2017 IEEE International Conference on Information Reuse and Integration

AAFA: Associative Affinity Factor Analysis for Bot Detection and Stance Classification in Twitter

Saad Sadiq[1], Yilin Yan[1], Asia Taylor[2], Mei-Ling Shyu[1], Shu-Ching Chen[3], Daniel Feaster[4]
[1]*Department of Electrical and Computer Engineering*
University of Miami, Coral Gables, FL, USA
saadsadiq@miami.edu, y.yan4@umiami.edu, shyu@miami.edu
[2]*Department of Computer Science*
University of Miami, Coral Gables, FL, USA
axt634@miami.edu
[3]*School of Computing and Information Sciences*
Florida International University, Miami, FL, USA
chens@cs.fiu.edu
[4]*Department of Public Health Sciences, Miller School of Medicine*
University of Miami, Miami, FL, USA
dfeaster@biostat.med.miami.edu

***Abstract*—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. It is noted that highly automated accounts with 450 tweets per day produced almost 18% of entire Twitter circulation in the 2016 U.S. Presidential election. It is also observed that those disruptive systems called bots are inclined more towards circulating negative news than positive information. This paper introduces a novel framework named Associative Affinity Factor Analysis (AAFA) designed for stance detection and bot identification. Using AAFA, the proposed framework identifies real people from bots and detects the stance in bipolar affinities. The 2016 U.S. Presidential election campaign was used as a test use case because of its significant and unique counter-factual properties. The results show that our proposed AAFA framework achieves high accuracy when compared to several existing state-of-the-art methods.**

***Keywords*-Bot detection; stance classification; association affinity; factor analysis**

I. Introduction

In recent years, there has been a major growth in the use of microblogging platforms. Microblogs allow the users to exchange small contents such as short videos, sentences, and links. Some previous research efforts were paid on these kinds of multimedia data [1]–[10]. Twitter is one of the most widely used microblog platforms. Users range from regular users to politicians, celebrities, and company representatives. Therefore, it is possible to collect posts of users from different social and interested groups. On the flip side, Twitter is littered with automated agents called chat bots. Chat bots are rudimentary software systems with minimal automation and basic conversation abilities. They direct their scripts on social media outlets to tirade, obscure the facts, or merely make the conversations cloudy. It is estimated that as many as 48 million Twitter accounts are bots and from the 19.4 million tweets during elections, 1300 tweets per day were produced by bots [11]. As an example, these anonymous chat machines were an integral part of a prearranged effort to disturb the 2016 U.S. Presidential election. To visualize the overall picture, Figure 1 is used to show the daily tweet count between Hillary Clinton and Donald Trump for the election time period in 2016. It is worthy to note that 33% of pro-Trump traffic was driven by bots and highly automated accounts, compared to 22% for Clinton. The popularity measure between the two candidates is shown in Figure 2 by mapping the retweet and favorite counts of the two candidates. The objective of this paper is to build an automated system for bot detection in Twitter accounts by the proposed Associative Affinity Factor Analysis (AAFA) framework.

Another important research direction is stance analysis which implies the political tendency of the public. In this paper, "stance classification" is defined as automatically determining whether a Twitter user tends to endorse the candidate of Democratic or Republican Party. By tweets from the Twitter accounts, researchers can deduce whether a user is either for or against the target. Therefore, another objective of this paper is to automatically infer the stances of Twitter users to see whether a user is likely a Hillary Clinton or Donald Trump supporter. While most election predictions reply on polls, automated stance classification can be applied to a much larger number of samples and bring complementary information to predict the election results.

The remaining of the paper is organized as follows. In Section 2, some previous work on bot detection and stance analysis are briefly presented. Then, some domain

978-1-5386-1562-1/17 $31.00 © 2017 IEEE
DOI 10.1109/IRI.2017.25

knowledge about the Twitter data and how to clean and extract the election dataset are introduced in Section 3. Section 4 shows the proposed framework in details. The experimental results are provided in Section 5, which proves the efficiency of AAFA. Finally, Section 6 concludes this paper with several future research directions.

II. Previous Work

Based on our best knowledge, though bot detection and stance analysis have not been used for election prediction, 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 [12]. The sentiments were identified as a powerful predictor in differentiating the behaviors of various accounts. Agarwal et al. [13] 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 [14] 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 [15] to focus and improve the corpus by an automatic collection and by using TreeTagger for POS-tagging. The wide scale effects of socioeconomic events on the overall general mood of tweets were explored by [14] 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. A significant impact was made by [16] by creating a 60-"honeypot" trap for 7 months to send gibberish tweets and consequently attracted 36000 fake Twitter accounts. They follow each other to avoid Twitter filters, resulting in thousands of followers among themselves.

A hypothesis was proposed in [17] that every non-hyperbolic tweet was from Donald Trump's staff while every hyperbolic tweet was from Donald Trump himself. The researchers collected Donald Trump's tweets from Donald Trump's account including the "source" information and found out that most tweets are from either iPhones or Android phones. Their analysis showed that the iPhone and Android tweets are 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 [18], 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 sentiment. Each posted tweet related to Hillary Clinton or Donald Trump was labeled 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 Hillary Clinton than Donald Trump. In [19], two groups of hashtags were defined arbitrarily, where each group was assumed to support Hillary Clinton or Donald Trump, respectively. After that, the author used descriptive statistics methods and concluded that Donald Trump's campaign knew more about how to use Twitter chat bots than the Hillary Clinton's side.

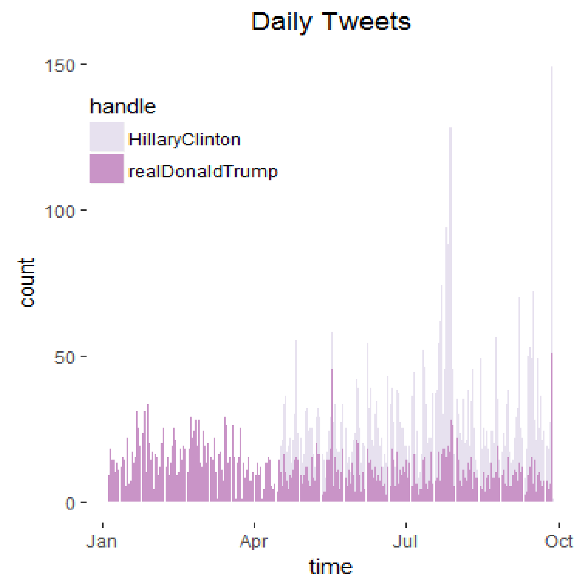

Figure 1: Daily tweets of the two candidates

III. Twitter Dataset

A. Data Collection and Pre-processing

In order to do stance analysis for the 2016 U.S. election test use case, a dataset that includes the supporters of both sides is necessary. However, due to 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 [20]. These lists include "big names" who have publicly claimed their endorsements for the office of the president to Hillary Clinton or Donald Trump as their presidential nominees. Since these supporters are notable individuals, the information was reliable and did not change much in the campaign. After data cleaning, 310 supporters of Hillary Clinton and 412 supporters of Donald Trump were included to build the experimental dataset.

In addition, the Twitter API was used to collect 3240 tweets from each supporter with time, resource, retweet,

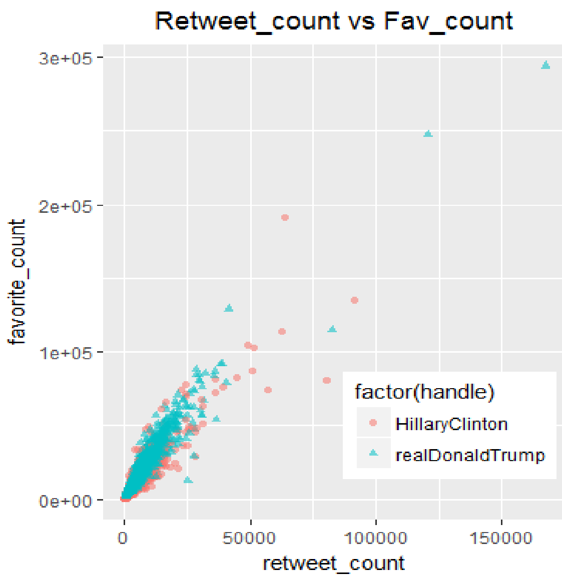

Figure 2: Popularity vs following measure of all supporters grouped by their candidates

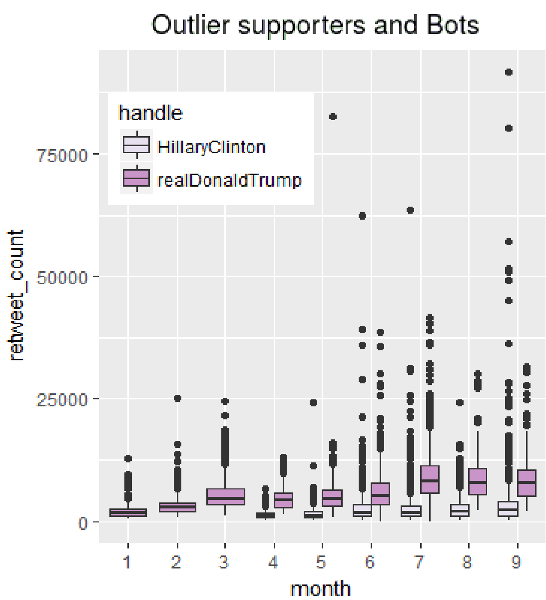

Figure 3: Popularity of supporters by the averaged monthly retweet count and the candidate they support

etc. After the data collection, we extracted the details of the supporters' accounts, cleaned the text data from all tweets, and mapped the truncated words to get the hashtag information.

B. Counterfactual Bipartisanship

Identifying bots in a binary stance topic such as Hillary Clinton vs Donald Trump is a huge challenge. It is observed that the election dataset is a unique domain where people

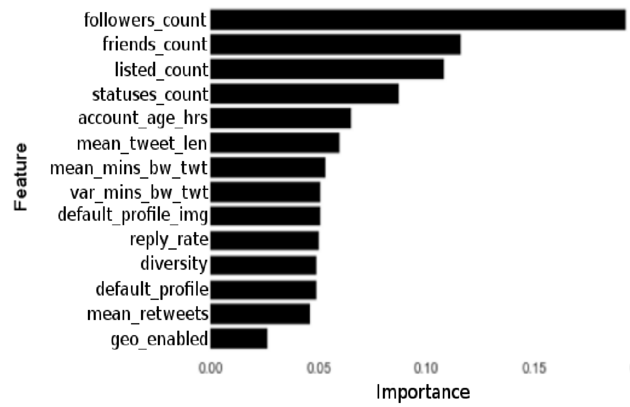

Figure 4: Ranked importance of Twitter metadata features

who support one candidate or party have counterfactual negative sentiments for the other party, i.e., the supporters of Hillary Clinton would predominantly be the haters of Donald Trump. Figure 3 shows the retweet counts of the month by month data between the two poles of our stance detection. The information contains followers' count, favorites, retweet count, and other account metadata of their accounts. Figure 4 ranks the most important features from the Twitter metadata.

C. Generating Personal Bot Army

It is currently an active research problem to correctly identify a bot from a real person. Some of the infamous twitter bots participating in the 2016 U.S. elections went under the names like @keksecorg__ , @NeilTurner_ , @WhiteGenocideTM, etc. However, the list of verified bots is very small and it was very difficult to get a dataset that has the ground truth information. To solve this problem, we purchased 3000 fake bots from the social freelance marketplace called Fiverr [21]. Moreover, the metadata of their accounts was extracted and around 50 tweets were obtained for each account. The bots were advertised to perform the following tasks:

1) retweet specific hashtags
2) plagiarize posts from specific accounts
3) tweets about specified topics
4) retweet specified accounts
5) post curated links every 30 minutes
6) tag targeted user account in retweets

To compare these bots with real accounts, the friends' counts were compared among Bots, Hillary Clinton's supporters, and Donald Trump's supporters in Figure 5. A stark separation was observed between human supporters and bot followers from the dataset. This shows that we have a strong case of detecting bots from humans in our dataset.

IV. The Proposed Framework

In this paper, an Association Affinity-based Factor Analysis (AAFA) framework [22]–[26] is proposed for the iden-

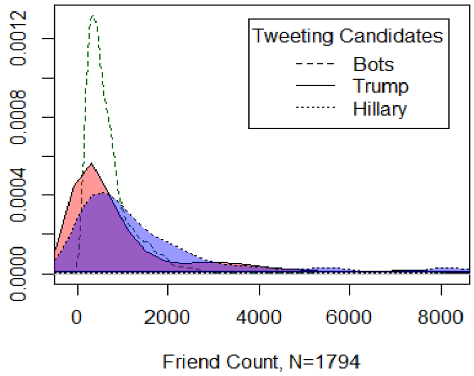

Figure 5: Density plot of the friend counts among Bots, Donald Trump's and Hillary Clinton's supporters

tification of bots and detection of stance in Tweets. Figure 6 illustrates the workflow of the proposed AAFA framework. To the best of our knowledge, this is the first implementation of using factor analysis in Twitter bot detection and stance analysis domain. There are studies that used correlations among attributes to refine the classification results [27]–[37]. Adopting such an idea, the proposed model first creates a latent class profile on Twitter user account metadata and the average sentiment score of their tweets, and then performs multiple factor analysis on the mixed feature dataset that is composed of the metadata and derived attributes. The AAFA framework results in a highly accurate framework to identify whether the tweets being posted in the election campaign were from real human or bot accounts. Furthermore, the filtered bot data is then passed to the stance detection section where the association affinity scores are evaluated for predictive hashtags of Twitter user accounts. The significant contributions of our proposed framework include:

1) highly accurate in bot classification
2) achieving a 19.5% increase in the F1 score when compared with the benchmark tool [38]
3) accurately identifying the stance of influencers
4) context invariant not affected by the bots of different domains
5) unsupervised learning

A. Multiple Factor Analysis for Mixed Data

The proposed AAFA framework uses Multiple Factor Analysis (MFA) [39], [40], generally a combination of Principal Component Analysis (PCA) [41] and Multiple Correspondence Analysis (MCA) [42], for mixed-variable Twitter election datasets. MFA is implemented in two stages. Initially, a PCA is executed on a subset of feature space j, as shown in Figure 7. This is further standardized by dividing the weights of the features by λ_1^j, i.e., the first eigenvalue of set j. These normalized and unit variance principal components form the basis function of MFA of a dataset with mixed variables. Second, the categorical variables are

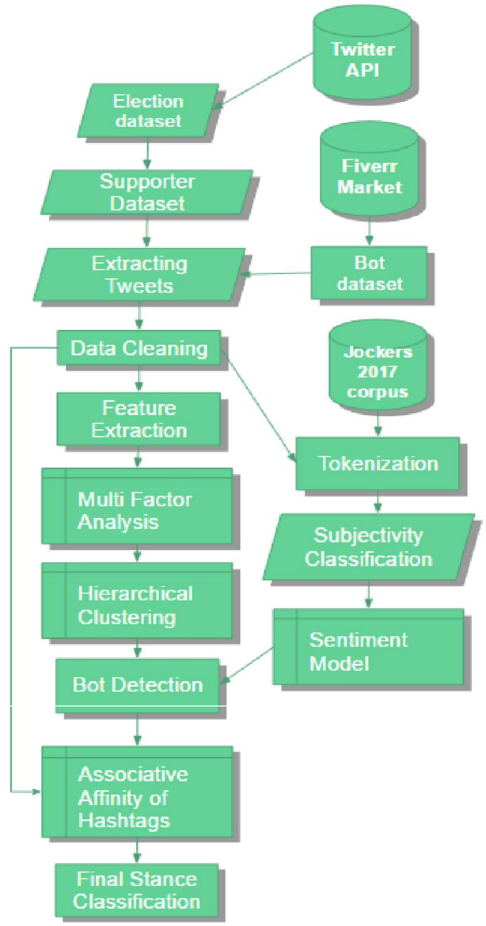

Figure 6: Proposed AAFA Framework

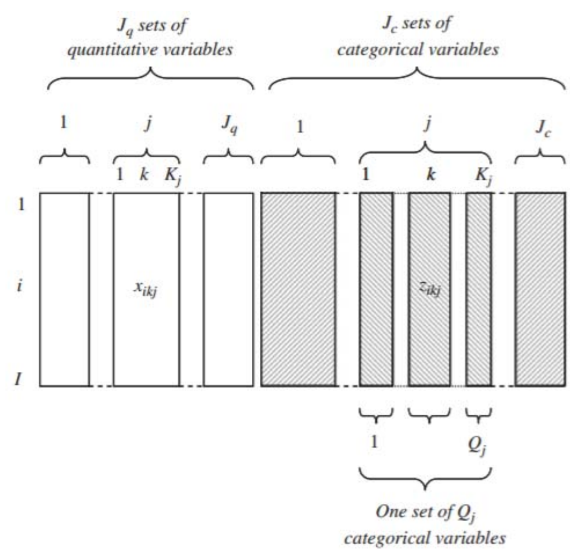

Figure 7: Combined dataset where quantitative and nominal/ordinal variables are juxtaposed

transformed into a disjunctive data table using crisp coding [43] in the form of indicator variables (either 0 or 1) by stratifying the variable categories. We then apply multiple correspondence analysis (MCA) to scale the variables and yield the eigenvalues.

Let us formulate this process by defining the dataset as I samples having J sets of data with mixed variables such that $J = J_q + J_c$, where J_q and J_c represent the sets of quantitative and categorical variables, respectively. Moreover, i refers to an arbitrary sample, k indicates a particular feature column, and j represents a group of features, such that at the crossing of row i and column k, belonging to set j, we have:

1) if j is a quantitative set, the value x_{ikj} of the variable k for the unit i;
2) if j is a categorical set, $z_{ikj} = 1$ if i belongs to the category k and 0 if it doesn't.

These standardized datasets are then combined together to build a distinct matrix. This balances the influence of both continuous and categorical variables in the analysis such that both variables can equally determine the dimensions of variability. To merge these two types of variables, (i.e., sets J_q and J_c), the equivalence between MCA and PCA is calculated as follows.

1) Applying Global PCA to the table with the general term $(z_{ikj} - w_{kj})/w_{kj}$;
2) Assigning the weight w_{kj}/Q_j to column k of set j;
3) Assigning the weight p_i to row i.

Here, $z_{ikj} = 1$ if i belongs to the category k, and 0 otherwise. $w_{kj} = \sum_{i \in I} p_i \cdot z_{ikj}$ with p_i being the uniformly distributed weight allocated to each sample i, with a default value of 1. Furthermore, a distance is generated among units in the form of a weighted sum created by each individual variable. Based on these weighted coordinates, the final square distance evaluated from the coordinates (and metric in the unit space) is defined as:

$$\begin{aligned} d^2(i,l) = \sum_{j \in J_q} \frac{1}{\lambda_1^j} \sum_{k \in K_j} \left[\frac{x_{ikj} - x_{lkj}}{s_{kj}} \right] + \\ \sum_{j \in J_c} \frac{1}{Q_j w_{kj}} [z_{ikj} - z_{lkj}]^2, \end{aligned} \tag{1}$$

where K_j refers to the total number of features (i.e., columns) of each data type in set j belonging to either J_q or J_c. Equation (1) signifies the part played by each of the variables to the global principal components as follows.

1) Quantitative variables j ($j \in J_q$) help evaluate the distance between units i and l when PCA is applied;
2) Categorical variables j ($j \in J_c$) help evaluate the distance between units i and l when MCA is applied.

Clustering can then be performed from the principal coordinates by computing the classical Euclidean distance, for which we use hierarchical clustering.

B. Hierarchical Clustering

Hierarchical trees considered in this paper use the Ward's criterion [44] performed on the principal components evaluated in the MFA step. This criterion is based on the Huygens theorem [45] that decomposes the total inertia to be:

$$\begin{aligned} \sum_{k=1}^{K} \sum_{c=1}^{C} \sum_{i=1}^{I_c} (x_{ick} - \bar{x}_k)^2 = \sum_{k=1}^{K} \sum_{c=1}^{C} I_c (\bar{x}_{ck} - \bar{x}_k)^2 \\ + \sum_{k=1}^{K} \sum_{c=1}^{C} \sum_{i=1}^{I_c} (x_{ick} - \bar{x}_{ck})^2 \end{aligned} \tag{2}$$

Here, the first and second terms on the right are Between and Within inertia, respectively; while the term on the left is the total inertia. Let x_{ick} be the value of variable k for sample i of cluster c, $\bar{x}_{ck}$ be the mean of variable k for cluster c, where K refers to the total number of columns in the dataset. Let $\bar{x}_k$ be the overall mean of variable k, and I_c be the number of samples in cluster c. The hierarchical clusters are depicted in the form of a dendogram ranked by the increase in the inertia. To explain the hierarchical divisions, we measure the correlation between each division (i.e., a categorical variable) and

1) each quantitative feature by the square correlation ratio η^2;
2) each categorical feature by the Cramer's coefficient V.

Cramer's coefficient normalizes the χ^2 statistic by using the maximum value of the χ^2 statistic to divide it as given in Equation (3).

$$V = \sqrt{\frac{\chi^2}{T \cdot min(I-1, K-1)}}, \tag{3}$$

where χ^2 is the chi-square statistic, T is the grand total of the table, and I and K refer to the total numbers of samples and features of the table.

C. Affinity-based Stance Detection

Consider each hashtag in a tweet as a concept and find the recurring itemset in Donald 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) [6], [46], [47], then they are bots. 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 Hillary Clinton tended to use different hashtags compared to those supported Donald Trump.

One attempt in the literature [19] built two groups of arbitrary hashtags, from their domain knowledge, to find the Hillary Clinton and Donald Trump supporters. However, this approach lacks reproducibility and domain invariance. To overcome this challenge and find distinct hashtags, we apply the log odds ratio approach [48]. For a hashtag n, we calculate C_n^H and C_n^T which represent the numbers of times n was used by the Hillary Clinton supporters and Donald Trump supporters. Similarly, U_n^H and U_n^T represent the numbers of distinct accounts of the Hillary Clinton and Donald Trump supporters that used hashtag n. Next, the scores S_n^C and S_n^U are calculated to measure the likelihood values of a hashtag being associated with either of the candidates (as shown in Equations (4) and (5)).

$$S_n^C = \log_2\left(\frac{\frac{C_n^H+1}{\sum_{i=1}^{N} C_i^H+1}}{\frac{C_n^T+1}{\sum_{i=1}^{N} C_i^T+1}}\right); \quad (4)$$

$$S_n^U = \log_2\left(\frac{\frac{U_n^H+1}{\sum_{i=1}^{N} U_i^H+1}}{\frac{U_n^T+1}{\sum_{i=1}^{N} U_i^T+1}}\right). \quad (5)$$

Here, N refers to the total number of supporters. The scores and the ranked hashtags are given in Table II. For comparison, the hashtag lists are shown by the domain knowledge [19] and the tweets from the candidates (i.e., Hillary Clinton and Donald Trump) [18] in Table I. It is clear that some unique tashtags can only be automatically found using the proposed framework.

V. Experiment and Results

Using the dataset extracted and the cleaned information, the experiments are conducted and three-fold cross validation is applied for the comparisons.

A. Results of Bot Detection

The clustering is based on the inertia gained when we go from one cluster to two clusters. We also get further significant inertia gain while going from two clusters to three clusters, which indicates that there are subgroups of human followers within the Hillary Clinton and Donald Trump camps. Hierarchical clustering performed on the principal components gives three kinds of insights, namely

1) the principal components
2) the projections of variables on these principal components
3) the variable associations and clusters

Figure 8 shows the graph of the quantitative variables after applying the PCA. The coordinate axes here represent the first two principal components and the arrows depict the cosine angles between the variables and the principal components. The percentage on each coordinate axis represents the proportion of variances retained by the each principal component. Variables close to the circle illustrate a high correlation with other variables and the direction of the variable vector indicates the correlation polarity between any two given variables.

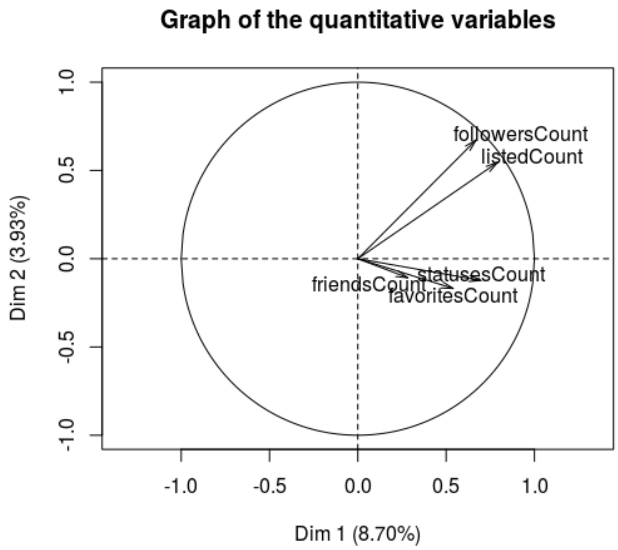

Figure 8: Correlation circle for the continuous variables after performing PCA

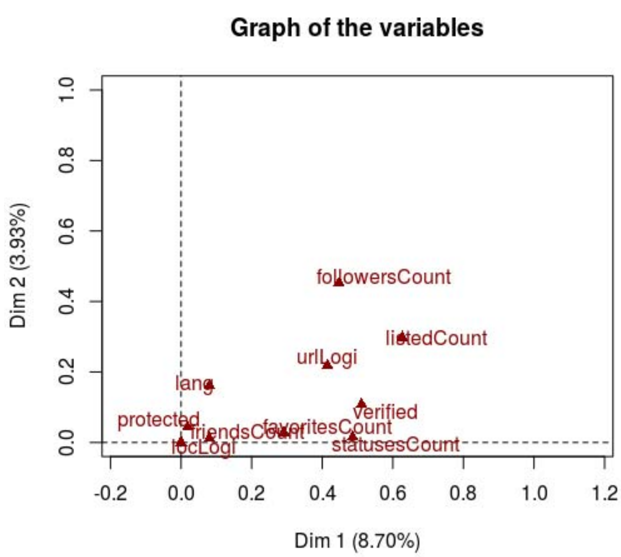

Figure 9: Square correlation ratio of the continuous and categorical variables

Figure 9 illustrates the square correlation ratio (variable associations) between both types of variables, continuous and categorical, along the coordinate axes of the first and second principal components. The squared correlation ratio quantifies the correlation between the continuous and categorical variables. It was used in the framework to calculate one way ANOVA test of data separation. The proposed AAFA framework was performed on the Twitter dataset to

Table I: 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	$votehillary2016$	$MakeAmericaGreatAgain$	$DemsInPhilly$	$Trump$
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	$TrumpUnfit$	$TrumpsArmy$	$LoveTrumpsHate$	$GOPDebate$

Table II: 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 III: Comparative evaluation of Multiple Factor Analysis with several leading models

	F1-score	Accuracy	Recall	Precision
UnSupRF	0.73	0.58	0.95	0.56
BotOrNot	0.71	0.66	0.69	0.73
K-Means	0.75	0.61	1.0	0.60
C-Means	0.75	0.61	1.00	0.60
ExpMax	0.93	0.92	0.88	0.99
Proposed	**0.96**	**0.96**	**0.97**	**0.95**

Table IV: Accuracy and F-score comparisons in stance analysis

Classifier	Accuracy	F1-score
SVM	0.8089	0.7128
RF	**0.8492**	**0.8156**
DAC	0.7493	0.6190
Linear	0.7812	0.6853
Logistic	0.7867	0.7061

identify bots vs humans, and compared with the following state-of-the-art unsupervised models in this domain:

1) Unsupervised Random Forest (UnSupRF) [49]
2) Truthy's BotOrNot [38]
3) K-Means Clustering [50]
4) Fuzzy C-Means Clustering [51]
5) Expectation Maximization (ExpMax) [52]

As can be seen from Table III, the proposed framework achieves a significant improvement over those models in the comparison. For example, our proposed framework achieves 0.96 in accuracy in comparison to 0.66 from the industry's goto platform Truthy's BotOrNot.

B. Results of Stance Classification

To apply the affinity-based stance detection method, the hashtags of people supporting Hillary Clinton and Donald 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. Each hashtag itemset has a dynamic threshold but the hashtags with the highest affinities were selected. Table II illustrates these case insensitive ranked hashtags for the two candidates. The final hashtag lists are selected based on both the "number of a hashtag being used" and the "number of distinct accounts that use a hashtag". The overlapped hashtags are cleaned and finally a list of 128 hashtags is created to generate the feature vectors for the Hillary Clinton and Donald Trump supporters. Based on the number of a hashtag used, a feature vector is generated and normalized for each account.

For comparison, our stance classification model is evaluated against several popular classifiers including Support Vector Machine (SVM) [53], Random Forest (RF) [54], discriminant analysis classifier (DAC), Linear Regression, as well as Logistic Regression (LR) [55]. As shown in Table IV, an average accuracy of 80 percent is obtained without any domain knowledge and polls. Random Forest performs the best for this task due to the nature of our feature vectors (i.e., different weights for the hashtags).

An important insight is to observe whether the accuracy of a clustering method would be affected if we use real accounts with unknown predilections. This would help us also identify and evaluate the undecided voters. Currently, it is out of the scope of this paper to assert the ground truth for accounts having relative unknowns, but an extension of this framework will be to collect and extend the dataset with hand-labeled real accounts, and re-evaluate the stance detection and bot identification.

VI. Conclusions

The power of propaganda is reinforced when a limited number of individuals believe that it is prevalent. The part played by false news and fabricated information in the 2016 U.S. elections proved to be a painful experience for the information technology industry. This paper proposes a novel framework to detect the stance between the followers of the two dominant presidential candidates, Hillary Clinton and Donald Trump, and to separate real vs bot accounts. For our best knowledge, we are the first group that uses machine learning algorithms for stance analysis in election predictions. We are able to accurately identify the truth behind the number of Twitter followers and social media popularity by dissecting the real followers from paid bots. Our results show that the proposed framework is more accurate than the industry's most popular tool, BotOrNot. In the future, other information in tweets including the resources, retweets, favorites, etc. would be also considered for better stance detection and bot classification.

References

[1] S.-C. Chen, M.-L. Shyu, C. Zhang, and R. L. Kashyap, "Identifying overlapped objects for video indexing and modeling in multimedia database systems," *International Journal on Artificial Intelligence Tools*, vol. 10, no. 4, pp. 715–734, 2001.

[2] S.-C. Chen, S. Sista, M.-L. Shyu, and R. Kashyap, "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*, 1999, pp. 175–182.

[3] X. Chen, C. Zhang, S.-C. Chen, and M. Chen, "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*, Dec 2005, pp. 37–44.

[4] Q. Zhu, L. Lin, M.-L. Shyu, and S.-C. Chen, "Effective supervised discretization for classification based on correlation maximization," in *Proceedings of the IEEE International Conference on Information Reuse and Integration*, 2011, pp. 390–395.

[5] D. Liu, Y. Yan, M.-L. Shyu, G. Zhao, and M. Chen, "Spatio-temporal analysis for human action detection and recognition in uncontrolled environments," *International Journal of Multimedia Data Engineering and Management*, vol. 6, no. 1, pp. 1–18, Jan. 2015.

[6] M.-L. Shyu, S.-C. Chen, and R. Kashyap, "Generalized affinity-based association rule mining for multimedia database queries," *Knowledge and Information Systems (KAIS): An International Journal*, vol. 3, no. 3, pp. 319–337, August 2001.

[7] M.-L. Shyu, C. Haruechaiyasak, and S.-C. Chen, "Category cluster discovery from distributed www directories," *Information Sciences*, vol. 155, no. 3, pp. 181–197, 2003.

[8] M. L. Shyu, Z. Xie, M. Chen, and S. C. Chen, "Video semantic event/concept detection using a subspace-based multimedia data mining framework," *IEEE Transactions on Multimedia*, vol. 10, no. 2, pp. 252–259, Feb 2008.

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

[10] M.-L. Shyu, K. Sarinnapakorn, I. Kuruppu-Appuhamilage, S.-C. Chen, L. Chang, and T. Goldring, "Handling nominal features in anomaly intrusion detection problems," in *15th International Workshop on Research Issues in Data Engineering: Stream Data Mining and Applications (RIDE-SDMA 2005)*, 2005, pp. 55–62.

[11] O. Varol, E. Ferrara, C. A. Davis, F. Menczer, and A. Flammini, "Online human-bot interactions: Detection, estimation, and characterization," *CoRR*, vol. abs/1703.03107, 2017. [Online]. Available: http://arxiv.org/abs/1703.03107

[12] E. Kouloumpis, T. Wilson, and J. D. Moore, "Twitter sentiment analysis: The good the bad and the omg!" *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media*, vol. 11, no. 538-541, p. 164, 2011.

[13] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. Passonneau, "Sentiment analysis of twitter data," in ., Association for Computational Linguistics. Proceedings of the Workshop on Languages in Social Media, 2011, pp. 30–38.

[14] J. Bollen, H. Mao, and A. Pepe, "Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena." in ., vol. 11. Barcelona, Spain: International AAAI Conference on Weblogs and Social Media, 2011, pp. 450–453.

[15] A. Pak and P. Paroubek, "Twitter as a corpus for sentiment analysis and opinion mining." in ., vol. 10. European Language Resources Association (ELRA), 2010.

[16] K. Lee, B. D. Eoff, and J. Caverlee, "Seven months with the devils: A long-term study of content polluters on twitter." in *International AAAI Conference on Weblogs and Social Media*, Barcelona, Spain, 2011.

[17] D. Robinson, "Text analysis of trump's tweets confirms he writes only the (angrier) android half," http://varianceexplained.org/r/trump-tweets, accessed May 7th, 2017.

[18] B. Stecanella, "Donald trump vs hillary clinton: sentiment analysis on twitter mentions," https://blog.monkeylearn.com/donald-trump-vs-hillary-clinton-sentiment-analysis-twitter-mentions, accessed May 7th, 2017.

[19] Y. Jia, "Trump vs hillary on twitter," http://yiyujia.blogspot.com/2016/09/trump-vs-hillary-on-twitter, accessed May 7th, 2017.

[20] Wikipedia, "List of donald trump presidential campaign endorsements, 2016," https://en.wikipedia.org/wiki/List_of_Donald_Trump_presidential_campaign_endorsements,_2016, accessed May 7th, 2017.

[21] Fiverr.com, "Get everything you need starting at $5 - fiverr," 2014. [Online]. Available: https://www.fiverr.com/gigs/bots

[22] S.-C. Chen, M.-L. Shyu, and C. Zhang, "Innovative shot boundary detection for video indexing," in *Video Data Management and Information Retrieval*, S. Deb, Ed. Idea Group Publishing, 2005, pp. 217–236.

[23] X. Huang, S.-C. Chen, M.-L. Shyu, and C. Zhang, "User concept pattern discovery using relevance feedback and multiple instance learning for content-based image retrieval," in *Proceedings of the Third International Workshop on Multimedia Data Mining, in conjunction with the 8th ACM International Conference on Knowledge Discovery & Data Mining*, July 2002, pp. 100–108.

[24] X. Li, S.-C. Chen, M.-L. Shyu, and B. Furht, "Image retrieval by color, texture, and spatial information," in *Proceedings of the 8th International Conference on Distributed Multimedia Systems*, September 2002, pp. 152–159.

[25] L. Lin, G. Ravitz, M.-L. Shyu, and S.-C. Chen, "Video semantic concept discovery using multimodal-based association classification," in *Proceedings of the IEEE International Conference on Multimedia & Expo*, July 2007, pp. 859–862.

[26] L. Lin and M.-L. Shyu, "Weighted association rule mining for video semantic detection," *International Journal of Multimedia Data Engineering and Management*, vol. 1, no. 1, pp. 37–54, 2010.

[27] S.-C. Chen and R. Kashyap, "Temporal and spatial semantic models for multimedia presentations," in *Proceedings of the 1997 International Symposium on Multimedia Information Processing*, 1997, pp. 441–446.

[28] C. Chen, Q. Zhu, L. Lin, and M.-L. Shyu, "Web media semantic concept retrieval via tag removal and model fusion," *ACM Transactions on Intelligent Systems and Technology*, vol. 4, no. 4, pp. 61:1–61:22, October 2013.

[29] S.-C. Chen, M.-L. Shyu, and R. Kashyap, "Augmented transition network as a semantic model for video data," *International Journal of Networking and Information Systems*, vol. 3, no. 1, pp. 9–25, 2000.

[30] S.-C. Chen, M.-L. Shyu, and C. Zhang, "An intelligent framework for spatio-temporal vehicle tracking," in *Proceedings of the 4th IEEE International Conference on Intelligent Transportation Systems*, August 2001, pp. 213–218.

[31] S. Pouyanfar and S.-C. Chen, "Semantic event detection using ensemble deep learning," in *The IEEE International Symposium on Multimedia (IEEE ISM)*, CA, USA, 2016, pp. 203–208.

[32] Y. Yan, M. Chen, M.-L. Shyu, and S.-C. Chen, "Deep learning for imbalanced multimedia data classification," in *Proceedings of the 2015 IEEE International Symposium on Multimedia (ISM)*, December 2015, pp. 483–488.

[33] Y. Yan, M.-L. Shyu, and Q. Zhu, "Negative correlation discovery for big multimedia data semantic concept mining and retrieval," in *Proceedings of the 2016 IEEE Tenth International Conference on Semantic Computing (ICSC)*, February 2016, pp. 55–62.

[34] Y. Yan, Y. Liu, M.-L. Shyu, and M. Chen, "Utilizing concept correlations for effective imbalanced data classification," in *Proceedings of the IEEE 15th International Conference on Information Reuse and Integration*, Aug 2014, pp. 561–568.

[35] Y. Yan, M.-L. Shyu, and Q. Zhu, "Supporting semantic concept retrieval with negative correlations in a multimedia big data mining system," *International Journal of Semantic Computing*, vol. 10, pp. 247–268, 2016.

[36] Y. Yan, Q. Zhu, M.-L. Shyu, and S.-C. Chen, "A classifier ensemble framework for multimedia big data classification," in *Proceedings of the 2016 IEEE 17th International Conference on Information Reuse and Integration (IRI)*, July 2016, pp. 615–622.

[37] M.-L. Shyu, S.-C. Chen, and C. Haruechaiyasak, "Mining user access behavior on the www," in *Systems, Man, and Cybernetics, 2001 IEEE International Conference on*, vol. 3. IEEE, 2001, pp. 1717–1722.

[38] C. A. Davis, O. Varol, E. Ferrara, A. Flammini, and F. Menczer, "Botornot: A system to evaluate social bots," in *Proceedings of the 25th International Conference Companion on World Wide Web*. International World Wide Web Conferences Steering Committee, 2016, pp. 273–274.

[39] H. Abdi, L. J. Williams, and D. Valentin, "Multiple factor analysis: principal component analysis for multitable and multiblock data sets," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 5, no. 2, pp. 149–179, 2013.

[40] M. Bécue-Bertaut and J. Pagès, "Multiple factor analysis and clustering of a mixture of quantitative, categorical and frequency data," *Computational Statistics & Data Analysis*, vol. 52, no. 6, pp. 3255–3268, 2008.

[41] I. Jolliffe, *Principal component analysis*. Wiley Online Library, 2002.

[42] H. Abdi and D. Valentin, "Multiple correspondence analysis," *Encyclopedia of measurement and statistics*, pp. 651–657, 2007.

[43] N. Lauro, R. Verde, and F. Palumbo, "Factorial methods with cohesion constraints on symbolic objects," in *Data Analysis, Classification, and Related Methods*. Springer, 2000, pp. 381–386.

[44] F. Murtagh and P. Legendre, "Wards hierarchical agglomerative clustering method: which algorithms implement wards criterion?" *Journal of Classification*, vol. 31, no. 3, pp. 274–295, 2014.

[45] M. Breaban and H. Luchian, "A unifying criterion for unsupervised clustering and feature selection," *Pattern Recognition*, vol. 44, no. 4, pp. 854–865, 2011.

[46] S. Sadiq, Y. Yan, M.-L. Shyu, S.-C. Chen, and H. Ishwaran, "Enhancing multimedia imbalanced concept detection using vimp in random forests," in *2016 IEEE 17th International Conference on Information Reuse and Integration (IRI)*, July 2016, pp. 601–608.

[47] Y. Yan, M. Chen, S. Sadiq, and M.-L. Shyu, "Efficient imbalanced multimedia concept retrieval by deep learning on spark clusters," *International Journal of Multimedia Data Engineering and Management (IJMDEM)*, vol. 8, no. 1, pp. 1–20, 2017.

[48] T. Li, F. Li, and X. Zhang, "Prediction of kinase-specific phosphorylation sites with sequence features by a log-odds ratio approach," *Proteins: Structure, Function, and Bioinformatics*, vol. 70, no. 2, pp. 404–414, 2008.

[49] T. Shi and S. Horvath, "Unsupervised learning with random forest predictors," *Journal of Computational and Graphical Statistics*, vol. 15, no. 1, pp. 118–138, 2006.

[50] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: Analysis and implementation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 24, no. 7, pp. 881–892, 2002.

[51] N. R. Pal, K. Pal, J. M. Keller, and J. C. Bezdek, "A possibilistic fuzzy c-means clustering algorithm," *IEEE Transactions on Fuzzy Systems*, vol. 13, no. 4, pp. 517–530, 2005.

[52] T. K. Moon, "The expectation-maximization algorithm," *IEEE Signal processing magazine*, vol. 13, no. 6, pp. 47–60, 1996.

[53] J. A. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural processing letters*, vol. 9, no. 3, pp. 293–300, 1999.

[54] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[55] T. Meng and M.-L. Shyu, "Leveraging concept association network for multimedia rare concept mining and retrieval," in *Proceedings of the IEEE International Conference on Multimedia and Expo*, Melbourne, Australia, July 2012, pp. 860–865.