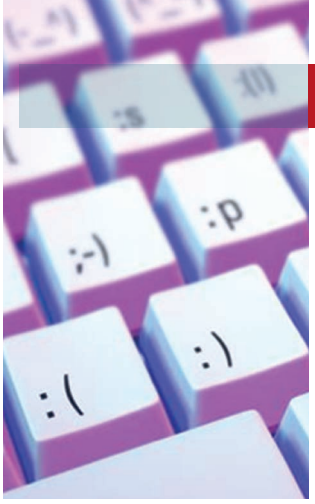

Editor: **Erik Cambria,** Nanyang Technological University, Singapore, cambria@ntu.edu.sg

Challenges of Sentiment Analysis for Dynamic Events

Monireh Ebrahimi, Amir Hossein Yazdavar, and Amit Sheth, *Kno.e.sis Center*

With the proliferation of social media over the last decade, the effort to determine people's attitudes with respect to a specific topic, document, interaction, or event has fueled research interest in natural language processing and introduced "sentiment and emotion analysis."[1] For instance, businesses routinely look to develop systems to automatically understand their customer conversations by identifying relevant content to better market their products and manage their reputations.[2] Previous efforts to assess people's sentiments on Twitter have suggested that Twitter could be a valuable resource for studying political sentiment and that it reflects the offline political landscape. According to a Pew Research Center report, in January 2016, 44 percent of US adults stated that they learned about the presidential election through social media. Furthermore, 24 percent reported using the two candidates' social media posts as a source of news and information, which is more than the 15 percent who used both candidates' websites or emails combined.[3] With 17.1 million tweets, the first presidential debate between Donald Trump and Hillary Clinton was the most tweeted debate ever.

Many opinion mining systems and tools provide users with people's attitudes toward products, people, or topics and their attributes/aspects. However, using sentiment analysis for predicting an election's result is still challenging. Though apparently simple, it is empirically challenging to train a successful model to conduct sentiment analysis on tweet streams for a dynamic event such as an election. Among the key challenges are changes in the topics of conversation and the people about whom social media posts express opinions. This article highlights some of the challenges related to sentiment analysis that we encountered during our monitoring of the presidential election using Kno.e.sis's Twitris system.[4] Twitris has successfully predicted several elections, including the 2012 US presidential election,[5] Brexit, and the 2016 US presidential election. The latter two involved collaboration between the Kno.e.sis Center and Cognovi Labs, a startup based on the Twitris technology that evaluates how the technology scales for real-time analysis. Figure 1 shows a dashboard used for real-time monitoring and analysis.

We first created a supervised multiclass classifier (positive versus negative versus neutral) for analyzing opinions about different election candidates as expressed in the tweets. To this end, we trained our model for each candidate separately. The motivation for this segregation comes from our observation that the same tweet on an issue can be positive for one candidate while negative for another. In fact, a tweet's sentiment is candidate-dependent. In the first round of training in July 2016, before the convention, we used 10,000 labeled tweets collected for five candidates (Bernie Sanders, Donald Trump, Hillary Clinton, John Kasich, and Ted Cruz) on nine issues (budget, finance, education, energy, environment, healthcare, immigration, gun control, and civil liberties). In addition to excluding retweets, we tested tweets for similarity using a ratio of Levenshtein distance to ensure that no two tweets were too similar. Afterward, through many experiments over different machine learning algorithms and parameter

1541-1672/17/$33.00 © 2017 IEEE
Published by the IEEE Computer Society

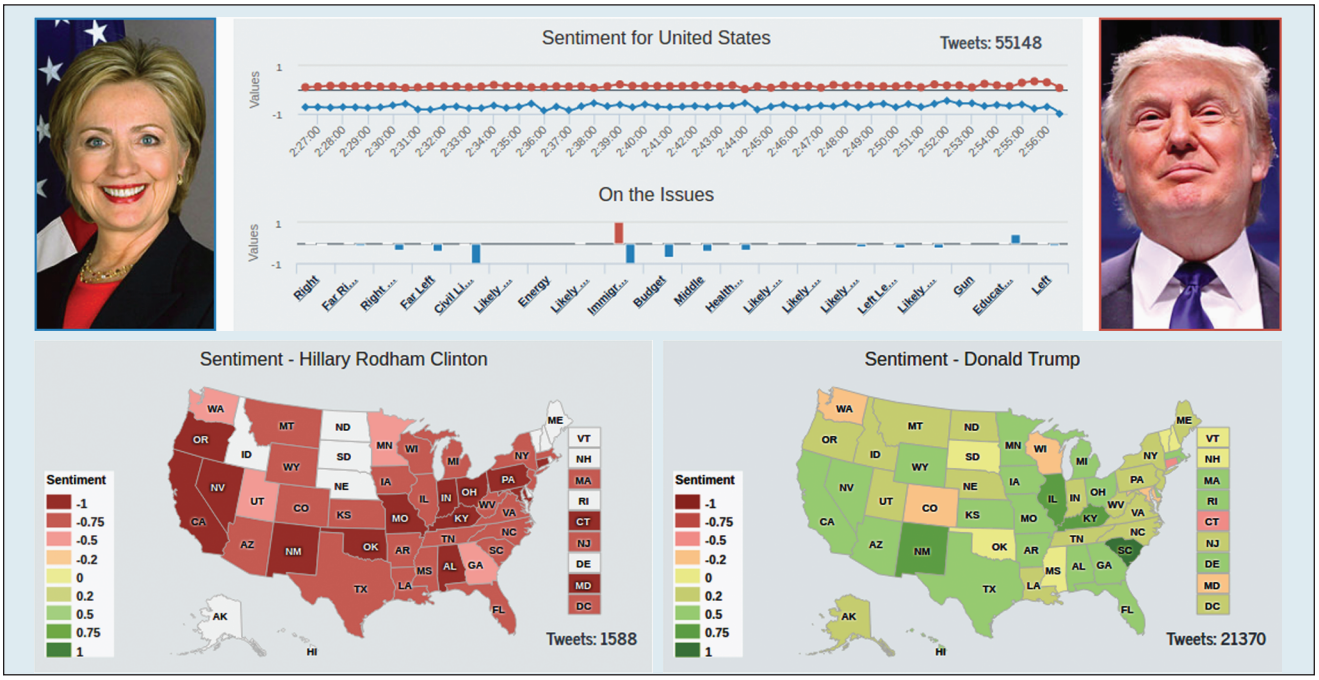

Figure 1. The Twitris system was used in real time to analyze the three presidential debates and for prediction on election day for the 2016 US election.

settings, we found our best model with respect to F-measure.

Our best traditional machine learning-based model for Clinton used a support vector machine (SVM) with term frequency-inverse document frequency (TF-IDF) vectorization of 1–3 grams, positive and negative hashtags for each candidate, and the number of positive and negative words (sentiment score), and achieved 0.66 precision, 0.63 recall, and a 0.64 F-measure. Through manual error analysis, we noticed the importance of considering a more comprehensive feature vector, including the number of positive and negative words, to avoid some outrageous errors. Therefore, we enriched our feature vector by leveraging a linguistic inquiry and word count (LIWC) program to glean linguistic style signals, including auxiliary verbs, conjunctions, adverbs, functional words, and prepositions, as well as the number of positive and negative words. Surprisingly, these features only improved our F-measure by around 1 percent. Apart from that, we also used a distributed vector representation of training instances obtained from a pretrained word2vec model on Twitter and Google News instead of using a discrete/traditional representation. Unfortunately, the performance decreased. Finally, we achieved the best accuracy using a deep learning-based model (convolutional neural network).

Fast-Paced Change in Dataset

The most challenging aspect of this work is creating a robust classification system to cope with the dynamic nature of election-related tweets. The election is active (or dynamic) since everyday people talk about new aspects of an election and the candidates in the context of unfolding events. Therefore, important features used to classify sentiment might soon become irrelevant, and emerging features would be neglected if we did not update the training set regularly. Furthermore, in the political domain, unlike many other domains, people mostly express their sentiment toward the candidates implicitly and without using sentiment words extensively.[6,7] This phenomenon makes the situation more challenging.

Another factor that might exacerbate the problem is differentiating the transient important features from lasting or recurring ones. Those features can disappear and then reappear in the future.[8] In the context of an election, for example, this scenario could occur because of the temporal changes in what each candidate's supporters talk about. Given the nonstationary characteristics of an election, we might encounter a concept drift/dataset shift problem—that is, learning when the test and training data have a different distribution. In fact, most machine learning approaches assume an identical distribution for the training and test set, although in many real-world problems, the test/target environment changes over time. This phenomenon is an important factor for selecting our classification model. SVM is one of the

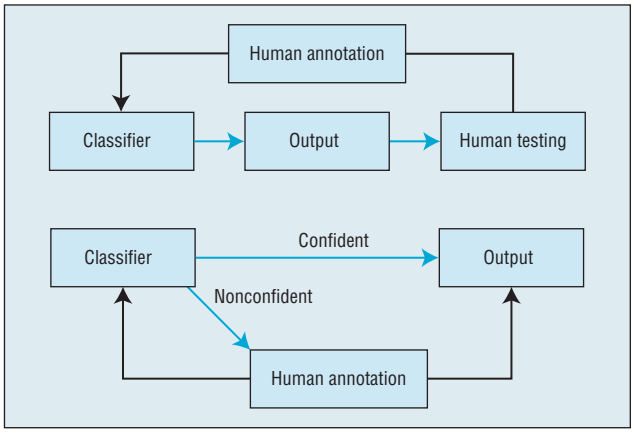

Figure 2. Active learning models.

most robust classification models for dataset shift.

All the aforementioned challenges make active learning necessary. As Figure 2 shows, there are two possible models for active learning that are useful to our problem. Both models are expensive since they involve humans in the loop for the labor-intensive and time-consuming task of annotation. Annotation is even more challenging here due to both the short length of tweets and the inherent vagueness in political tweets that requires awareness of the political context on the part of the annotators. A question might arise about why we have not used an unsupervised approach, such as a lexicon-based approach, when the annotation part is so challenging and our annotated dataset rapidly becomes obsolete and outdated. The answer is that in political tweets, people often do not use many sentiment words; hence, the performance of a lexicon-based method would be low. As a test, when we employed the multiperspective question answering (MPQA) subjectivity lexicon to capture the subjectivity of each tweet, the accuracy maxed out at 0.49.

Despite the costs, updating the training set regularly was the most effective measure for keeping the classifier reasonably good during the 2016 election. In fact, no matter how well our system might have been working, what worked well until yesterday could become useless today after a new political event, set of propaganda, or scandal. For example, our first model trained during the primaries performed quite poorly during and after the conventions. Therefore, feeding the dataset with a new training set was the key task for keeping the system reliable. To do so, we updated the training data in an opportunistic manner, usually every few days. It might also be worth trying to include more important/influential tweets in the training data. To achieve this goal, we collected the data mostly at specific times, such as during the presidential debates.

Candidate Dependence

Most sentiment analysis tools work in a target-independent manner. However, a target-independent sentiment analyzer is prone to yield poor results on our dataset because after conventions, a huge number of our tweets contain the names of both candidates. "I am getting so nervous because I want Trump to win so bad. Hillary scares me to death and with her America will be over," and "I don't really want Hillary to win but I want Trump to lose can we just do the election over," are examples of such tweets. Based on our observation, about 48 percent of our instances contained variants of both Clinton's and Trump's names. In such cases, the sentiment of those tweets might get misclassified for a given candidate because of the interference of features related to another candidate.

Current approaches for supervised target-dependent sentiment analysis can be grouped as syntax-based or context-based. The first group merely relies on part-of-speech tagging or syntax parsing for feature extraction,[9] while the second group defines the left and right context for each target.[10] The latter outperforms the former in the classification of informal texts such as tweets.[10] To further enhance performance, sentiment lexicon-expansion-related works[11] can be used to extract the sentiment-bearing, candidate-specific expressions, and those expressions can be added to a classifier's feature vector. In our case, since we have trained one classifier per candidate, we can include the instances containing the name of more than one candidate in the training sets of both classifiers. The key is to include features related to the target candidate in the corresponding classifier and exclude irrelevant ones in both the training and testing phases. To do that, we can use either dependency or proximity (similar to the two aforementioned works) to include the on-target features and ignore the off-target ones. Similarly, in the testing phase, depending on the classifier, we should include and exclude some of the features from our feature vector.

Identifying Users' Political Preferences

The ultimate goal of sentiment analysis over political tweet streams is predicting election results. Hence, obtaining some information about users' political preferences can provide more fine-grained sources of information to a political pundit or analyzer for insight. Inspired by work by Lu Chen and his colleagues,[11] we developed a simple but effective algorithm to categorize users into five groups of far left-leaning,

left-leaning, far right-leaning, right-leaning, and independent users.

The idea behind our approach is the tendency of users to follow others whose political orientation is similar to theirs. The more right-/left-leaning a user's followees are, the more likely it is that the user is also right-/left-leaning. Our approach involved collecting a set of Twitter users with known political orientations, including all senators, congresspersons, and political pundits. Then, we estimated the probability that a user is right-/left-leaning by calculating the ratio of right-/left-leaning followees of a user to the user's total number of followers. Finally, we determine a user's political preference by comparing this ratio with a threshold. Gaining this information about users helps to improve the social media-based prediction of the election.

Content-Related Challenges: Hashtags

Recently, there has been a surge of interest in distant supervision, which is training a classifier on a weakly labeled training set.[12] In this method, the training data gets automatically labeled based on a set of heuristics. In the context of sentiment analysis, using emoticons such as :) and :(as positive and negative labels, respectively, is one way of using distant supervision. Hashtags are also widely used for different machine learning tasks such as emotion identification.[13]

People use a plethora of hashtags in their tweets about an election. Because of the dynamic nature of the election domain, the quality, quantity, and freshness of labeled data play a vital role in creating a robust classifier. It is therefore desirable to use popular hashtags that each candidate's supporters use as a weak label in our dataset. However, our analysis for the 2016 election showed that hashtags were widely used for sarcasm, so using popular hashtags for automatic labeling leads to incorrectly labeling instances. For example, throughout the election, only 43 percent of tweets containing #Imwithher were positive for Clinton, while 27 percent used the hashtag sarcastically. Consequently, our experiments show that using those hashtags as a feature for our classifier will decrease accuracy rather than increase it.

The idea behind our approach is the tendency of users to follow others whose political orientation is similar to theirs. The more right-/left-leaning a user's followees are, the more likely it is that the user is also right-/left-leaning.

Content-Related Challenges: Links

All existing techniques for tweet classifiers rely solely on tweet content and ignore the content of the documents they point to through a URL. However, around 36 percent of the 2016 election tweets contained a URL to an external link. In the 2012 election,[11] we noticed that 60 percent of tweets from very highly engaged users contain URLs. Those links are crucial since without them, the tweet is often incomplete and inferring the sentiment is impossible or difficult even for a human annotator.

Therefore, we hypothesize that incorporating the content, keywords, or title of the documents that a URL points to as a feature will increase our performance. To the best of our knowledge, there is no work on tweet classification that expands tweets based on their URLs. However, link expansion has successfully been applied to other problems such as topical anomaly detection and distant supervision.[14]

Content-Related Challenges: Sarcasm

To date, many sophisticated tools and approaches have been proposed to deal with sarcasm. More recently, Soujanya Poria and his colleagues employed a deep neural network (pretrained convolutional neural network) for identifying sentiment, emotion, and personality features for sarcasm detection.[15] These works mostly focus only on detecting the sarcasm in the text and not on how to cope with it in the sentiment analysis task. This raises the interesting question about how sarcasm might or might not affect the sentiment of the tweets and how to deal with sarcastic tweets in both the training and prediction phases.

Ellen Rillof and her colleagues proposed an algorithm to recognize the common form of sarcasm that flips the polarity in the sentence.[16] These kinds of polarity-reversing sarcastic tweets often express the positive (negative) sentiment in the context of a negative (positive) activity or situation. However, Diana Maynard and Mark Greenwood show that determining the scope of sarcasm in tweets is still challenging.[17] In fact,

the polarity of sarcasm might apply to part of a tweet or its hashtags but not necessarily the whole. As a result, dealing with sarcasm in the task of sentiment analysis is an open research issue worth more work.

Based on our observation, 7 percent of Trump's tweets and 6 percent of Clinton's tweets are sarcastic. Among these sarcastic tweets, our system incorrectly classified 39 and 32 percent of them. In terms of the training set, we hypothesize that excluding the sarcastic instances from the training set will remove the noise and improve the quality of our training set.

Interpretation-Related Challenges: Sentiment Versus Emotion Analysis

The study of sentiment has evolved to the study of emotions, which has finer granularity. Positive, negative, and neutral sentiments can be expressed with different emotions such as joy and love for positive polarity; anxiety and sadness for negative; and apathy for neutral sentiment.

Our emotion analysis on who tweeted #IVOTED in the 2016 US presidential election showed that Trump had many more tweets and individuals expressing joyful emotion than Clinton. Although the sentiment analysis favored Clinton in the early hours, emotion analysis was showing better support for Trump. We considered emotion as a better criterion for predicting people's actions, such as voting, and usually there are significant emotional differences in the tweets that belong to the same polarity. This was key to our successful prediction of the 2016 election.

Interpretation-Related Challenges: Vote versus Engagement Counting

Most or all of the aforementioned challenges affect the quality of our sentiment analysis approach. It is also important to correlate a user's online behavior and opinion with that individual's actual vote. Chen and his colleagues show the more important role of highly engaged users in predicting results of the 2012 election.[11] There are two plausible explanations for this. First, the more a user tweets, the more reliably we can predict the user's opinion. Second, highly active people are usually more influential and more likely to actually vote in the real world. That is why an election monitoring system should report both user-level normalized sentiment and tweet-level sentiment. The end user analyzer must consider both factors in prediction.

> What happens when a large number of participants in a conversation are biased robots that artificially inflate social media traffic by manipulating public opinion and spreading political misinformation?

Importance of Location

An application that predicts the election result must consider each state's influence on the election using the number of electoral votes for that state. Many tools and approaches have been developed for both fine-grained[18] and coarse-grained[4] location identification in tweets for different purposes, such as disaster management and election monitoring. In the latter case, the geographic location of a tweet or the user location in the profile can be used to estimate the user's approximate location. During the 2016 election, the spatial aspect of our Twitris system played a crucial role in assisting users to predict the election.

Trustworthiness-Related Challenges

What happens when a large number of participants in a conversation are biased robots that artificially inflate social media traffic by manipulating public opinion and spreading political misinformation? A social bot is a computer algorithm that automatically generates content over social media and tries to emulate and possibly change public attitude. For the last few years, social bots have inhabited social media platforms. Similar to media reports,[19] we also witness bot wars between the two sides.

Research targeting pinpointing sources include use of supervised statistical models utilizing network features including retweets, mentions, and hashtag co-occurrence,[20] user features (such as language, geographical locations, account creation time, and number of followers and followees), and timing features (such as content generation and consumption, measuring tweet rate, and intertweet time distribution). Our effort to identify the source that generates a tweet (checking whether or not it originates from an API) using a hybrid and empirical approach gave fairly good results as elsewhere.

To sum up, in this study we highlighted the challenges/difficulties of building a robust sentiment analysis platform to capture subjective

signals from transient topics by focusing on the 2016 US presidential election. ▪

Acknowledgments

This work is supported in part by US National Science Foundation (NSF) award IIP 1542911, "PFI:AIR-TT: Market-Driven Innovations and Scaling Up of Twitris: A System for Collective Social Intelligence." Cognovi Labs has licensed the Twitris technology, and Amit Sheth is a cofounder of Cognovi Labs with an equity stake. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of NSF.

References

1. E. Cambria et al., "New Avenues in Opinion Mining and Sentiment Analysis," *IEEE Intelligent Systems*, vol. 28, no. 2, 2013, pp.15–21.
2. E. Cambria, "Affective Computing and Sentiment Analysis," *IEEE Intelligent Systems*, vol. 31, no. 2, 2016, pp. 102–107.
3. "Candidates Differ in Their Use of Social Media to Connect with the Public," Pew Research Center, 18 July 2016; http://j.mp/PewSocM.
4. A. Sheth et al., "Twitris: A System for Collective Social Intelligence," *Encyclopedia of Social Network Analysis and Mining*, 2nd ed., R. Alhajj and J. Rokne, eds., Springer, 2018, p. 1–23.
5. L. Chen, W. Wang, and A. Sheth, "Are Twitter Users Equal in Predicting Elections? A Study of User Groups in Predicting 2012 US Republican Presidential Primaries," *Proc. Int'l Conf. Social Informatics*, Springer, 2012, pp. 379–392.
6. M. Ebrahimi, et al., "Recognition of Side Effects as Implicit-Opinion Words in Drug Reviews," *Online Information Rev.,* vol. 40, no. 7, 2016, pp. 1018–1032.
7. A.H. Yazdavar, M. Ebrahimi, and N. Salim, "Fuzzy Based Implicit Sentiment Analysis on Quantitative Sentences," arXiv: 1701.00798, 2017.
8. F.A. Pinage, E.M. dos Santos, and J.M. Portela da Gama, "Classification Systems in Dynamic Environments: An Overview," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery,* vol. 6, no. 5, 2016, pp. 156–166.
9. L. Jiang et al., "Target-Dependent Twitter Sentiment Classification," *Proc. 49th Ann. Meeting Assoc. for Computational Linguistics: Human Language Technologies*, vol. 1, 2011, pp. 151–160.
10. D. Vo and Y. Zhang, "Target-Dependent Twitter Sentiment Classification with Rich Automatic Features," *Proc. 24th Int'l Conf. Artificial Intelligence* (IJCAI), 2015, pp. 1347–1353.
11. L. Chen et al., "Extracting Diverse Sentiment Expressions with Target-Dependent Polarity from Twitter," *Proc. 6th Int'l AAAI Conf. Weblogs and Social Media*, 2012, pp. 50–57.
12. A. Go, R. Bhayani, and L. Huang, "Twitter Sentiment Classification Using Distant Supervision," CS224N Project Report, Stanford Univ., 2009.
13. W. Wang et al., "Harnessing Twitter 'Big Data' for Automatic Emotion Identification," *Proc. Int'l Conf. Privacy, Security, Risk and Trust* (PASSAT) and *Proc. Int'l Conf. Social Computing* (SocialCom), 2012, pp. 587–592.
14. W. Magdy et al., "Distant Supervision for Tweet Classification Using YouTube Labels," *Proc. 9th Int'l AAAI Conf. Web and Social Media* (ICWSM), 2015, pp. 638–641.
15. S. Poria et al., "A Deeper Look into Sarcastic Tweets Using Deep Convolutional Neural Networks," *Proc. 26th Int'l Conf. Computational Linguistics: Technical Papers* (COLING), 2016. pp. 1601–1612.
16. E. Riloff et al., "Sarcasm as Contrast Between a Positive Sentiment and Negative Situation," *Proc. Conf. Empirical Methods in Natural Language Processing* (EMNLP), 2013, pages 704–714.
17. D. Maynard and M.A. Greenwood, "Who Cares about Sarcastic Tweets? Investigating the Impact of Sarcasm on Sentiment Analysis," *Proc. 9th Int'l Conf. Language Resources and Evaluation* (LREC), 2014, pp. 4238–4243.
18. Z. Ji et al., "Joint Recognition and Linking of Fine-Grained Locations from Tweets," *Proc. 25th Int'l Conf. World Wide Web*, 2016, pp. 1271–1281.
19. N. Byrnes, "How the Bot-y Politic Influenced This Election," *MIT Technology Rev.*, 8 Nov. 2016; http://bit.ly/BotyE16.
20. K. Lee, B.D. Eoff, and J. Caverlee, "Seven Months with the Devils: A Long-Term Study of Content Polluters on Twitter," *Proc. Int'l AAAI Conf. Web and Social Media* (ICWSM), 2011.

Monireh Ebrahimi is a research assistant at the Kno.e.sis Center and a PhD candidate in the Computer Science Department at Wright State University. Her research interests include deep learning, natural language processing, and social media analysis. Ebrahimi has an MS in computer science from Wright State University. Contact her at monireh@knoesis.org.

Amir Hossein Yazdavar is a research assistant at the Kno.e.sis Center and a PhD candidate in the Computer Science Department at Wright State University. His research interests include machine learning, deep learning, natural language processing, and social media analysis. Yazdavar has an MS in computer science. Contact him at amir@knoesis.org.

Amit Sheth is the LexisNexis Ohio Eminent Scholar, executive director of the Ohio Center of Excellence in Knowledge-Enabled Computing (Kno.e.sis) at Wright State University, and an IEEE Fellow. Contact him at amit@knoesis.org; http://knoesis.org.amit.