Predicting Poll Trends Using Twitter and Multivariate Time-Series Classification

Tom Mirowski, Shoumik Roychoudhury, Fang Zhou, and Zoran Obradovic(✉)

Center for Data Analytics and Biomedical Informatics,
Temple University, Philadelphia, USA
{tmirowski,shoumik.rc,fang.zhou,zoran.obradovic}@temple.edu

Abstract. Social media outlets, such as Twitter, provide invaluable information for understanding the social and political climate surrounding particular issues. Millions of people who vary in age, social class, and political beliefs come together in conversation. However, this information poses challenges to making inferences from these tweets. Using the tweets from the 2016 U.S. Presidential campaign, one main research question is addressed in this work. That is, can accurate predictions be made detecting changes in a political candidate's poll score trends utilizing tweets created during their campaign? The novelty of this work is that we formulate the problem as a multivariate time-series classification problem, which fits the temporal nature of tweets, rather than as a traditional attribute-based classification. Features that represent various aspects of support for (or against) a candidate are tracked on an hour-by-hour basis. Together these form multivariate time-series. One commonly used approach to this problem is based on the majority voting scheme. This method assumes the univariate time-series from different features have equal importance. To alleviate this issue a weighted shapelet transformation model is proposed. Extensive experiments on over 12 million tweets between November 2015 and January 2016 related to the four primary candidates (Bernie Sanders, Hillary Clinton, Donald Trump and Ted Cruz) indicate that the multivariate time-series approach outperforms traditional attribute-based approaches.

1 Introduction

Traditionally public opinion has been measured via costly and time-consuming polls. The rise of social media, such as Twitter, has provided people with a new way of making their voices heard. One recent event that has generated a significant amount of buzz within the Twitterverse is the 2016 U.S. Presidential election. From November 2015 to January 2016, over 12 million tweets have been sent directly to those vying for control of the United States. On one hand, these tweets are a rich data source of information regarding each of those candidates standings within the eyes of the potential voters [5]. On the other hand, this data poses challenges to making inferences from the tweets, which is particularly important for politicians, journalist, and so on. Using these tweets many questions can be answered: *can we accurately categorize a candidate as*

E. Spiro and Y.-Y. Ahn (Eds.): SocInfo 2016, Part I, LNCS 10046, pp. 273–289, 2016.
DOI: 10.1007/978-3-319-47880-7_17

improving or deteriorating? Whether the public sentiment strength, or just the tweet volume, could aid in the prediction? Whether the public opinion from the Twitter has a clear inference related to one political party more than the other?

These questions are addressed by applying multivariate time-series classification models. Therefore, the focus of this paper is quite different from other related work. Existing works [7,12,13,15] only consider the volume of (positive or negative) tweets. In doing so the temporal aspect of these features is lost. The methods utilized in this work, however, aim to extract discriminative patterns of the fluctuations in these features over time.

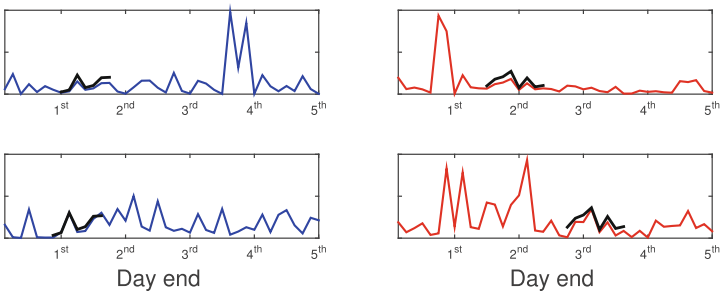
Tweets express public opinion, thus we study a variety of features that are related to sentiment strength, user support and tweet volume. Each feature is examined on an hour-by-hour basis. The superiority of using a time-series classification model is that distinguishable temporal patterns can be extracted for prediction. For example, in Fig. 1, the temporal patterns of time-series when a poll score increases (Fig. 1(a), colored in blue) is quite different from the ones when a poll score decreases (Fig. 1(b), colored in red). Summing the values of a feature within a time period into a single value results in the loss of this temporal correlation, as the summing process nullifies the temporal aspect of the data and how feature values changes over time.

The goal of this work is to predict the trend of poll scores of candidates, particularly in the case of the United States presidential campaign for 2016, by examining the people's voice. One commonly used approach is based on the majority voting scheme [10]. The main idea is to conduct predictions from individual univariate time-series first, and select the majority result as the prediction for the multivariate time-series. Applied in this paper is a model using this idea, called Majority-vote Learning Shapelets Algorithm (MLS). However, the majority voting scheme assumes that the univariate time-series from different features are equally important. This may not be true in many real-world cases. Therefore, another approach is considered, called Weighted Shapelet Transformation model (WST), which is a linear classifier that learns weights of patterns extracted from multivariate time-series.

The contributions of this work include: 1. Utilizing the temporal patterns in tweets, and formulating the problem as a multivariate time-series classification. 2. Identifying multiple features to characterize public opinions and examining their individual roles for prediction. 3. Proposing a model, WST, which outperforms MLS, univariate time-series classification models, and traditional attribute-based classification models. 4. From the extensive experimental results, we found that (a) not all features are equally important; (b) Trends in Democratic candidates are easier to predict than Republicans and (c) Features based on positive sentiment tend to have higher predictive prowess than their negative counterparts.

2 Related Work

Examined in this section is work related to two relevant topics: (1) Tweets related to elections; (2) Time-series classification.



(a) Poll score increases (b) Poll score decreases

Fig. 1. An example of time-series when (a) Poll score increases and (b) Poll score decreases. The temporal patterns (colored in black) are unique to each class. The temporal pattern in (a) has a trend of increase, and the temporal pattern in (b) has a trend of decline. (Color figure online)

2.1 Tweets Related to Elections

Support Strength. The first common theme is a focus on the support that comes from tweets [7] to determine the victor of an election campaign. For example, in [13] a linear regression model is applied to predict the value of a poll score. They consider tweet volume, overall user count, and unique user count as features within their regression model. The number of tweets about a candidate was shown to be significant in determining the outcome of election [15]. In [12], the number of unique users making posts in Twitter is considered as a feature for predicting the number of congressional seats that would be assigned to each of the political parties. U.S. politicians having proficiency in using Twitter to circulate positive information and favorable URLs about themselves is also noted [7].

Sentiment Strength. The second set of works in regards to predicting elections using Twitter focuses on the content of the tweets particularly the sentiment of them. It is noted that an improvement is observed in their regression models when incorporating sentiment from tweets [1]. Similarly, in [12] it is stated that they remove negative tweets as part of their data pre-processing step, implying they feel that negative information hurts their predictive model. Precedent established that an understanding of a candidates campaign can be achieved through analyzation tweet-content [15]. Time-series and sentiment are tied together in [9] demonstrating that the sentiment towards candidates over time is correlated to the fluctuations that can be observed in that candidates' poll score. For a detailed review of these works, please refer to [2].

Discussion. In this paper, many of the features mentioned above are used. The temporal element of these features is preserved, making use of their fluctuations over time. This work differs from other existing work in that (1) Instead of

predicting an actual value for the poll score, the focus is on the general trend that is expected to be presented by poll score. (2) Predictions are performed based on detected patterns (or sub-sequences) within time-series, as tweets are observed continuously. The previous work [9] identified direct correlation between public opinion and poll score, but did not offer any method for future prediction.

2.2 Time-Series Classification

Recently, in the realm of time-series classification, short segments within a time-series are used to characterize the time-series. The main idea is to extract sub-sequence, also known as shapelets [17], which are highly discriminative and have been used for classification purposes. Most works that are related to shapelet based time-series classification are univariate time-series classification [3,4,11, 16,17]. One of the state-of-the-art univariate time-series classification models is Learning Time-series Shapelet (LTS) model [4]. In the LTS model, shapelets [17] are learned jointly with a linear classifier rather than searching over all possible time-series segments, and the shapelet tranformed data [6] is used to classify unknown univariate time-series.

Only a few works have focused on the shapelet based multivariate time-series classification. One is to concatenate univariate time-series from multi sensor (or features) together to form a longer time-series [8]. The other work is based on the majority-voting scheme [10].

3 Data

3.1 Data Collection

Tweets were gathered every twenty-four hours using Twitter's official REST API in conjunction with the httr package for R 3.2.3. More than twelve million tweets were collected from November 12th, 2015 until January 10th, 2016, for a total of 60 days.

Tweets from the account of a U.S. Presidential candidate were collected. For this work, the focus is on four U.S. Presidential candidates, Bernie Sanders and Hillary Clinton from the Democratic Party, Donald Trump and Ted Cruz from the Republican Party. While we are examining public opinion, tweets from candidates themselves were kept throughout the analysis as their thoughts and opinions have the potential to guide the conversation surrounding them.

Additionally, tweets which contain a mention (@username) to a U.S. Presidential candidate's official Twitter username were extracted. The benefit of using mentions is two-fold. The first is that other users who may be looking at tweets about that individual will see your comment, extending your reach to others in the community. The second is that it serves as an alert to the individual whom you are targeting that they are receiving a message. In the event that a tweet is extracted multiple times due to multiple mentions, duplicated tweets (identified by their unique tweet ID) have been removed before analysis, ensuring a tweet

Table 1. Tweets break-down by candidate, and the number of univariate time-series that belong to each class using the *max* labeling scheme.

Candidate	# of Tweets	# of poll increases	# of poll decreases
Bernie Sanders	2,359,938	54	36
Hillary Clinton	2,056,540	27	54
Donald Trump	7,011,224	54	27
Ted Cruz	1,234,402	81	9

only counts once per candidate. Retweets are treated as their own unique tweet, signifying agreement with the original post. This work examines over 12 million tweets within a two month period. The breakdown of how the tweets were distributed among the candidates can be found in Table 1.

3.2 Features

In this section nine features, representing various aspects of tweets with reference to a given candidate across time, are discussed in detail. Features are divided into three distinct groups: sentiment strength, user support and tweet volume.

1. Sentiment Strength
Sentiment analysis was conducted by using the sentiment analysis software "SentiStrength" [14]. Each individual tweet, regardless of length, was treated as a single document. As per SentiStrength's algorithm, each tweet is given an overall positive score [1,5] and an overall negative score $[-1, -5]$. Tweets were classified as positive if the overall positive score was higher than the absolute value of the overall negative score. If the tweets contained matching sentiment strength for both levels, they were classified as neutral. Otherwise, they were classified as negative. For example, if a tweet was assigned the scores $[5, -3]$, then it was regarded as positive.

Sentiment strength is evaluated from two aspects: the average sentiment strength per tweet and the average sentiment strength per unique user.

- ***Positive Average*** **[PA]** refers to the average positive sentiment score among all tweets that were classified as positive within a user-specified time period, which is defined as Eq. 1. Let s_p represent the sentiment of any tweet p classified as positive, and n_h represent the total number of tweets classified as positive at time h.

$$f_h^{PA} = \frac{\sum s_p}{n_h} \tag{1}$$

- ***Negative Average*** **[NA]** thusly represents the average negative sentiment score among all tweets that were classified as negative within a user-specified time period.

– ***Unique Positive User Average*** **[UPUA]** represents the average sentiment among unique users who make positive postings with reference to the given candidate. First, the average sentiment of an individual user is calculated. Next, the average sentiment among all unique users is determined. Note that here only users whose tweets are all classified as positve were considered. This is represented in Eq. 2. Here $\overline{s^u}$ denotes the average sentiment score of a unique user u, and n_h^u denotes the number of unique users with positive sentiment at time h.

$$f_h^{UPUA} = \frac{\sum \overline{s^u}}{n_h^u} \tag{2}$$

– ***Unique Negative User Average*** **[UNUA]** thusly refers to the average sentiment among unique users who make negative postings with reference to the given candidate. Similarly, only users whose tweets are all classified as negative were considered.

2. User Support
The following two features are representative of the magnitude of individuals who show support for or against a candidate.

– ***Unique Positive Users*** **[UPU]** refers to the number of unique users who made a post that was classified as positive. This accounts for the fact that a single user may have made multiple posts, so they are only accounted for once.
– ***Unique Negative Users*** **[UNU]** similarly refers to the number of users who made a post that was classified as negative.

3. Tweet Volume

– ***Number of Tweets*** **[NT]** refers to the number of tweets that a candidate issued/received within a user-specified time period. This feature was reported as a good predictor for measuring interest in a candidate [7,12,15].
– ***Number of Positive Tweets*** **[NPT]** represents the number of tweets where the overall positive sentiment was stronger than the overall negative sentiment [1].
– ***Number of Negative Tweets*** **[NNT]** thusly represents the number of tweets where the overall negative sentiment was stronger than the overall positive sentiment [1].

3.3 Univariate and Multivariate Time-Series of Poll Score Trends

A univariate time-series $T_i^q = \{f_{i,1}^q, \cdots, f_{i,L}^q\}$ is a set of time-ordered observations f_h^q starting at time $h = 1$ and ending at $h = L$, each one being the value of feature f^q recorded at time h. The label of the univariate time-series T_i^q, denoted by Y_i^q, represents which class the univariate time-series belongs to.

The univariate time-series are characterized by two user-specified values: length and granularity. Five-days is chosen as the length of time-series, as it

is long enough to capture the potential changes in poll scores. Granularity represents how much information is shown by one time point. Since real-time tweets are collected, the granularity could be daily, hourly, and even by the minute. If granularity is too small, then the data will be very noisy and algorithms will overfit to detect localized temporal patterns (or sub-sequences) rather than identifying temporal patterns that give global discriminative power. If the granularity is too large, then the data will be very smooth losing potentially interesting patterns. In such a case the patterns would not be identified. In this work, 3 h is selected as granularity, because it splits the day into several time periods representative of different sections of the day, such as midday, evening or late-night. Therefore, each univariate time-series has 40 time points, with 8 points representing a single day.

A multivariate time-series $T_i = \{T_i^{PA}, T_i^{NA}, ...T_i^{NNT}\}$ is a set of univariate time-series that are related to individual features described in Sect. 3.2. Figure 2 shows the structure of our data. For each example $i \in I$, it represents a multivariate time-series $\{T_i^{PA}, T_i^{NA}, ...T_i^{NNT}\}$ with reference to a given candidate, where T_i^q represents a univariate time-series related to feature q.

The label of the multivariate time-series T_i, denoted by Y_i is same as the label of univariate time-series T_i^q. Therefore, this work predicts the label Y_i of the multivariate time-series T_i. This work only focuses on the cases when poll scores have obvious increment or decrement, that is, $Y_i \in \{+1, -1\}$. Two schemes of assigning labels are discussed in Sect. 3.4.

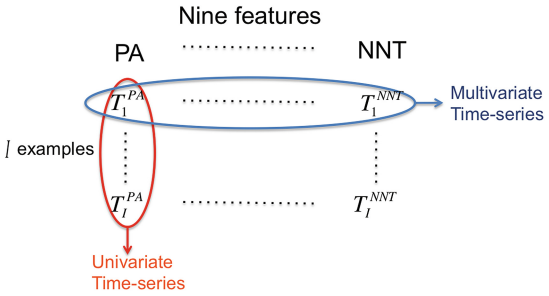


Fig. 2. Multivariate time-series data structure. T_i^{PA} represents the univariate time-series related to feature PA in the multivariate example i.

3.4 Labels of Time-Series

The research problem is a binary classification. That is, a label of 1 represents an increase in the candidate's poll score, and a label of -1 represents a decrease in the candidate's poll score. The daily average poll scores were collected from RealClearPolitics.com. These scores were used as they are an average of a wide-variety of national polls.

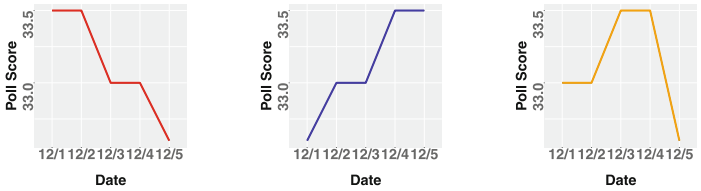


Fig. 3. Three possible poll score trends. (a) Poll scores decreased, so the label of this time period is −1. (b) Poll scores increased, so the label is 1. (c) Poll scores first increased, and then decreased. The label will be different based on the chosen scheme.

One method, which is referred to as *max*, consists of identifying when the maximum poll score occurs, and assigning an appropriate label based on it's position relative to the first and last poll score within the time period. Figure 3 illustrates three possibilities for generating a label. Figure 3(a) represents the scenario where the maximum poll score occurs on the first observed day, and then a label of −1 is assigned, representing a decrease in poll score. Figure 3(b) illustrates the case where the maximum observed poll score occurs on the last day. In this case, a label of 1 indicating a poll score increase is assigned. Figure 3(c) shows the situation where the maximum observed poll score happens on an internal date. Here, if the difference between the maximum value and the starting value is larger than the difference between the maximum value and the ending value, a label of 1 is assigned. If the difference between the maximum value and the ending value is larger, then a label of −1 is assigned. If the differences are equal, then the case is considered as neutral and is discarded from further analysis. In addition, if the poll score repeatedly fluctuates back and forth within the five-day window, the time-series will be discarded.

In comparison with the *max* label scheme, another labeling method, referred to as *count*, is used as a baseline. Creating these labels relies on counting the number of times that the poll score increases or decreases between any two consecutive days within the five-day window. If the poll score increases more frequently than it decreases (Fig. 3(b)), a label of 1 is assigned. If the poll score decreases more frequently than it increases (Fig. 3(a)), a label of −1 is assigned. If the poll score increases and decreases an equal amount, or never changes, then it is treated as a neutral case and is discarded from analysis. For example in Fig. 3(c), poll score increases once, and decreases once.

In total, after the removal of instances where a neutral label exists, we have 342 univariate time-series, that is 38 multivariate time-series. The distribution of these labeled univariate time-series instances are provided in Table 1. Assuming each time-series instance to be i.i.d, we generate three cross-validation datasets for evaluating both univariate and multivariate time-series classification performance, and report the mean performance results in the experiments.

4 Approach Overview

Two multivariate time-series classification models are presented in this Section. The first approach is based on majority-voting scheme. The second approach is a linear classifier which learns weights of patterns with respect to the class labels.

In both models, the first step is to utilize Learning Time-series Shapelet model [4] (denoted as LTS henceforth), one of the state-of-the-art univariate time-series classification models, to discover shapelets [17], which are local discriminative patterns (or sub-sequences) and are used to characterize the target class. The detailed explanation of LTS is provided in the appendix section.

4.1 Majority-Vote Learning Shapelets Model

Majority-vote Learning Shapelets (MLS) model is a majority-voting based algorithm, which effectively combines the benefits of majority-voting and the learning shapelets procedure of univariate time-series model LTS. MLS differs from LTS in that MLS aggregates the individual univariate time-series model predictions, as a univariate time-series related to a single feature often does not contain sufficient information for the prediction task.

In the framework of MLS, the LTS model is first applied to the individual univariate time-series T_t^q, and makes a prediction related to time-series T_t^q. Let t represent a multivariate time-series example in the test data, and q represent one of the features described in Sect. 3.2. The predicted value related to T_t^q is denoted as $\hat{Y_t^q}$. Then, the predicted value of the multivariate time-series example t is determined through the majority voting scheme, which selects the class that has more than half the votes. Equation 3 represents this approach mathematically.

$$\hat{Y_t} = max(\hat{Y_t^q}), \quad q \in \{PA, \cdots, NNT\} \tag{3}$$

4.2 Weighted Shapelet Transformation Model

MLS takes each univariate time-series independently and assumes that each univariate time-series have equal importance. To handle scenarios where importance is not equal, we propose one method, called Weighted Shapelet Transformation (WST), which is a linear classifier that learns weights of shapelets from multivariate time series.

In the framework of WST, the first step is to apply LTS to learn shapelets from the individual univariate time-series with respect to a particular feature. The learned shapelets can be considered as the attributes of multivariate time-series. The minimum distances between the learned shapelets and the observed time-series are calculated. These distance values are then used as the values of the attributes. The second step is to apply Logistic Regression to learn the weights of shapelets with respect to the target class.

5 Experimental Evaluation

In this section, the following questions are addressed:

Q1: How good are the predictions using multi-variate time-series? Does temporal modeling provide better predictive performance over traditional attribute-based modeling?
Q2: What are the prediction performances for individual features? Which features are good predictors?
Q3: Which labeling scheme best represents the poll trend?
Q4: Which political party is more predictable?

5.1 Experimental Setup

The experiments were conducted on 3 cross-validation datasets. Each contains 20 multivariate time-series used as training data, and 10 multivariate instances used as test data, after removing the 8 multivariate time-series that contain missing data. Internal cross-validation was conducted on the training sets in order to acquire the optimal hyper-parameters of the LTS model for shapelet extraction. These hyper-parameters were then used to train the entire training set. All training datasets were balanced in order to nullify bias in the learned model. For experiments related to individual features, the setting is same.

Evaluation Metric. Three evaluation measures were used in the conducted experiments: *sensitivity*, which refers to poll score increment detection rate; *specificity*, which refers to poll score decrement detection rate and *accuracy* which combines the sensitivity and specificity scores.

Baselines. In order to highlight the effectiveness of the proposed model, it is compared with multiple baselines. WST is compared to attribute-based models where singular values are obtained to represent the time-series in a given time period. The individual time points from univariate time-series related to a single feature are summed together within the five-day time period, and is represented as an attribute in both Logistic Regression and SVM. For features involving averages, values are adjusted using the number of positive or negative tweets and users accordingly. All attributes were then normalized. Comparisons with KNN are made to show the advantage of the shapelet-based classification over naive time-series classification methods. The proposed method is also compared against the commonly used majority vote scheme (MLS).

5.2 Performance of Multivariate Time-Series Classification

Examined here are the predictive power of two multivariate models, Majority-vote Learning Shapelets (MLS) and Weighted Shapelet Transformation (WST). Figure 4 shows that WST produces better results than MLS, with an accuracy of

70 %. This increased accuracy is due to WST learning weights for shapelets and considering univariate time-series related to different features are not equally important. MLS, which treats univariate time-series related to different features independently and equally important, has an average accuracy of 60 %. This gives evidence that not all features have similar predictive performance, which will be further discussed in Sect. 5.4.

Moreover, WST outperforms univariate time-series models with different features (see Table 2, Sect. 5.4). This leads the conclusion that making use of all features trumps focusing on individual features.

5.3 Time-Series vs. Attribute-Based Models

In this section, two multivariate time-series models, MLS and WST, are compared with two traditional attribute-based classification models, LR and SVM. Figure 4 clearly shows that on average both WST and MLS outperform both LG and SVM, and WST produces 20 % higher accuracy on average. This provides evidence that by utilizing the temporal nature of tweets, the time-series classification model produces better prediction results than traditional attribute-based models.

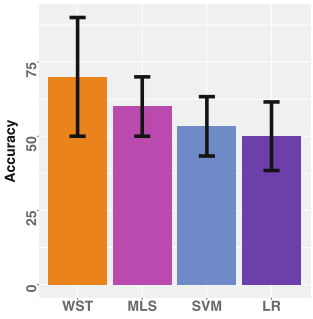
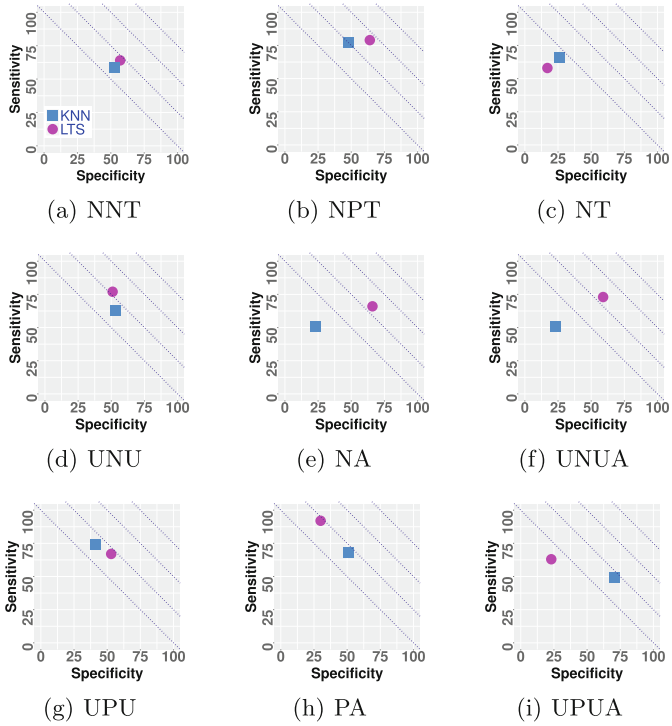


Fig. 4. Accuracy obtained from multivariate time-series models (WST, MLS) and attribute based models (SVM, LR).

5.4 Characterization of Individual Features

Next is to assess the predictive performance of each individual feature. The univariate time-series model, Learning Time-series Shapelet model (LTS) [4], was applied to each individual feature to perform prediction. For comparison, the baseline K-Nearest Neighbors (KNN) was applied, which only considers the Euclidean difference between time-series.

Features are analyzed in terms of sensitivity, specificity (Fig. 5) and accuracy (Table 2). Ideally, a method with perfect sensitivity and specificity would find



(a) NNT (b) NPT (c) NT

(d) UNU (e) NA (f) UNUA

(g) UPU (h) PA (i) UPUA

Fig. 5. The prediction performance of individual features defined in Sect. 3.2.

its corresponding icon located in the upper right corner. This region represents both a sensitivity and specificity of 100 %, perfect detection for both poll score increases and decreases. Increasing values along the x-axis indicates high true negative detection (e.g. poll score decreases). Increasing values along the y-axis indicates high true positive detection (e.g. poll score increases). Any method which finds its icon atop the main diagonal contains at least one, sensitivity or specificity, performing above random results.

LTS, indicated by a red circle, outperforms the baseline in most cases. Interesting patterns can be observed among these features. (1) In regards to the features dealing with tweet-volume, individual volume for "positive" and "negative" tweets are more accurate than the overall volume. (2) Features dealing with "negative" sentiment tend to be consistent predictors for poll increase and decrease. (3) The performance of features dealing with "positive" sentiment have more fluctuation. For example, both NPT and PA produce high accuracy, accuracies of 69.99 % and 63.33 % respectively (see Table 2), while UPUA falls just below random results. One possible justification for this is that Twitter users who post favorable information about a candidate likely have a vested interest in that candidate. Furthermore, the polls being examined in this dataset were from primary elections. It is possible that a significant portion of negative

things being posted about a candidate came from users across political party lines. While their voice is heard and potentially influential, their influence on primary polls is likely to be more limited, as primary polls tend to focus on users from within the party.

5.5 Different Labeling Scheme Comparison

In this section, the labeling scheme *max* is compared with the baseline labeling scheme *count*. In Table 2, the first nine rows compare the results across all nine features, and the last two lines compare the results of all features, utilizing both labeling schemes.

For all attributes, labels generated using the *max* scheme outlined in Sect. 3.4 provide significantly more accurate predictions, with some accuracies improving by up to 30 %. The reason for this increased accuracy is the *max* scheme being more representative of the actual changes that are occuring in the poll score. For example, using the *count* labeling scheme, three small downward trends in poll scores would outweigh one very large shift upwards. The *max* scheme, by using the differences that occur between the poll scores, takes into consideration the magnitude of the overall change that occured. As per these results, all other experiments in this paper make use of the labels generated from the *max* scheme.

The bottom two rows in Table 2 show the difference when using the *max* labeling scheme versus the *count* labeling scheme in the multivariate approaches discussed in this paper. The difference between the results from two labeling schemes is large. This shows that the performance of the MLS model remains highly dependant on the performance of the univariate time-series model on each feature. This is overcome by learning and utilizing the weights of shapelets.

Table 2. Mean accuracy (± standard deviation) obtained from individual features and all features with different labeling schemes

Features	*Max*	*Count*
NumNeg	**56.6 ± 9.4**	29.1 ± 15.4
NumPos	**63.3 ± 12.4**	37.5 ± 20.4
NumTweets	**30.0 ± 8.1**	45.8 ± 15.5
UniqNegUsers	**56.6 ± 4.7**	29.1 ± 5.8
NegAvg	**56.6 ± 9.4**	16.6 ± 11.7
UniqNegUserAvg	**56.6 ± 9.4**	20.8 ± 15.5
UniqPosUsers	**60.0 ± 14.1**	33.3 ± 5.8
PosAvg	**69.9 ± 8.1**	33.3 ± 11.7
UniqPosUserAvg	**40.0 ± 8.1**	25.0 ± 10.2
All features (MLS)	**60.0 ± 10.0**	25.0 ± 12.5
All features (WST)	**70.0 ± 20.0**	45.8 ± 7.2

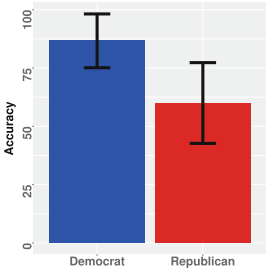


Fig. 6. Mean accuracy (± standard deviation) comparison between political parties using WST.

5.6 Characterization w.r.t. Political Party

Next we compare the prediction performance between the two primary political parties, Democrat and Republican. The model was trained and tested under two scenarios - Examining only Democrats, and examining only Republicans. The parameters of the underlying LTS model are fixed for shapelet extraction in both scenarios as the data is unbalanced within the individual parties. The accuracy for Democrat-party predictions were significantly higher.

Reflecting the election itself, there exists significant differences across political party lines. Utilizing WST the prediction accuracy is 26 % higher for Democrats. This result is not entirely shocking given the nature of the presidential race. The Democrat party tends to have a younger demographic that is more likely to take their discussion to Twitter. Furthermore, the dynamic of the election cycle was far more reactive on the Democratic side where both candidates were relatively close to one another, while the Republican race was much more one-sided in terms of polls.

While the results demonstrating the differences in predictive power between the two parties is very interesting, they should be taken with caution. One possible explanation is the difference in general patterns of poll scores between the Democratic candidates and Republican candidates. Donald Trump and Ted Cruz, both Republican, generally increased their poll score consistently across time. That is, the number of positive labels is much higher than the number of negative labels (see Table 1). Bernie Sanders and Hillary Clinton of the Democratic party experienced a much more dynamic race for the presidency in terms of poll scores. Both candidates experienced many more increases and decreases individually. Furthermore, Republican candidates on the whole had 1.8 times as many tweets as Democratic candidates as shown in Table 1, but were also less evenly distributed (Fig. 6).

6 Conclusion

In this paper, the primary research question was to address if a candidate's poll score trend could accurately be classified as increasing or decreasing using

only Twitter. The temporal nature of tweets was considered in this work. Nine different features were used to characterize public opinion (both positive and negative) and were examined temporally. These features were then used in two multivariate time-series classification models, MLS and WST. Over 12 million tweets were analyzed using these models to provide the answers to this question.

From our extensive experimental results, we conclude that: (1) Our proposed approach, WST, produces higher accuracy than the MLS model. (2) Time-series based classification models outperform traditional attribute based classification models. (3) Using distance-based metric, *max*, for creating labels outperforms the simple *count* scheme, and (4) There exists a difference between the predictability across political party lines on social media. With an accuracy of 70 % when using WST, Twitter can serve as a substitute to the time-consuming polling options that are traditionally used to gather information on public opinion.

Future work includes expanding the use of time-series classification in social media. Optimal combinations of parameters will be considered, rather than the one-or-all approach currently used. Additional factors present in social media will also be considered, such as favorites, shares, whether images were included in the post, and whether URLs were shared within the tweet.

Acknowledgments. This research was supported in part by NSF BIGDATA grant 14476570 and ONR grant N00014-15-1-2729.

Appendix

Learning Time-Series Classification Model (LTS)
LTS [4] is one of the state-of-the-art univariate time-series classification models. The method discovers short time-series sub-sequences known as shapelets [17], which are local discriminative patterns (or sub-sequences) that can be used to characterize the target class, for determining the time-series class membership. In the LTS model, shapelets are learned jointly with a linear classifier rather than searching over all possible time-series segments. More specifically, the algorithm jointly learns the weights of the classifier hyper-plane as well as the generalized shapelets.

A shapelet of length W is a sub-sequence of an instance of the time-series. There can be at most $L - W + 1$ sub-sequences, and each can be represented as $\{f^q_{i,j}, ..., f^q_{i,j+W-1}\}$. K shapelets are initialized using K-Means centroid of all segments.

Equation 4 represents a linear model, where $M_{i,k}$ is the minimum distance between the i-th series in T^q and the k-th shapelet S^q_k.

$$\hat{Y^q_i} = \beta_0 + \sum_{k=1}^{K} M_{i,k}\beta_k \qquad \forall i \in \{1, ..., I\} \tag{4}$$

The minimum distance $M_{i,k}$ is the predictor in this framework for shapelet learning and can be defined by a soft-minimum function:

$$M_{i,k} = \frac{\sum D_{i,k,j} e^{\alpha D_{i,k,j}}}{\sum e^{\alpha D_{i,k,j'}}} \tag{5}$$

where $D_{i,k,j}$ is defined as the distance between the j^{th} segment of series i and the k^{th} shapelet given by the formula

$$D_{i,k,j} = \frac{1}{W} \sum_{w=1}^{W} (T^q_{i,j+w-1} - S^q_{k,w})^2 \tag{6}$$

Equation 7 shows the regularized objective function, composed of a logistic loss defined by Eq. 8 and the regularization terms.

$$argmin_{S,\beta} F(S, W) = argmin_{S,\beta} \sum_{i=1}^{I} \mathcal{L}(Y^q_i, \hat{Y^q_i}) + \lambda_\beta ||\beta||^2 \tag{7}$$

$$\mathcal{L}(Y^q_i, \hat{Y^q_i}) = -\ Y^q_i\ ln(\sigma(\hat{Y^q_i})) - (1 - Y^q_i) ln(1 - \sigma(\hat{Y^q_i})) \tag{8}$$

Equation 7 is optimized using a stochastic gradient descent algorithm. The weights β and the shapelet S^q are jointly learned to minimize the objective function. Once the model is learned, classifying an unknown instance is simply computing $\hat{Y^q_t}$ for the t-th test instance of the q-th feature and determining the class label via Eq. 9

$$\hat{Y^q_t} \leftarrow argmax_{c \in \{1,-1\}}\ \sigma(\hat{Y^q_{t,c}}), \tag{9}$$

where $\sigma(\cdot)$ denotes the sigmoid function.

For more details about individual gradient computation of the objective function, the reader is referred to [4].

References

1. Bermingham, A., Smeaton, A.F.: On using Twitter to monitor political sentiment and predict election results. In: Sentiment Analysis where AI meets Psychology (SAAIP), p. 2 (2011)
2. Gayo-Avello, D.: A meta-analysis of state-of-the-art electoral prediction from Twitter data. Social Science Computer Review, pp. 649–679 (2013)
3. Ghalwash, M., Radosavljevic, V., Obradovic, Z.: Utilizing temporal patterns for estimating uncertainty in interpretable early decision making. In: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 402–411 (2014)
4. Grabocka, J., Schilling, N., Wistuba, M., Schmidt-Thieme, L.: Learning time-series shapelets. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2014, pp. 392–401. ACM (2014)
5. Graham, T., Jackson, D., Broersma, M.: New platform, old habits? Candidates use of Twitter during the 2010 British and Dutch general election campaigns. New Media Soc. **18**(5), 765–783 (2016)

6. Hills, J., Lines, J., Baranauskas, E., Mapp, J., Bagnall, A.: Classification of time series by shapelet transformation. Data Min. Knowl. Disc. **28**(4), 851–881 (2014)
7. Larsson, A.O., Moe, H.: Studying political microblogging: Twitter users in the 2010 Swedish election campaign. New Media Soc. **14**, 729–747 (2012)
8. Mueen, A., Keogh, E., Young, N.: Logical-shapelets: an expressive primitive for time series classification. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2011, pp. 1154–1162 (2011)
9. O'Connor, B., Balasubramanyan, R., Routledge, B.R., Smith, N.A.: From Tweets to polls: linking text sentiment to public opinion time series. ICWSM **11**(122–129), 1–2 (2010)
10. Patri, O.P., Sharma, A.B., Chen, H., Jiang, G., Panangadan, A.V., Prasanna, V.K.: Extracting discriminative shapelets from heterogeneous sensor data. In: 2014 IEEE International Conference on Big Data, Big Data 2014, Washington, DC, USA, 27–30 October 2014, pp. 1095–1104 (2014)
11. Roychoudhury, S., Ghalwash, M.F., Obradovic, Z.: False alarm suppression in early prediction of cardiac arrhythmia. In: 2015 IEEE 15th International Conference on Bioinformatics and Bioengineering (BIBE), pp. 1–6 (2015)
12. Sang, E.T.K., Bos, J.: Predicting the 2011 Dutch senate election results with Twitter. In: Proceedings of the Workshop on Semantic Analysis in Social Media, pp. 53–60. Association for Computational Linguistics (2012)
13. Shi, L., Agarwal, N., Agrawal, A., Garg, R., Spoelstra, J.: Predicting us primary elections with Twitter (2012)
14. Thelwall, M., Buckley, K., Paltoglou, G.: Sentiment strength detection for the social web. J. Am. Soc. Inform. Sci. Technol. **63**(1), 163–173 (2012)
15. Tumasjan, A., Sprenger, T.O., Sandner, P.G., Welpe, I.M.: Predicting elections with Twitter: what 140 characters reveal about political sentiment. ICWSM **10**, 178–185 (2010)
16. Xing, Z., Pei, J., Yu, P.S., Wang, K.: Extracting interpretable features for early classification on time series. In: SIAM International Conference on Data Mining, pp. 247–258 (2011)
17. Ye, L., Keogh, E.: Time series shapelets: a new primitive for data mining. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2009, pp. 947–956. ACM (2009)