RNN, as well as a similarity matrix (Mohtarami, Baly, Glass, Nakov, Màrquez, & Moschitti, 2018).

Building the Twitter Dataset

The rise in popularity of social interacting websites such as Facebook, Twitter, and Snapchat has been challenged by the upsurge of unwelcomed and troubling bodies on these systems. This includes spam senders, malware systems, and other content contaminators. Before we start to introduce our Twitter dataset, it is notable that highly automated accounts with 450 tweets per day produced almost 18% of entire Twitter circulation in the 2016 U.S. presidential election. Since it is also observed that those disruptive systems called bots are inclined more towards circulating negative news than positive information, we apply a novel framework named Associative Affinity Factor Analysis (AAFA) designed for bot identification which can identify real people from bots in order to remove fake accounts before the stance analysis (Sadiq et al., 2017; Sadiq, Tao, Yan, & Shyu, 2017).

While stance detection models can have a positive social impact and are of practical interest to non-profits organizations and companies, researches in this direction were hindered by the unavailability of suitable datasets and lexicons for systematical training, development, and testing. In order to do stance analysis for the U.S. presidential election test use case, a dataset that includes the supporters of both sides is necessary. However, due to the privacy issues, it is nearly impossible to get the account names of the supporters. Luckily, Wikipedia provides the lists of Hillary Clinton and Donald Trump presidential campaign endorsements (Wikipedia, 2016). These lists include "big names" who have publicly claimed their endorsements for the office of the president of Clinton and Trump as presidential nominees. Since these supporters are notable individuals, the information was reliable and did not change much in the campaign.

For the collection of tweets, we visited Twitter.com and searched up the names of all the senators and representatives. Then, we looked for their official accounts, mostly by finding the accounts with a blue verification check on them. Upon finding the official accounts, we moved over to a data list, in which we input the Senators' or Representatives' account names and organized them among those that supported Hilary and those that supported Trump. After data cleaning, 310 supporters of Clinton and 412 supporters of Trump were included to build the experimental dataset. Then, the Twitter API was used to collect 3240 tweets from each supporter with time, resource, retweet, etc. In addition, we extracted the details of the supporters' accounts, cleaned the text data from all tweets, and mapped the truncated words to get the information of the hashtags.

Affinity-based Stance Detection

Consider each hashtag in a tweet as a concept and find the recurring itemset in Trump's retweets or comment feed. If we are able to find multiple instances of people continuously together based on the Association Affinity Network (AAN), then they are bots (Meng, Liu, Shyu, Yan, & Shu, 2014; Sadiq, Yan, Shyu, Chen, & Ishwaran, 2016; Yan, Chen, Sadiq, & Shyu, 2017; Shyu, Chen, & Kashyap, 2001). The confidence score is replaced by the average sentiment score. It was observed that bots usually have consistently positive or negative sentiments in their tweets. In addition, for real human supporters, individuals who endorsed Clinton tended to use different hashtags comparing to those supported Trump, and vice versa.

One attempt in the literature (Jia, 2017) built two groups of arbitrary hashtags, from their domain knowledge, to find the Clinton and Trump supporters. However, this approach lacks reproducibility and domain invariance. To overcome this challenge and find distinct hashtags, the log odds ratio approach was applied (Li, Li, & Zhang, 2008). For a hashtag *n*, we calculated $C_n^{Hillary}$ and C_n^{Trump} which represented the numbers of times *n* was used by the Clinton and Trump supporters, respectively. Similarly, $U_n^{Hillary}$ and U_n^{Trump} represented the numbers of distinct accounts of the Clinton and Trump supporters who used hashtag *n*. Next, the scores S_n^C and S_n^U were calculated to measure the likelihood values of a hashtag being associated with either of the candidates as shown in Equations (1) and (2).

$$S_n^C = log_2 \frac{\frac{C_n^{Hillary}+1}{\sum_{i=1}^{N} C_i^{Hillary}+1}}{\frac{C_n^{Trump}+1}{\sum_{i=1}^{N} C_i^{Trump}+1}} \tag{1}$$

$$S_n^U = log_2 \frac{\frac{U_n^{Hillary}+1}{\sum_{i=1}^{N} U_i^{Hillary}+1}}{\frac{U_n^{Trump}+1}{\sum_{i=1}^{N} U_i^{Trump}+1}} \tag{2}$$

Here, *N* refers to the total number of supporters. The scores and the ranked hashtags are given in Table 1. For comparison, the hashtag lists are shown in Table 2 by the domain knowledge (Jia, 2017) and the tweets from the candidates, i.e., Clinton and Trump (Stecanella, 2017). It is clear that some unique hashtags can only be automatically found using the proposed framework, e.g., CIR (Comprehensive Immigration Reform Act), RenewUi (federal unemployment extension), RestoreTheVRA (Voting Rights Act), Dobbs (Lou Dobbs), PJNET (Patriot Journalist Network), and VAWA (Violence Against Women Act).

Table 1. Ranked hashtags based on the proposed framework (case insensitive).

Rank	# of hashtags used		# of distinct accounts that use a hashtag	
	Hillary Clinton	**Donald Trump**	**Hillary Clinton**	**Donald Trump**
1	*CIR*	*Dobbs*	*RaiseTheWage*	*TrumpPence16*
2	*RenewUi*	*TrumpPence16*	*HoldTheFloor*	*CrookedHillary*
3	*RaiseTheWage*	*PJNET*	*RestoreTheVRA*	*WakeUpAmerica*
4	*ActOnClimate*	*WakeUpAmerica*	*VAWA*	*PJNET*
5	*WomenSucceed*	*TrumpTrain*	*MarriageEquality*	*VoteTrump*
6	*DoYourJob*	*AmericaFirst*	*WorldAidsDay*	*Jesus*
7	*RestoreTheVRA*	*ProLife*	*GunViolence*	*TrumpRally*
8	*DisarmHate*	*TeaParty*	*ProtectOurCare*	*Hannity*
9	*TimeIsNow*	*MakeAmericaGreatAgain*	*StopGunViolence*	*Trump45*
10	*GetCovered*	*ConfirmGorsuch*	*LoveIsLove*	*TrumpPence2016*

Table 2. Ranked hashtags based on domain knowledge and tweets from the candidates (case insensitive).

Rank	Domain knowledge		Candidate tweets	
	Hillary Clinton	**Donald Trump**	**Hillary Clinton**	**Donald Trump**
1	*votehillary2016*	*MAGA*	*DemsInPhilly*	*Trump*
2	*VoteHillary*	*HillarysBigotry*	*RNCinCLE*	*MakeAmericaGreatAgain*
3	*NeverTrump*	*CrookedHillary*	*DebateNight*	*VoteTrump*
4	*IAmWithHer*	*Hillary4Prison*	*debatenight*	*AmericaFirst*
5	*WeAreWithHer*	*NeverHillary*	*TBT*	*MAGA*
6	*NoTrump*	*TrumpTrain*	*NBCNewsForum*	*ImWithYou*
7	*TrumpLies*	*VoteTrump*	*DemConvention*	*TrumpTrain*
8	*StopTrump*	*LockHerUp*	*WomanCard*	*TrumpPence*
9	*DumpTrump*	*WakeUpAmerica*	*EstoyConElla*	*FITN*
10	*TrumpUnfit*	*TrumpsArmy*	*LoveTrumpsHate*	*GOPDebate*

Results of Affinity-based Stance Detection

To apply the affinity-based stance detection method, the hashtags of people supporting Clinton and Trump were extracted. The associative affinity was evaluated for one-itemset and two-itemset hashtags occurring in their tweets. The final ranking of the predictive hashtags was ranked according to an empirically selected threshold, i.e., S_n^C or S_n^U greater than a reasonable value. Each hashtag itemset has a dynamic threshold but the hashtags with the highest affinities were selected. Table 1 illustrates these case insensitive ranked hashtags for the two candidates. The final hashtag lists were selected based on both the "number of a hashtag being used" and the "number of distinct accounts that use a hashtag". The overlapped hashtags were cleaned and finally a list of 128 hashtags was created to generate the feature vectors for the Clinton and Trump supporters. Based on the number of a hashtag used, a feature vector with 128-dimension was generated and normalized for each account.

Table 3. Result comparison for different classifiers.

Classifier	F-score	Accuracy
DAC	0.6190	74.9%
Linear Regression	0.6853	78.1%
Logistic Regression	0.7061	78.7%
SVM	0.7128	80.9%
Proposed	**0.8156**	**84.9%**

For comparison, our stance detection model is evaluated against several popular classifiers including discriminant analysis classifier (DAC), linear regression, logistic regression (Meng & Shyu, 2012), as well as support vector machine (SVM) (Suykens & Vandewalle, 1999). As shown in Table 3, an average accuracy of about 85% is obtained by the proposed framework without any domain knowledge and polls. Using the novel feature vectors generated in the previous section and random forest (Breiman, 2001), our proposed framework performs the best for this task due to the nature of our feature vectors, i.e., different weights for the hashtags. We could even achieve higher F-score as well as accuracy value by removing some testing instances whose numbers of hashtags in Table 1 are lower than a certain threshold, since many people may refuse to disclose their endorsements in polls. Nevertheless, we would rather prefer not to remove any data instance since the size of the dataset used (the number of accounts included) is already small. Another reason is that we can apply a classifier fusion model to further improve the performance which will be explained in details in the following section.

Deep Learning-based Stance Detection

With the enormous amounts of data collected worldwide every day, the most non-trivial tasks in training a deep neural network is the training part apparently (Schmidhuber, 2015). While GPUs are well-known for fast training, how to build an efficient deep learning model remains a big problem which is in fact the main target of this paper. By applying a TextCNN model (Kim, 2014), this stance detection task can also be solved by deep learning. After several common pre-processing steps, a TextCNN model for this Twitter dataset is built for binary classification. The first layer in our model is the embedding layer, which maps vocabulary word indices into low-dimensional vector representations, i.e., word2vec vectors (Yan, 2018). It is essentially a lookup table that we learn from the data. The next layer performs convolutions over the embedded word vectors using the filters of different sizes. In this paper, it slides over 3 to 9 words at a time. Finally, we max-pool the result of the convolutional layer into a fixed-length feature vector, add dropout regularization (0.5), as well as classify the result using a softmax layer. With an additional max pooling layer to fuse the results from whatever number of tweets in the final stage, the owner of a Twitter account can be labelled as a Clinton or Trump supporter.

Table 4. Confusion matrix for stance analysis by deep learning trained on raw text data.

Predicted label Ground truth	Clinton supporter	Trump supporter
Clinton supporter	**97.1%**	2.9%
Trump supporter	15.3%	**84.7%**
Total	**90.0%**	

The experimental results are provided in Table 4 with the 90.0% accuracy. Please note that the total accuracy is not equal to the average of the accuracy values on Clinton side and Trump side since they have different numbers of samples. Nevertheless, there are more than 1 million tweets in this dataset per candidate, which makes the computational complexity in the training procedure very high, in terms of both time and space. Additionally, many tweets are irrelevant with politics. For instance, most tweets of Katy Perry, the biggest celebrity supporter of Clinton, are regarding music. The training effort paid on those tweets are therefore useless and may potentially hurt the model.

Using the results from the previous section, only those tweets with the list of 128 hashtags are used to build the TextCNN model and thus 90% training time can be saved. For a few numbers of the accounts that do not use any of the selected hashtags, they are removed from the dataset for convenience. The accuracy on the filtered Twitter data is 88.8% as shown in Table 5, which is just 1.2 percent lower than the results on the raw text data. Since only 10 percent of raw data (after filtering) are used for training in this part, the confusion matrix on the same testing data in Table 5 looks different from the one in Table 4.

Table 5. Confusion matrix for stance analysis by deep learning trained on filtered data.

Predicted label Ground truth	Clinton supporter	Trump supporter
Clinton supporter	**78.6%**	21.4%
Trump supporter	3.6%	**96.4%**
Total	**88.8%**	

Table 6. Confusion matrix for affinity-based stance detection on validation set.

Predicted label Ground truth	Clinton supporter	Trump supporter
Clinton supporter	**70.9%**	29.1%
Trump supporter	1.4%	**98.6%**
Total	**86.7%**	

Furthermore, the time complexity of the algorithm in the previous section is clearly less than the model built using deep learning. Using a novel classifier fusion framework by the so-called “judgers” (Yan, Zhu, Shyu, & Chen, 2016), we find that the affinity-based method is a good classifier for the Clinton supporters by the confusion matrix on the validation set in Table 6. Note that the accuracy value on the Clinton side is 98% (i.e., 70.9/(70.9+1.4)), which means it is an excellent “judger” for the Clinton supporters. By applying the classifier ensemble framework to a validation set, we get 98% accuracy on the Clinton supporters, which means if the affinity-based classifier considers that a testing Twitter account supports Clinton, the owner is highly possible a real Clinton supporter (Yan, Zhu, Shyu, & Chen, 2016). Also, if the affinity-based method labels the owner of a Twitter account as a Clinton supporter, it does not need to go through the testing stage in the deep learning-based model.

The final experiments are conducted using the aforementioned models and the results from the affinity-based and deep learning-based classifiers can be thus integrated together. The final results which prove the efficiency of the overall framework for this application are shown in Table 7. Since the time complexity of the affinity-based method is much less than the deep learning one, 50% of the testing time could be saved approximately (assuming that Clinton has a 50% favorable rating).

Table 7. Accuracy comparisons on Twitter dataset.

Classifier	Accuracy
Affinity-based (Table 3)	84.9%
Deep learning-based (trained on raw text data)	90.0%
Deep learning-based (trained on filtered data)	88.8%
Proposed framework	**90.8%**

Proof of the Efficiency of the Stance Detection Framework

To show the effectiveness of the proposed framework, a mathematical proof is given below. Let $\hat{p}$ be the favorable rating of Clinton which is unknown, p be the rating in a sample set, and $\hat{\sigma}$ be the estimated favorability. For a two-candidate election poll, based on the central limit theorem (CLT), if a polling organization samples n adults, the 95% confidence interval is:

$$[\hat{p}-2\times\hat{\sigma},\hat{p}+2\times\hat{\sigma}];\ \text{ where }\ \hat{\sigma}=\sqrt{\frac{\hat{p}(1-\hat{p})}{n}} \qquad (3)$$

Let d be the sampling error, i.e., the radius of the confidence interval. Since $\hat{p}\in[0,1]$, $\hat{p}\times(1-\hat{p})\leq 0.25$. Therefore, we have:

$$1.96\times\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}\leq 1.96\times\sqrt{\frac{1}{4n}}\leq d; so\ n\geq\frac{1.96^2}{4d^2} \qquad (4)$$

Thus, if $d=0.03$ or 3%, $n\geq 1067.11\approx 1068$; and if $d=0.01$ or 1%, $n\geq 9604$. Based on the law of large numbers (LLN), $\lim_{n\to\infty}\hat{p}=p$ while there is a trade-off between the sampling error and the number of samples. Therefore, a polling company often only samples one thousand adults. Even if the company can afford the cost to get ten thousand samples, the time cost for the poll would be too long. Meanwhile, 3% is a big number in many swing states, especially when Clinton swamps Trump in popular vote for just 2.1%.

Using the proposed framework, assume the classification accuracy is θ and n is big enough. $p$actually follows a Beta distribution as shown in Equations (5) and (6). If $\hat{p}=50\%$, $\theta=90.8\%$, and $n=1000$, a high kurtosis probability density function (PDF) of the Beta distribution is drawn in Figure 3 and the corresponding cumulative distribution function (CDF) is drawn in Figure 4(a). Figure 4(b) shows the CDF when n equals one million, where p has a 99.9999% probability falling into the interval of $50\pm 0.3\%$. The model designed in this Section is able to process hundreds of testing samples per minute and thus handling one million samples becomes possible. All in all, the proposed framework in this application could save 90% of the training time and 50% of the testing time and generate better classification results.

$$B_{pdf}=\frac{[\theta\times p+(1-\theta)\times(1-p)]^{\hat{p}\times n}[(1-\theta)\times p+\theta\times(1-p)]^{(1-\hat{p})\times n}}{\int_0^1[\theta\times\bar{p}+(1-\theta)\times(1-\bar{p})]^{\hat{p}\times n}[(1-\theta)\times\bar{p}+\theta\times(1-\bar{p})]^{(1-\hat{p})\times n}d\bar{p}} \qquad (5)$$

$$Beta_{pdf}(x;\alpha,\beta) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{\int_0^1 \bar{x}^{\alpha-1}(1-\bar{x})^{\beta-1}d\bar{x}}; \tag{6}$$

$$where\ x = 1 - \theta - p + 2\theta p, \alpha = n\hat{p} + 1, \beta = n - n\hat{p} + 1$$

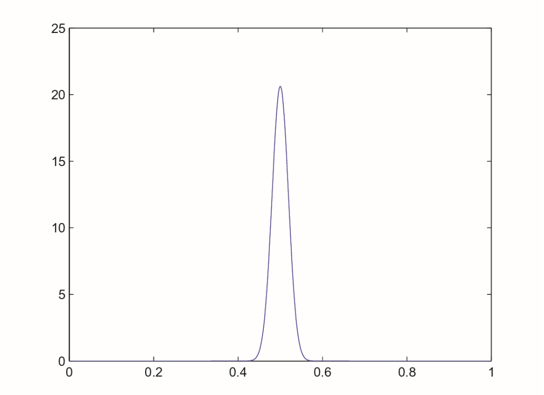
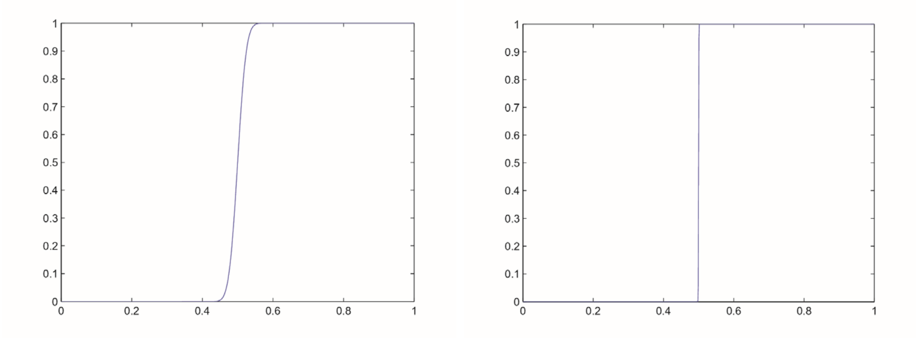


Figure 3. Probability density function when n=1000



(a) n=1000 *(b) n=1000000*

Figure 4. Cumulative density function

Conclusions and Future Work

The power of propaganda is reinforced when a limited number of individuals believe that it is prevalent. This paper proposes a novel deep learning-based stance detection framework in bipolar affinities on Twitter. The Twitter data in the 2016 United States presidential election campaign is used as a test use case to detect the stance between the followers of the two dominant presidential candidates, i.e., Hillary Clinton and Donald Trump. Based on our best knowledge, this is the first attempt to use machine learning and deep learning algorithms for stance detection in election predictions. The experimental results demonstrate the effectiveness of the proposed framework; while the mathematical proof shows the reliability of the proposed model. In the future, other information in tweets including the resources, retweets, favorites, etc. would be also considered for better stance detection.

REFERENCES

Agarwal, A., Xie, B., Vovsha, I., Rambow, O., & Passonneau, R. (2011). Sentiment analysis of twitter data. In Proceedings of the Workshop on Languages in Social Media (pp. 30–38).

Augenstein, I., Rocktäschel, T., Vlachos, A., & Bontcheva, K. (2016). Stance detection with bidirectional conditional encoding. arXiv preprint arXiv:1606.05464.

Bollen, J., Mao, H., & Pepe, A., (2011). Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena. International AAAI Conference on Weblogs and Social Media, 11 (pp. 450– 453).

Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5–32.

Chen, L. C., Hsieh, J. W., Yan, Y., & Chen, D. Y. (2015). Vehicle make and model recognition using sparse representation and symmetrical SURFs. Pattern Recognition, 48(6), 1979-1998.

Chen, S.-C., Rubin, S., Shyu, M.-L., & Zhang, C. (2006). A dynamic user concept pattern learning framework for content-based image retrieval. IEEE Transactions on Systems, Man, and Cybernetics: Part C, 36(6), 772-783.

Chen, S.-C., Shyu, M.-L., & Kashyap, R. L. (2000). Augmented transition network as a semantic model for video data. International Journal of Networking and Information Systems, 3(1), 9-25.

Chen, S.-C., Shyu, M.-L., Zhang, C., & Kashyap, R. L. (2001). Identifying overlapped objects for video indexing and modeling in multimedia database systems. International Journal on Artificial Intelligence Tools, 10(4), 715–734.

Chen, S.-C., Sista, S., Shyu, M.-L., & Kashyap, R. L. (1999). Augmented transition networks as video browsing models for multimedia databases and multimedia information systems. In Proceedings of the 11th IEEE International Conference on Tools with Artificial Intelligence (pp. 175–182).

Chen, C., Zhu, Q., Lin, L., & Shyu, M.-L. (2013). Web media semantic concept retrieval via tag removal and model fusion. ACM Transactions on Intelligent Systems and Technology, 4(4), 1-22.

Chen, X., Xiang, S., Liu, C.-L., & Pan, C.-H. (2014). Vehicle detection in satellite images by hybrid deep convolutional neural networks. IEEE Geoscience and Remote Sensing Letters, 11(10), 1797–1801.

Chen, X., Zhang, C., Chen, S.-C., & Chen, M. (2005). A latent semantic indexing based method for solving multiple instance learning problem in region-based image retrieval. In Proceedings of the 7th IEEE International Symposium on Multimedia (pp. 37–44).

Chiang, H. F., Hsieh, J. W., Chuang, C. H., Chuang, K. T., & Yan, Y. (2015). Modeling and recognizing action contexts in persons using sparse representation. Journal of Visual Communication and Image Representation, 30, 252-265.

DeepMind (2017). AlphaGo Zero. Retrieved from https://deepmind.com/blog/alphago-zero-learning-scratch.

Dey, K., Shrivastava, R., & Kaushik, S. (2018). Topical Stance Detection for Twitter: A Two-Phase LSTM Model Using Attention. In European Conference on Information Retrieval (pp. 529-536). Springer, Cham.

Jia, Y. (2017). Trump vs Hillary on twitter (Online technical report). Retrieved from http://yiyujia.blogspot.com/2016/09/trump-vs-hillary-on-twitter.

Kim, Y. (2014). Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882.

Kouloumpis, E., Wilson, T., & Moore, J. D. (2011). Twitter sentiment analysis: The good the bad and the omg!. In Proceedings of the 5th International AAAI Conference on Weblogs and Social Media, 11 (pp. 538-541).

Krejzl, P., Hourová, B., & Steinberger, J. (2017). Stance detection in online discussions. arXiv preprint arXiv:1701.00504.

Li, T., Li, F., & Zhang, X. (2008). Prediction of kinase-specific phosphorylation sites with sequence features by a log-odds ratio approach. Proteins: Structure, Function, and Bioinformatics, 70(2), 404–414.

Li, X., Chen, S.-C., Shyu, M.-L., & Furht, B. (2002). An effective content-based visual image retrieval system. In Proceedings of the Computer Software and Applications Conference (pp. 914–919).

Lin, L., Chen, C., Shyu, M.-L., & Chen, S.-C. (2011). Weighted subspace filtering and ranking algorithms for video concept retrieval. IEEE Multimedia, 18(3), 32-43.

Lin, L. & Shyu, M.-L. (2010). Weighted association rule mining for video semantic detection. International Journal of Multimedia Data Engineering and Management, 1(1), 37-5.

Liu, D., Yan, Y., Shyu, M.-L., Zhao, G., & Chen, M. (2015). Spatio-temporal analysis for human action detection and recognition in uncontrolled environments. International Journal of Multimedia Data Engineering and Management, 6(1), 1–18.

Mao, Q., Dong, M., Huang, Z., & Zhan, Y. (2014). Learning salient features for speech emotion recognition using convolutional neural networks. IEEE Transactions on Multimedia, 16(8), 2203–2213.

Meng, T., Liu, Y., Shyu, M.-L., Yan, Y., & Shu, C.-M. (2014). Enhancing multimedia semantic concept mining and retrieval by incorporating negative correlations. In Proceedings of the IEEE International Conference on Semantic Computing (pp. 28–35).

Meng, T. & Shyu, M.-L. (2012). Leveraging concept association network for multimedia rare concept mining and retrieval. In Proceedings of the IEEE International Conference on Multimedia and Expo (pp. 860– 865).

Mohammad, S., Kiritchenko, S., Sobhani, P., Zhu, X., & Cherry, C. (2016). Semeval-2016 task 6: Detecting stance in tweets. In Proceedings of the 10th International Workshop on Semantic Evaluation (pp. 31-41).

Mohtarami, M., Baly, R., Glass, J., Nakov, P., Màrquez, L., & Moschitti, A. (2018). Automatic Stance Detection Using End-to-End Memory Networks. arXiv preprint arXiv:1804.07581.

Pak, A. & Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. Proceedings of the Seventh conference on International Language Resources and Evaluation, 10 (pp. 1320-1326).

Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M. P., Shyu, M.-L., Chen, S.-C., & Iyengar, S. S. (2018). A Survey on Deep Learning: Algorithms, Techniques, and Applications. ACM Computing Surveys (CSUR), 51(5), 92.

Robinson, D. (2017). Text analysis of Trump's tweets confirms he writes only the (angrier) android half (Online technical report). Retrieved from http://varianceexplained.org/r/trump-tweets.

Sadiq, S., Tao, Y., Yan, Y., & Shyu, M.-L. (2017). Mining anomalies in medicare big data using patient rule induction method. In Proceedings of the IEEE Third International Conference on Multimedia Big Data (pp. 185–192).

Sadiq, S., Yan, Y., Shyu, M.-L., Chen, S.-C., & Ishwaran, H. (2016). Enhancing multi- media imbalanced concept detection using vimp in random forests. In Proceedings of the IEEE 17th International Conference on

Information Reuse and Integration (pp. 601–608).

Sadiq, S., Yan, Y., Taylor, A., Shyu, M.-L., Chen, S.-C., & Feaster, D. (2017). AAFA: Associative affinity factor analysis for bot detection and stance classification in twitter. In Proceedings of the IEEE 18th International Conference on Information Reuse and Integration (pp. 356–365).

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. Neural networks, 61, 85-117.

Shyu, M.-L., Chen, S.-C., & Kashyap, R. L. (2001). Generalized affinity-based association rule mining for multimedia database queries. Knowledge and Information Systems (KAIS): An International Journal, 3(3), 319–337.

Shyu, M.-L., Haruechaiyasak, C., & Chen, S.-C. (2003). Category cluster discovery from distributed WWW directories. Information Sciences, 155(3), 181–197.

Shyu, M.-L., Xie, Z., Chen, M., & Chen, S.-C. (2008). Video semantic event/concept detection using a subspace-based multimedia data mining framework. IEEE Transactions on Multimedia, 10(2), 252–259.

Sobhani, P., Mohammad, S., & Kiritchenko, S. (2016). Detecting stance in tweets and analyzing its interaction with sentiment. In Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics (pp. 159-169).

Stecanella, B. (2017). Donald Trump vs Hillary Clinton: sentiment analysis on twitter mentions (Online technical report). Retrieved from https://monkeylearn.com/blog/trump-vs-hillary-sentiment-analysis-twitter-mentions.

Swietojanski, P., Ghoshal, A., & Renals, S. (2014). Convolutional neural networks for distant speech recognition. IEEE Signal Processing Letters, 21(9), 1120–1124.

Suykens, J. A. & Vandewalle, J. (1999). Least squares support vector machine classifiers. Neural Processing Letters, 9(3), 293–300.

Tang, D., Qin, B., & Liu, T. (2015). Document modeling with gated recurrent neural network for sentiment classification. In Proceedings of the 2015 conference on empirical methods in natural language processing (pp. 1422-1432).

Wang, D., Zhang, H., Liu, R., Lv, W., & Wang, D. (2014). t-Test feature selection approach based on term frequency for text categorization. Pattern Recognition Letters, 45, 1-10.

Wikipedia (2016). List of Donald Trump presidential campaign endorsements, 2016. Retrieved from https://en.wikipedia.org/wiki/List_of_Donald_Trump_presidential_campaign_endorsements,_2016.

Yan, Y. (2018). Deep Learning Based Imbalanced Data Classification and Information Retrieval for Multimedia Big Data.

Yan, Y., Chen, M., Sadiq, S., & Shyu, M.-L. (2017). Efficient imbalanced multimedia concept retrieval by deep learning on spark clusters. International Journal of Multimedia Data Engineering and Management, 8(1), 1–20.

Yan, Y., Chen, M., Shyu, M.-L., & Chen, S.-C. (2015). Deep learning for imbalanced multimedia data classification. In Proceedings of the 2015 IEEE International Symposium on Multimedia (pp. 483–488).

Yan, Y., Hsieh, J.-W., Chiang, H.-F., Cheng, S.-C., & Chen, D.-Y. (2014). PLSA-based sparse representation for object classification. In Proceedings of the 2014 22nd International Conference on Pattern Recognition (pp. 1295-1300).

Yan, Y., Liu, Y., Shyu, M.-L., & Chen, M. (2014). Utilizing concept correlations for effective imbalanced data classification. In Proceedings of the 2014 IEEE 15th International Conference on Information Reuse and Integration (pp. 561-568).

Yan, Y., Pouyanfar, S., Tian, H., Guan, S., Ha, H.-Y., Chen, S.-C., Shyu, M.-L., & Hamid, S. (2016). Domain knowledge assisted data processing for Florida Public Hurricane Loss Model. In Proceedings of the 17th IEEE International Conference on Information Reuse and Integration (pp. 441-447).

Yan, Y. & Shyu, M.-L. (2016). Enhancing rare class mining in multimedia big data by concept correlation. In Proceedings of the IEEE International Symposium on Multimedia (pp. 281-286).

Yan, Y., Shyu, M.-L., & Zhu, Q. (2016). Supporting semantic concept retrieval with negative correlations in a multimedia big data mining system. International Journal of Semantic Computing, 10(2), 247-268.

Yan, Y., Shyu, M.-L., & Zhu, Q. (2016). Negative correlation discovery for big multimedia data semantic concept mining and retrieval. In Proceedings of the 10th IEEE International Conference on Semantic Computing (pp. 55-62).

Yan, Y., Zhu, Q., Shyu, M.-L., & Chen, S.-C. (2016). A classifier ensemble framework for multimedia big data classification. In Proceedings of the 17th IEEE International Conference on Information Reuse and Integration (pp. 615-622).

Yan, Y., Zhu, Q., Shyu, M. L., & Chen, S. C. (2016). Classifier Fusion by Judgers on Spark Clusters for Multimedia Big Data Classification. In Quality Software Through Reuse and Integration (pp. 91-108). Springer, Cham.

Zhu, Q., Lin, L., Shyu, M.-L., & Chen, S.-C. (2011). Effective supervised discretization for classification based on correlation maximization. In Proceedings of the 12th IEEE International Conference on Information Reuse and Integration (pp. 390–395).

Zhu, Q., Lin, L., Shyu, M.-L., & Chen, S.-C. (2010). Feature selection using correlation and reliability based scoring metric for video semantic detection. In Proceedings of the Fourth IEEE International Conference on Semantic Computing (pp. 462–469).