Soc. Netw. Anal. Min. (2014) 4:193
DOI 10.1007/s13278-014-0193-5

ORIGINAL ARTICLE

Predicting political preference of Twitter users

Aibek Makazhanov · Davood Rafiei · Muhammad Waqar

Received: 31 January 2014 / Revised: 30 March 2014 / Accepted: 18 April 2014

Abstract We study the problem of predicting the political preference of users on the Twitter network, showing that the political preference of users can be predicted from their Twitter behavior towards political parties. We show this by building prediction models based on a variety of contextual and behavioral features, training the models by resorting to a distant supervision approach and considering party candidates to have a predefined preference towards their respective parties. A language model for each party is learned from the content of the tweets by the party candidates, and the preference of a user is assessed based on the alignment of user tweets with the language models of the parties. We evaluate our work in the context of two real elections: 2012 Albertan and 2013 Pakistani general elections. In both cases, we show that our model outperforms, in terms of the *F*-measure, sentiment and text classification approaches and is at par with the human annotators. We further use our model to analyze the preference changes over the course of the election campaign and report results that would be difficult to attain by human annotators.

Keywords Social network · Twitter · Political elections · User preference

A. Makazhanov
Nazarbayev University Research and Innovation System,
Astana, Kazakhstan
e-mail: aibek.makazhanov@nu.edu.kz

D. Rafiei (✉) · M. Waqar
University of Alberta, Edmonton, AB, Canada
e-mail: drafiei@ualberta.ca

M. Waqar
e-mail: mwaqar@ualberta.ca

1 Introduction

Today Twitter stands out as one of the most popular microblogging services, where information propagates in no time, and words and actions trigger immediate responses from users. Such an environment is ideal for advertising political views, especially during the heat of election campaigns. Political discourse on Twitter has been studied over the past few years. The focus of studies varied from analyzing social networks of candidates and politically active users (Livne et al. 2011; Conover et al. 2011b) to predicting results of actual elections (Marchetti-Bowick and Chambers 2012; Tumasjan et al. 2010; O'Connor et al. 2010). However, with few exceptions (Golbeck and Hansen 2011; Conover et al. 2011a), most of the previous work focused on the analysis of individual tweets (Choy et al. 2011; Marchetti-Bowick and Chambers 2012; Wang et al. 2012) or aggregate properties of a corpus of tweets (O'Connor et al. 2010; Tumasjan et al. 2010), and not on the political preference of individual users.

In the present work, we address the problem of predicting political preference of users given the record of their past activity on Twitter. We believe that approaches to political discourse analysis could benefit from the knowledge of political alignment of users. For instance, predicted user preferences can be used as noisy labels when evaluating approaches to the community mining in political communication networks (Conover et al. 2011a, b). Similarly, predicted political affiliations of users can be used as additional features in methods that focus on the extraction of political sentiments (O'Connor et al. 2010; Marchetti-Bowick and Chambers 2012). Same goes for real-time analysis systems that could, in principle, use trained models to predict user preferences "on the fly" (Ratkiewicz et al. 2011; Wang et al. 2012). Finally, when used

Published online: 06 May 2014

in the context of elections (Marchetti-Bowick and Chambers 2012; Tumasjan et al. 2010; O'Connor et al. 2010), political preference prediction has implications in better understanding changes in public opinion, and possible shifts in popular vote.

We set the preference prediction problem in the context of Alberta 2012 general election, and consider a vote for a party to be the best indicator of the political preference of users. More formally, the problem is modeled as a multilabel classification task, where given a user and a set of parties, the goal is to predict which party, if any, the user is most likely to vote for. For each party considered in this work, we construct a profile, which includes a ranked list of weighted party-specific topics. Also, for each party, we train a binary classifier, using the partisan identifications of the party candidates as the ground truth. For a given user–party pair, such a classifier provides a confidence with which a user can be considered a party supporter. Thus, a party with the highest predicted confidence is considered as the most preferred one.

We evaluate our method on a set of users whose political preferences are known based on the explicit statements (e.g., *'I voted NDP today!'*) made on the election day or soon after. Measuring the performance on a per-party basis in terms of precision, recall and *F*-measure, we compare our method to human annotators, sentiment and text classification approaches, and to chance. We found that although less precise than humans, for some parties, our method outperforms human annotators in recall. Another experiment, where we analyzed how preferences of users change over time, revealed that politically active or so-called *vocal users* are less prone to changing their preferences than users who do not get actively involved, i.e., *silent users*. We also observed that the dynamics of the popular vote shift among silent users closely resembles that of the actual election.

To check if the proposed method is readily applicable to any given election, and to address a concern of a possible election-dependent bias, we evaluated our method in the context of another election, specifically 2013 Pakistani general election. Choosing this particular election allowed us to address the issue of the language dependency of some of the features as tweets in Urdu and Roman Urdu (transliteration) were encountered. We found that the performances of our method across the elections are consistent. Also, the change in user population and nearly 1-year long temporal gap between the two elections suggests that our method is robust with respect to the ever-changing behavior of Twitter users.

This paper is organized as follows. In the remainder of this section, we briefly outline our contributions and then provide background on Albertan election because the remaining sections have references to the election. We discuss the related work in Sect. 2. In Sect. 3, we give the definition of user–party interactions, and explain the procedure of building the interaction profiles of the parties. Following that, we describe our data collection and cleaning procedures in Sect. 4. In Sect. 5, we explain the essence of the method and discuss the experiments and the results. In Sect. 6, we analyze how the predicted political preference of users changes during the election campaign. We evaluate our method on a different election in Sect. 7 and analyze some of the election-related issues. Finally, in Sect. 8, we draw conclusions and discuss future work.

1.1 Contributions

Our contributions can be summarized as follows:

- we introduce a notion of a user–party interaction, based on which we propose an interaction-driven approach to the problem of predicting the political preference of users;
- we explore the dynamics of the election campaign, showing the difference in preference change across different types of users;
- of the studies concerned with a Twitter-based political discourse analysis, our work is, to the best of our knowledge, the first to report an extensive data cleaning effort;
- we evaluate our proposed methods in a real setting with data covering the tweets posted during an election campaign and in connections to real events;
- we demonstrate that our method performs comparably to human annotators for two different elections, and shows no sign of election-specific bias.

1.2 Background of the 2012 Alberta General Election

On April 23, 2012, a general election was held in Alberta, Canada to elect 87 members of the Legislative Assembly. The event was highly anticipated[1] as according to polls, for the first time since 1971, the ruling Progressive Conservative (PC) party could have lost the election. Since 2009, Wildrose Alliance party (WRA) started to rapidly gain popularity, and by the beginning of the 2012 election, they were leading in polls and becoming the main challenger to PC. Two other major parties who nominated 87 candidates, one per each riding, were Alberta Liberals (LIB) and New Democratic Party (NDP). Parties with low polling numbers and popular vote share (e.g., Alberta party) are not considered in this work. To form a majority government, a

[1] 57 % Voter turnout, the highest since 1983: http://www.cbc.ca/news/canada/albertavotes2012/story/2012/04/24/albertavotes-2012-voter-turnout.html.

party was required to win at least 44 seats. The election resulted in Conservatives winning one seats and defending their ruling party status. Wildrose followed with 17 seats, forming the official opposition. Liberals and NDP won five and four seats, respectively. Although WRA had lost almost 10 % of the popular vote to PC, their candidates were second in 56 ridings, losing by a tight margin in dozens of constituencies[2].

2 Related work

Our work is related to a vast body of research dedicated to the problem of extracting political sentiment from Twitter. To address the problem, numerous works employ a two-phase content-driven approach, where at the first phase a set of relevant tweets is identified, and at the second phase an actual sentiment is extracted. Typically, a tweet is considered relevant to a given topic, if it contains at least a term from a list of target keywords constructed manually (Wang et al. 2012; O'Connor et al. 2010; Tumasjan et al. 2010), or semi-automatically (Conover et al. 2011a, b) by expanding a seed set. Once a set of relevant tweets is identified, various supervised or unsupervised methods are employed to extract the polarity of expressed sentiments. Unsupervised methods rely on the so-called opinion lexicons, lists of "positive" and "negative" opinion words, estimating a sentiment polarity based on the positive-to-negative words ratio (O'Connor et al. 2010) or just the raw count of opinion words (Choy et al. 2011). More sophisticated approaches employ supervised learning, and train prediction models either on manually labeled tweets (Conover et al. 2011a; Wang et al. 2012) or on tweets with an emotional context (Marchetti-Bowick and Chambers 2012), i.e., emoticons and hashtags, such as *:-)*, *#happy*, *#sad*, etc.

Conover et al. (2011a) took a two-phase approach to the problem of predicting political alignments of users. At the first phase, using a co-occurrence-based ranking, the authors expanded a set of two widely used political hashtags to a list of 66 target keywords, and extracted more than 250,000 relevant tweets. Following that, the tweets were grouped on a per-user basis, and SVM models were built on unigram features. As a ground truth, the authors used a random set of 1,000 users whose political affiliations were identified based on a visual examination of the content of their tweets. The authors reported an accuracy of about 79 %, which could be boosted up to almost 91 % when the features were restricted to hashtags only. In contrast, in our work, we use a keyword search only to get test data, and our prediction models are trained on the content generated by known accounts of party candidates. Also, apart from using content-based features, we use a rich set of behavioral features, such as the frequency of user–party interactions, and *following* and *retweeting* preferences.

A number of works chose a different methodology, employing an interaction-driven approach to extract sentiments expressed implicitly in the form of preferential following (Sparks 2010; Golbeck and Hansen 2011), and retweeting (Conover et al. 2011b). While Sparks (2010) showed that most of the time users tend to preferentially follow only those politicians whose ideological believes they share, Golbeck and Hansen (2011) showed that average political preference of media followers reflects political bias in media. Similarly, Conover et al. (2011b) showed that users preferentially retweet only those tweets that agree with their own political views. As we will show later, such a behavior is also observed among candidates of political parties.

In terms of the definitions of user–party interactions and interaction-based features, our work relates to that of Sepehri et al. (2012), who explored the correlation between the interaction patterns of Wikipedia editors and their votes in admin elections. While the authors defined an interaction as an act of co-revising of a page by a pair of contributors, we extend the proposed notion to the Twitter environment, and treat tweeting on party-specific topics as a form of user–party interactions.

Finally, a preliminary version of this work appeared in Makazhanov and Rafiei (2013). Our early work reported experiments on the 2012 Albertan election only. Clearly, some relevant questions were whether our method was general enough for our findings to be applicable to other elections and robust enough to withstand constantly changing behavior of Twitter users. The current paper addresses these questions while extending some of our presentations and discussions. In particular, the newly inserted Sect. 7 studies the problem in the context of the recent Pakistani election with further analysis and comparisons to the Albertan election.

3 User–party interactions

Not all postings of users reflect their political preference. One straightforward approach to filter out irrelevant tweets is to identify popular hashtags from the local political trends and consider any tweet containing those tags to be of relevance. However, tweets that use the same hashtag(s) may not be relevant to the same degree. For instance, although both tweets in the example below contain *#Cdnpoli* (Canadian politics) hashtag, the second one is clearly of more relevance to the election.

[2] Alberta general election, 2012: http://en.wikipedia.org/wiki/Alberta_general_election,_2012.

T1: *Huzzah! Happy birthday to Her Majesty, Queen Elizabeth II, Queen of Canada. Long may she reign! #Cdnpoli*

T2: *A win for @ElectDanielle will be the first step towards the libertarian #CPC regime under Max "The Axe" Bernier. #ABvote #Cdnpoli*

On the other hand, Twitter activities, such as (re)tweets, mentions, and replies, that concern parties or its candidates, may suggest the political preference (or its absence) of users. To detect a preference expressed in this way, for each party, we compile a ranked list of weighted terms, or the *interaction profile* of a party, and define a user–party interaction as follows:

Definition 1 Given a user u, a party p, and a tweet t, we say t interacts with p if it contains at least one term from the interaction profile of p. Similarly, u interacts with p if the user has at least one tweet that interacts with p.

Note that we use the term *interaction* for the following two reasons: (i) we assume that collective behavior of users triggers some kind of response from political entities, hence interactions take place; (ii) the term *interaction* makes exposition much more concise, than perhaps a more accurate term *a directed behavior of users towards parties*.

Building interaction profiles. It is reasonable to assume that during a campaign, party candidates will use Twitter to advertise central issues for their parties, discuss television debates and criticize their opponents. We aim at utilizing the content resulted from such a behavior to capture party-specific topics in the form of weighted unigrams and bigrams.

Given a set C of candidates, we associate with each candidate $c \in C$, a document dc that consists of all postings of candidate c; in our work, dc is modeled as a *bag of words*. Let $D = \{dc \mid c \in C\}$, and for each party $p \in P$, $Dp = \{dc \mid c \text{ is a member of p}\}$. We denote the vocabulary of D with V. To build a language model (LM) for each party, we use a term-weighting scheme similar to Livne et al. 2011 where the Kullback–Leibler (KL) divergence between the LM probabilities of the party and the LM probabilities of all parties gives a measure of importance to terms in the party profiles. To be consistent, we refer to the general corpus as the election corpus and to its LM as the election LM. Similarly, we refer to a collection of documents associated with a party as a party corpus and its LM as a party LM. Term probabilities in the aforementioned language models are calculated using tf-idf scores. The marginal probability of a term $t \in V$ in the election LM is calculated as

$$P(t|D) = \overline{\mathrm{tf}}(t, D)\mathrm{udf}(t, D)$$

where $\overline{\mathrm{tf}}(t, D)$ denotes the average term frequency in a document in D, and $\mathrm{udf}(t, D) = \mathrm{df}(t, D)/|D|$ denotes the probability of t appearing in a document in D. Here, $\mathrm{df}(t, D)$ is the document frequency of t in D.

For the party LMs, initial term weights are calculated as

$$w(t|p) = \overline{\mathrm{tf}}(t, D_\mathrm{p})\mathrm{udf}(t, D_\mathrm{p})\mathrm{idf}(t, D)$$

where $\mathrm{idf}(t, D)$ is the inverse document frequency of t over the set of all documents D. Then the obtained weights and scores are normalized:

$$P^N(t|D) = \frac{P(t|D)}{\sum_{t \in V} P(t|D)}; \quad w^N(t|p) = \frac{w(t|p)}{\sum_{t \in V} w(t|p)}$$

and further smoothed to account for terms that are not observed in the party corpora,

$$w^S(t|p) = (1 - \lambda)w^N(t, p) + \lambda P^N(t|D)$$

with the normalization factor λ set to 0.001. Finally, we calculate the probability of term $t \in V$ in the LM of party p as

$$P(t|p) = \frac{w^S(t|p)}{\sum_{t \in V} w^S(t|p)}.$$

Now, we can calculate the KL divergence between probability distributions of party LMs and the election LM as follows:

$$KLp(P(t, p)||P(t, D)) = \sum_{t \in V} P(t|p) \ln \frac{P(t|p)}{P(t|D)}$$

However, rather than sum, which characterizes the overall content difference, we are much more interested in the individual contribution of each term. Hence, the final weight of term $t \in V$ in the interaction profile of party p, or the importance score of the term is calculated as

$$I(t, p) = P(t|p) \ln \frac{P(t|p)}{P(t|D)}$$

The higher the weight of the term, the more it deviates from the "common election chatter", and as such becomes more important to the party.

As the final step in building the interaction profiles, we chose 100 top-ranked bigrams and unigrams representing party hashtags, Twitter user names of candidates, and the last names and the given names of party leaders. Note that by including candidate and party account names into the profiles, we naturally count tweets that mention, reply or retweet candidates as interactions, i.e., our definition is flexible, and conveys standard forms of user activities on Twitter. Moreover, the fact that the profiles are weighed and ranked allows us to weigh and rank interactions. Specifically, given a user u and a party p, the weight of a u–p interaction is calculated as the collective weight of all terms from the interaction profile of p found in the interaction. Correspondingly, the average, minimum and

Table 1 Basic characteristics of the interaction profiles of the parties

Party	Profile size	Top terms
LIB	170	*Alberta liberals, vote strategically*
NDP	157	*Orange wave, renewable energy*
PC	173	*Premier alison, family care*
WRA	182	*Energy dividend, balanced budget*

maximum ranks are calculated, respectively, as the average, minimum and maximum ranks of such terms. Table 1 gives the number of terms in the interaction profiles of the parties and lists some of the top weighted bigrams.

4 Data collection and cleaning

To build the interaction profiles of the parties, we need to identify Twitter accounts of the candidates and obtain their tweets posted during the campaign. We collected candidate accounts and verified that they belonged to the actual candidates. We also collected non-candidate accounts by monitoring the campaign-related trends over the course of 10 days before the election. As a preparation step, we identified and removed non-personal accounts, e.g., media, organizations, businesses, fan clubs. On the day after the election, for each account of the two groups of users, we downloaded as many tweets as Twitter API would allow. Thus, candidate and non-candidate accounts and their tweets are used as the training and the testing data, respectively. In the following subsections, we give a detailed description of our data collection and cleaning methodology.

4.1 Collecting user accounts

We semi-automatically collected Twitter accounts of candidates. As in Livne et al. (2011), we retrieved the first three google search results for each candidate name and the *twitter* keyword, and manually verified the accounts. Additionally, if we could not find or verify a candidate account from the search results we looked up the official website of a candidate party. This way we collected 312 Twitter accounts of 429 registered candidates. Of those, 252 belonged to candidates of the four major parties considered in this work. To that list, we also added the official Twitter accounts of the parties to have a total of 256 accounts. To collect non-candidate accounts, we monitored campaign-related trends over the course of 10 days prior to the election, using a manually selected 27 keywords, such as party hashtags, party names, leader names and general election hashtags. As a result, we obtained 181,972 tweets by 28,087 accounts of which 27,822 belonged to non-candidates. Removing accounts with reported communication language other than English left us with 24,507 accounts. For each of these non-candidate and 256 candidate accounts, we retrieved up to 3,000 tweets. The retrieved content was limited to the tweets posted since March 27, 2012, the official first day of the campaign[3].

4.2 Data cleaning

Benevenuto et al. (2010) have shown that spammers often “hijack” popular hashtags and trends to reach a larger target audience. To test this hypothesis, we trained SVM and logistic regression (Lgstc) models based on the data set and the top ten features in Benevenuto et al. (2010)[4], but to account for the specifics of local trends in our data set, we extracted features for candidate accounts (the only ground truth we had) and added them to the data set as positive examples of non-spam accounts. The features were designed to distinguish the behavior and the quality of the content generated by the two types of users, i.e., spammers and non-spammers. For instance, spammers tend to include URLs (spam links) in almost every tweet, label their tweets with popular and often unrelated hashtags, and rarely reply to tweets of other users. We performed tenfold cross-validation and found that only 1 out of 256 candidate accounts was misclassified as spam account. In general, spammers were detected with lower F-measure than non-spammers (in agreement with the original work). The Lgstc classifier performed slightly better than SVM, and F-measure for respective classes were 85 and 94 %. However, results of spammers detection across the entire data set were rather unexpected. Out of 24,507 accounts, 74 or 0.3 % were labeled as spam. We manually checked these accounts and did not find any evidence of spam-related activity. As a rough estimate of misclassification of spammers as non-spammers, we examined a random sample of 244 or 1 % of 24,433 accounts labeled as non-spam. Again, we did not find any spammers apart from two accounts that were already suspended by Twitter.

From the conducted experiment, we drew the following conclusions: first, our model may have been subject to a certain degree of overfitting, given that training data were collected much earlier than our data set and the behavior of spammers might have changed. Second, it could be that local political trends were not attractive enough for spammers and our data set may naturally contain negligibly low number of spam accounts. Third, certain accounts

[3] http://en.wikipedia.org/wiki/Alberta_general_election_2012#Timeline.

[4] The authors have shown that the top ten features perform as good as the full set of features.

behave like spam accounts and generate content that shares some features with spam tweets.

When we were verifying accounts labeled as spam, we noticed that some of them represented *media* (e.g., provincial and federal news papers, radio stations), *businesses* (e.g., companies, real estate agencies), and even *official entities* and *organizations* (e.g., City of Calgary, Canadian Health Coalition). Some of the accounts had high ratio of URLs per number of tweets, while others had low numbers of generated and received replies. One common characteristic of these accounts is that almost none of them expresses a *personal opinion* of a potential voter. Moreover, owners of these accounts often do not or should not have political preference (at least media and official entities). While investigating media bias and influence or support and disapproval of unions is an interesting research direction, in the present work, we focus on predicting the political preference of individuals. We will refer to individual accounts as personal or P-accounts, and correspondingly non-personal accounts will be referred to as NP-accounts.

Removing NP-accounts. We approached the problem by extending the method that we used for spammers detection. For training we used the set of accounts that we annotated during the evaluation of spammers detection. The set contained 161 P-accounts and 25 NP-accounts. To get more NP-accounts for training, we extracted the list of Albertan newspapers[5] and searched the data set for accounts with names matching those in the list. We did the same for the list of Alberta cities[6]. We annotated the extracted set of accounts and obtained a final data set that consisted of 161 P- and 132 NP-accounts. We started with the feature set used in the spammers detection task, but after a tenfold cross-validation, we revised the set, removing some irrelevant features. Also, our observations suggested that personal accounts frequently contain first-person singular pronouns (*I, me, mine, myself*, etc.) and words like *father, mother, wife* or *husband* in the account description. Moreover, account names differ drastically with location names, abbreviations and business-related terms being frequently used by NP-accounts, and personal names being frequently used by P-accounts. To capture these differences in the types of accounts, using the training data, we build unigram language models for the names and the descriptions of P- and NP-accounts. After a feature selection procedure, our final model consisted of 11 features. We classified the entire data set, and labeled 535 out of 24,507 accounts as NP. A visual inspection of these accounts revealed that in 447 cases, the model made correct predictions yielding 83 % precision. Out of 447 NP-accounts, 160 or 36 % were associated with media. As a final cleaning step, we removed NP-accounts from the data set leaving 24,060 accounts. All the experiments reported further were conducted on this cleaned data.

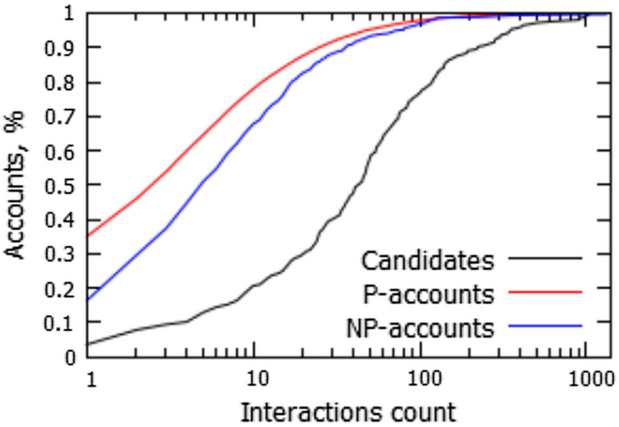


Fig. 1 Cumulative percentage of accounts at each interaction count for candidate, P- and NP-accounts

Table 2 Data set properties

User group	Accounts	Interactions	Interactions per account
Candidates	256	23,195	90.6
LIB	62	7,916	127.7
NDP	50	5,813	116.3
WRA	73	6,029	82.6
PC	71	3,437	48.4
P-accounts	24,060	311,443	12.9
NP-accounts	447	8,359	18.7

4.3 Discussion

Figure 1 shows the cumulative percentage of accounts that have generated at least one election-related tweet. Data are presented for different types of accounts across varying counts of interactions. The plot clearly illustrates the silent majority effect (Mustafaraj et al. 2011), showing that almost half of all P-accounts have produced at most only two interactions. Moreover, it shows that concentration of the silent majority is not even across different target groups, with representatives of media and organizations containing less silent users relative to individuals. Needless to say that candidates engaged in more interactions with about 20 % of candidate accounts producing more than 100 tweets. For the other two groups of accounts this ratio hardly reaches 5 %. However, CDF of NP-accounts grows slower than that of P-accounts and this is expected given that 35 % of NP-accounts belong to media sources.

Table 2 contains interaction statistics for different user groups, and statistics for candidates is further split on a per-

[5] http://en.wikipedia.org/wiki/List_of_newspapers_in_Alberta.

[6] http://en.wikipedia.org/wiki/List_of_cities_in_Alberta.

party basis. We can see a similar trend, with candidates having much more interactions per account than the other two user groups, and owners of NP-accounts having 1.4 times as much interactions per account than individuals, i.e., owners of P-accounts. Considering that the bulk of the interactions of NP-accounts is either news or statements without personal political preference, we conclude that our efforts in data cleaning had paid off and we were able to remove certain amount of noise.

5 Predicting political preference

We state the problem of predicting political preference of a user as follows:

> Given a user u and a set of parties P with whom u has interactions, predict which party $p \in P$, if any, u is most likely to vote for in the upcoming election.

To address the problem, we employ a one vs. all classification strategies, where for each party we train a binary classifier. Given the interaction statistics of a user–party pair, a classifier trained for the corresponding party provides the confidence with which this user may be considered a supporter of that party. If a user has interacted with multiple parties, confidence scores are compared and the user is said to prefer a party which was given the highest confidence score. Ties are broken randomly. According to our problem statement, a user may prefer none of the parties she interacted with. This special case is implemented by setting the confidence threshold t. Thus, for a given interacting user–party pair u–p, if the confidence provided by the corresponding classifier is lower than the threshold, we conclude that u will not support p in the election. If this is the case for all parties $p \in P$, user u is said to prefer none of the parties. In all of our experiments, we use a standard threshold of $t = 0.5$.

Let us now explain the training procedure. For each candidate account in our data set, we extract interactions, group them on a per-party basis, and build a feature vector for each group of interactions, as described in the following subsection. For instance, if candidate c from party p_1 has interacted with parties p_2, p_3, and, of course, her own party p_1, then feature vectors c–p_1, c–p_2, and c–p_3 will be created and fed, respectively, to classifiers C_1, C_2, and C_3. In this case, feature vector c–p_1 is considered a positive training example, and the remaining two feature vectors represent negative training examples, for, in all probability, candidate c would not have supported parties p_2 and p_3.

5.1 Building prediction models

We design the features of the models based on the interaction records and the behavior of users, and build a feature vector for each interacting user–party pair. Clearly, if a user does not interact with any party, we can make no predictions, and such a user is considered to prefer none of the parties. To capture basic patterns of user–party interactions, we gather the following statistics for each user–party pair: (i) raw count of the interactions, (ii) average weight, (iii–v) average/min/max rank, (vi, vii) average length (in characters, in words), (viii) average frequency (interactions per day), (ix) number of positive terms per interaction, (x) number of negative terms per interaction, (xi) average offset of a party hashtag in the postings of the user, (xii) political diversity, i.e., number of parties the user has interactions with. For positive and negative terms, we use an opinion lexicon of 6,800 words compiled by Hu and Liu (2004) and the Wikipedia's list of emoticons[7]. To account for Twitter-specific behavior of users towards parties, our features also include: (i) number of times a user has retweeted party candidates, (ii) average retweet time, (iii) number of replies a user received from party candidates, (iv) average reply time, and (v) number of party candidates that a user follows. Here, retweet time corresponds to the amount of time, in seconds, passed between the moment when the original tweet of a candidate was created and the time the retweet was made. Reply time is calculated similarly.

Domain	Value in $u - p_1$	Value in $u - p_2$
T	6	14
O	20	20
R	0.3	0.7
Δ	-14	-6

Fig. 2 An example calculation of values of the *interactions count* feature over different domains

Feature domains. Recall that a user can interact with multiple parties. To provide prediction models with the *overall* statistics, calculated for all parties with whom a user has interacted, and the *relative* statistics, calculated for a target party in relation to all parties, we define features over different *domains*. Let us explain a concept of feature domains on the example shown on Fig. 2. Suppose user u interacted 6 times with party p_1 and 14 times with party p_2. First of all, we create two feature vectors: u–p_1 and u–p_2. In the target, or *T-domain*, the *interactions count* feature will have its respective per-party value in each feature vector, i.e., 6 in u–p_1, and 14 in u–p_2. In the overall, or *O-domain*, the feature will be calculated as the sum over all

[7] http://en.wikipedia.org/wiki/List_of_emoticons.

parties, and will have the same value of 20 in both feature vectors. Lastly, in the relative, or *R-domain*, the feature will be calculated as the fraction of its values in *T*- and *O-domains*, i.e., $6/20 = 0.3$ in u–p_1, and $14/20 = 0.7$ in u–p_2. For some features, such as interactions weight and rank, we also use a variation of relative domain, the Δ *-domain*, in which a feature is calculated as the absolute difference of its values in *T*- and *O-domains*. Overall, counting all features, defined over all domains, our prediction models use 51 features.

Table 3 gives the top ten ranked features based on the information gain (as computed by Weka). Relative statistics turned out to be effective features, as the top seven ranked features are all based on the relative statistics. Six out of top ten features are interaction based (IB) and four remaining are Twitter-specific (TS) features, as indicated in the *Type* column of the table. In line with previous studies (Sparks 2010; Golbeck and Hansen 2011; Conover et al. 2011a), we observe a strong discriminative power of a preferential following and retweeting, as the corresponding features ranked 2nd and 4th in the *R*-domain, and 9th and 10th in the *T*-domain, respectively. When restricted to the top ten features only, the prediction models showed better performance on the training data. Therefore, for our experiments we will use models built on these ten features.

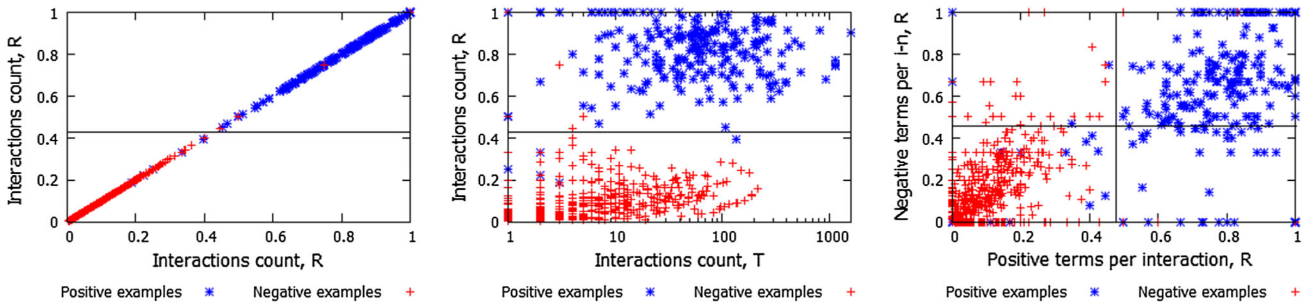
Table 3 Top ten features for predicting political preference

Feature	Domain	Type	Avg. rank
Interactions count	R	IB	1.3 ± 0.46
Followees count	R	TS	1.7 ± 0.46
Positive terms per interaction	R	IB	3 ± 0
Retweets count	R	TS	4.1 ± 0.3
Interactions frequency	R	IB	4.9 ± 0.3
Negative terms per interaction	R	IB	6.2 ± 0.4
Interactions weight	R	IB	7 ± 0.77
Followees count	T	TS	8 ± 0
Interactions weight	Δ	IB	9 ± 0.77
Retweets count	T	TS	9.8 ± 0.4

Figure 3 shows the distributions of training examples across different feature spaces. Each data point corresponds to a user–party feature vector. As it can be seen from Fig. 3a, the *interaction count in the R-domain* is a very strong feature which divides positive and negative examples very accurately. Figure 3b further illustrates this point, showing that while it is hard to separate positive examples from negative ones, judging only by the raw count of interactions, based on the relative count of interactions one can easily split the data. Finally, as it can be seen from Fig. 3c, the data points that represent positive and negative training examples are placed (mostly) at the upper-right and bottom-left quadrants, respectively, suggesting that, regardless of the term polarity, candidates tend to use more opinion words when they interact with their own parties, and less when they interact with their opponents.

5.2 Evaluation of the model

A major evaluation challenge was obtaining the test data. To have the ground truth preference of non-candidate users, we used the content generated during and up to 3 days after the election, i.e., precisely, everything between April 23, 9:00 a.m., MDT (ballot boxes are opened) and April 26, 11:59 p.m., MDT. We searched for the occurrences of words *vote* or *voted* followed or preceded by a *party marker* in a window of three words. Here, a party marker can be a party hashtag, a name of a party leader, a mention of a party account or any known account of a party candidate. This search resulted in a collection of 799 tweets by 681 users. We asked three annotators to classify each tweet in the collection as a statement that either supports certain party or not. Our criterion of support was the clear statement of the fact that vote has been casted or was about to be casted. Retweets of such statements were also counted as signs of support, as the evidence (Conover et al. 2011a) suggests that retweets generally express the



(a) Relative interactions count **(b)** Target vs. relative interactions count **(c)** Positive vs. negative terms rel. count

Fig. 3 Distribution of training examples across different features

Table 4 Characteristics of the test set

Supported party	Accounts	Interactions	Interactions per account
WRA	28	6,883	245.8
LIB	11	1,404	127.6
NDP	12	1,329	110.7
PC	24	2,510	104.6
Total	75	12,126	161.7

agreement with the original tweets. Cheering for parties, e.g., *vote NDP!*, was asked to be ignored. Annotators agreed that 99 out of 681 users had expressed prevailingly supporting statements. In the case of 64 users, the agreement was unanimous and for the remaining 35 users, two vote majority were achieved. The rate of inter-annotator agreement calculated as Fleiss' kappa (Fleiss et al. 1971) was 0.68. Recall that the vote statements were extracted from the content generated after the election. It is possible that users who expressed support for certain parties after the election did not interact with those parties during the election campaign. We exclude them from the test set, and focus on the remaining 75 users to whom we will refer to as the test users. Table 4 shows basic interaction statistics of the test users.

Human evaluation. To assess the difficulty of the task on human scale, we set an experiment in which we provided the same three annotators with randomly selected interactions of test users. For each test user and each party the user has interacted with, we chose up to 50 random interactions out of those that happened before the election. To create equal prediction conditions for humans and classifiers, each annotator was given four sets of interactions—one per each party. These sets were provided sequentially, one after another to avoid possibility of comparison. In other words, if a test user interacted with four parties, annotators would encounter postings of this user in four different sets of interactions. In such cases, the annotators, just like our classifiers, may predict that the same user will support multiple parties. We use the rate of annotator's agreement with the majority as the analog of classification confidence.

Baselines. Apart from human annotators, we compare our method with three more baselines. The first two baselines are text-based and do not take into account any Twitter-specific features such as following and retweeting. First, we use SentiStrength (Thelwall et al. 2010), a lexicon-based sentiment analysis tool optimized for informal writing style common to social media services. SentiStrength has been successfully used in estimating positive and negative sentiment in short texts such as twitter posts (Thelwall et al. 2011) and Yahoo answers (Kucuktunc et al. 2012). Being well suited for tweets and Twitter-specific lexicon, SentiStrength is expected to perform well (compared to our method) unless our behavioral features compensate for the lack of a content-specific orientation. Under default settings, for a given text, SentiStrength provides two scores as respective measures of the strength of positive and negative sentiment expressed in the text. These scores are varied in the range $[+1, +5]$ for positive sentiment and $[-1, -5]$ for negative sentiment. By analogy with our human evaluation experiment, we provide the tool with interactions of each user–party pair. For each interaction, the tool returns a sentiment strength score. We sum up these scores and treat the sum as the classification confidence. A resulting sum may be negative, in which case we conclude that SentiStrength predicted no preference.

Second, we employ a widely used Naive Bayes document classification method, treating all tweets of a user as a document belonging to a certain class (party), and computing prior and conditional probabilities based on the postings of candidates. The Naive Bayes document classification learns a language model for each party; it is expected to pick up all terms and concepts that are specific or important to each party, hence it is a good baseline. In other words, we chose a maximum a posteriori party as follows:

$$p_{\text{map}} = \arg\max_{p \in P} \left[\hat{P}(p) \prod_{t \in V} \hat{P}(t|p) \right]$$

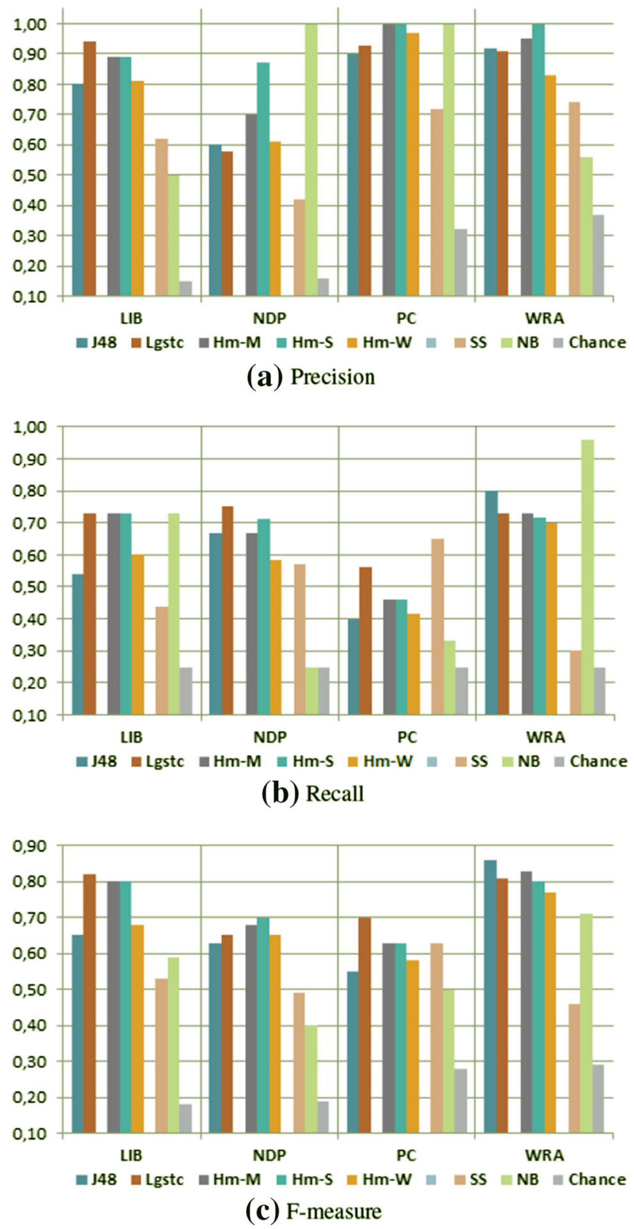
where P is the set of all considered parties; V is the vocabulary of all documents (all tweets of the candidates); $\hat{P}(p)$ is the apriori probability of a class, i.e., the share of documents (candidates) belonging to a class (party) p; $\hat{P}(t|p)$ is the probability of term t appearing in all postings of all candidates of party p, which is computed as follows:

$$\hat{P}(t|p) = \frac{Np(t) + 1}{\sum_{\tau \in V} Np(\tau) + |V|}$$

where $Np(t)$ is the count of term t in all postings of the candidates of party p, and the summation $\sum_{\tau \in V} Np(\tau)$ gives the total count of all terms in all postings of the candidates of party p. As it can be seen, we have opted for a simple add-one smoothing.

In a sense, all three baselines (human annotators, SentiStrength and document classification) operate at a content-level only, and as such, they can all be considered as content-driven approaches, with humans being the most advanced of all since they can account for humor, sarcasm, and other aspects of natural language currently almost untraceable by the machines.

Finally, as a sanity check, we compare our method to chance; it has been argued that when reporting predictions on Twitter, chance is a viable baseline (Metaxas et al. 2011). This baseline predicts that user will prefer one of the



(a) Precision

(b) Recall

(c) F-measure

Fig. 4 Preference prediction of our classifiers (J48 and Lgstc), *Hm-M* majority vote of annotators, *Hm-S* strongest annotator, *Hm-W* weakest annotator, *SS* SentiStrength, *NB* Naive Bayes, and Chance for Albertan election

parties with an equal probability. For each user, we randomly generate 1,000 predictions and choose the party that was predicted most of the times. Ties are broken randomly.

5.3 Results

We experimented with both a decision tree-based J48 and Logistic regression classifiers. We train one classifier per party, and present the results on a per party basis. As for human annotators, with respect to each evaluation metric, we report results for the strongest and the weakest performing annotators, as well as for the "majority vote". Figure 4 shows the results of the experiment. Each of three plots corresponds to an evaluation metric, and consists of four clusters of bars associated with supporters of four major parties. Each such cluster consists of seven bars, corresponding to the performances of two classifiers, three human annotators, and three baselines. The order of the bars, from left to right, corresponds to that of the legend items.

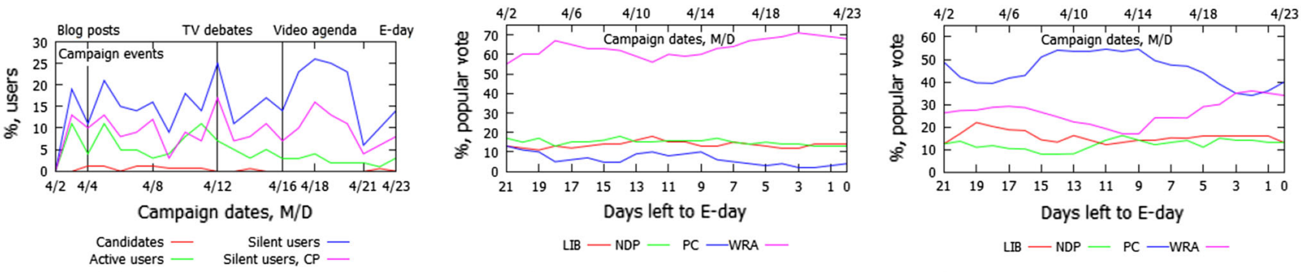
As it can be seen from Figure 4a, both classifiers make less precise predictions than the annotators, although Lgstc shows better precision than J48 especially for LIB and PC parties. Moreover, this classifier outperforms the least accurate annotator for LIB and WRA parties. As for baselines, NB makes 100 % precise predictions for PC and NDP parties, surprisingly outperforming humans in the latter task. A possible explanation for this could be the fact that test users who support PC and NDP frequently retweet party candidates, literary copying the content, therefore making NB assign higher probabilities to their posts.

In terms of recall, classifiers again perform close to human annotators, cf. Figure 4b. It is interesting that for the PC party, both Lgstc and SentiStrength outperform human annotators, and for the WRA party, J48 and NB do the same. One of the possible explanations is that a simple quantification of opinion words and estimation of the content similarity employed by the baselines turned out to be more effective (in this particular instance) than a human logic that puts opinion words in the context (sometimes rendering them as neutral) and has no means to compare the content (again, in this particular instance). Similarly, it could be that for the annotators, the interactions of some of the users with the PC party did not seem to have meaning, but our learned models could have exploited different features, such as the following and the retweeting preference, interaction frequency, etc. Thus, as we can see, in combination with the content-based features, the behavioral features, such as the aforementioned ones, can be very effective.

Finally, as shown on Fig. 4c, in terms of *F*-measure, the classifiers outperform all of the baselines, except for the PC party where Lgstc shows slightly lower performance than SentiStrength. Lgstc shows great performance outperforming all of the annotators for the LIB and PC parties and the weakest annotator for the WRA party. In turn, J48 outperforms all of the annotators for the WRA party.

6 Temporal analysis

Having a prediction model, we wanted to explore how the predicted political preference of users changes with the progression of the election campaign, and if some group of



(a) Political preference **(b)** Popular vote of active users **(c)** Popular vote of silent users, CP

Fig. 5 Changes of political preference and popular vote over time

users are more likely to change their preference than others. To study such changes, we set up an experiment as follows: we choose a window of a fixed number of days, say N, and by sliding this window (one day at a time) over the timespan of the campaign (28 days), we obtain a sequence of time periods, each N days long. We arrange the interactions of each user into these time periods according to their registered tweet time. Suppose user u has interactions in two consecutive periods p_1 and p_2. Let predicted political preference of u for periods p_2 and p_1 be P_2 and P_1, respectively. If $P_2 \neq P_1$, we say that on the given day d, the predicted preference of user u for the current period p_2 has changed compared to the that for the previous period p_1. To capture changes in preference for the whole duration of the campaign, we need to repeat this procedure for all consecutive periods. This, however, requires a user to have interactions in all of these periods. Hence, for this experiment we select only those users who satisfy this condition. Experimentally, we chose the window of size seven, i.e., we measure changes in the preference of users on a weekly basis. This choice allowed us to identify 3,413 users who satisfied the condition of the experiment. Of those, 129 were active and 791 were silent users[8]. We consider a user to be active if she contributed at least ten interactions per day over the course of the campaign. Correspondingly, a user who has engaged in at most 1.5 interactions per day on average is considered a silent user. For the experiment, we randomly selected 100 users from each of these user groups.

Discussion. Figure 5a shows the percentage of users for whom on a given date, the political preference for the current period has changed compared to that of the previous period. As one would expect, the preference of candidates did not change during the campaign, apart from negligible deviations of about 1 %. On contrary, the preference of the silent users is changing constantly and for certain points in the campaign rather drastically. A further examination revealed that due to a small number of interactions in some periods, our method predicted no preference for significant number of users. Needless to say that for the active users "no preference" was almost never predicted. To account for the interaction sparsity, we repeated the experiment for the silent users under a *constant preference* (CP) setting, where we assume that if for the current period a user was predicted to have no preference, then user's preference did not change since the last period. In other words, a no preference prediction for the current period is replaced with the prediction made for the previous period.

As seen from Fig. 5a, although the CP setting reduced the share of the silent users with changing preference, it is still higher than that of the active users' almost for the whole duration of the campaign. Correspondingly, the share of the active users with changing preference never exceeds 11 % and since April 13 (10 days before the E-day) never exceeds 5 %. Figure 5b and c shows the distributions of the popular vote of the active and the silent users for each period on a given date. It can be seen that for both active and silent users, the preference generally changes between WRA and PC parties. In agreement with our findings, popular vote of the active users changes gradually, and sharp transitions mostly occur before April 12. For the silent users, however, shifts in popular vote occur rather sharply for the whole duration of the campaign. From this, we conclude that the active users are less likely to change their political preference compared to the silent users.

There is an interesting point in Fig. 5 about the silent users. The popular vote for PC remains higher than WRA throughout the campaign except for the last few days. In those last few days, the popular vote shows a decreasing trend for PC and an increasing trend for WRA until the two are neck-to-neck 3 days prior to the election day. A CBC poll conducted 3 days before the election day also put the two parties neck-to-neck. However, we see a not-so-strong changing trend starting to show 1 day before the election with popular vote for PC rising and the popular vote for liberals declining, as a possible sign of strategic voting changing the course of the election. Another point about Figure 5b and c is that the popular vote prediction for silent

[8] The rest were moderate users which are ignored here.

users shows more resemblance to the actual election outcome than active users, with the ordering of the parties: PC, Wildrose, Liberals and NDP from the largest to the least number of received votes.

We wanted to see if major changes in predicted political preference occur spontaneously or on certain dates that have some significance to the campaign. From the social media highlights of the campaign[9] we found descriptions of the following events, which were extensively discussed in blogsphere, on Facebook and Twitter: (i) *Blogposts* by Kathleen Smith (April 2) and Dave Cournoyer (April 4) criticizing WRA party. (ii) Television broadcast of party leaders' *debates* (April 12). (iii) *YouTube video* titled *"I never thought I'd vote PC"* (April 16), asking people to vote strategically against WRA party. The vertical lines on Fig. 5a represent these events together with their occurring dates. As it can be seen, the highest change in the preference of silent users occurred on April 12, the day of the TV broadcast of the party leaders' debate. It is interesting that for the active users the peak change in preference occurred 1 day before the debates. This could be the case where the discussion of the event was more interesting than the event itself. In the case with blogposts and the video, the rise in the preference change rate occurs only on the next day after events took place. Twitter discussion of these events might have had the "long term" effect gaining more popularity on the next day and influencing the predictions for the next period.

7 Evaluating the method across the elections

In this section, we address the issues of a possible election-dependent bias and the applicability of the method to any given election. To this end, we evaluate our method in the context of another election, specifically 2013 Pakistani general election. We start by providing a brief background on the election. Then, in an orderly fashion, we summarize the data collection process and the experimental setup, pointing out some minor differences in the methodology. Finally, we thoroughly discuss the results and some interesting findings.

7.1 Election background

General elections were held in Pakistan on May 11, 2013 to elect the members of the 14th National Assembly and the four provincial assemblies. The elections took place in 272 directly elected constituencies. Voter turnout was 55.02 %, the highest since 1970 and 1977. None of the dozens of the registered parties achieved the 137 seats overall majority. The Pakistan Muslim League—Nawaz (PML-N) won 126 seats—almost four times as much as the closest contestant Pakistan Peoples Party (PPP). Two other major parties considered in the present work, namely Pakistan Tehreek-e-Insaf (PTI) and Muttahida Qaumi Movement (MQM), won 28 and 19 seats, respectively[10].

Table 5 Characteristics of the training set

User group	Accounts	Interactions	Interactions per account
Candidates	86	105,848	1,230.8
PML-N	11	24,007	2,182.5
PPP	13	28,235	2,171.9
PTI	43	32,761	761.9
MQM	19	20,845	1,097.1

Table 6 Basic characteristics of the interaction profiles of the parties

Party	Profile size	Top terms
MQM	209	*#mqm, altaf hussain*
PML-N	199	*#roshanpakistan, nawaz sharif*
PPP	202	*#ppp, president zardari*
PTI	223	*#pti, imran khan*

7.2 Data set description

Candidate accounts were collected semi-automatically by parsing the official Election Commission of Pakistan sources[11]. Just as we described in Sect. 4, for the collected accounts, we retrieved up to 3,000 tweets posted during the campaign period. More specifically, we collected every tweet posted between April 1st (i.e., a few days before the official campaign started) and May 11th, 2013 (the day of the election). In total, we collected 86 accounts and extracted 105,848 interactions, as depicted in Table 5. Then, following the methodology described in Sect. 3, we built interaction profiles of the parties; Table 6 gives a summary of the profiles.

To collect a test set, we used a different approach, due to the fact that we started collecting the data almost 4 months after the election took place, which resulted in difficulties in obtaining keyword search results for such a long-gone period in a Twitter scale. On the other hand, for our target experiment (i.e., cross-election evaluation), we did not need to obtain thousands of accounts of which only a small fraction probably voted, as in the case with Albertan

[9] http://blog.mastermaq.ca/2012/04/28/alberta-election-social-media-highlights.

[10] Pakistani general election, 2013: http://en.wikipedia.org/wiki/Pakistani_general_election,_2013.

[11] http://www.ecp.gov.pk.

Table 7 Characteristics of the test set

Supported party	Accounts	Interactions	Interactions per account
PML-N	14	10,936	781.1
PPP	10	12,325	1,232.5
PTI	39	19,859	509.2
Total	63	43,120	684.4

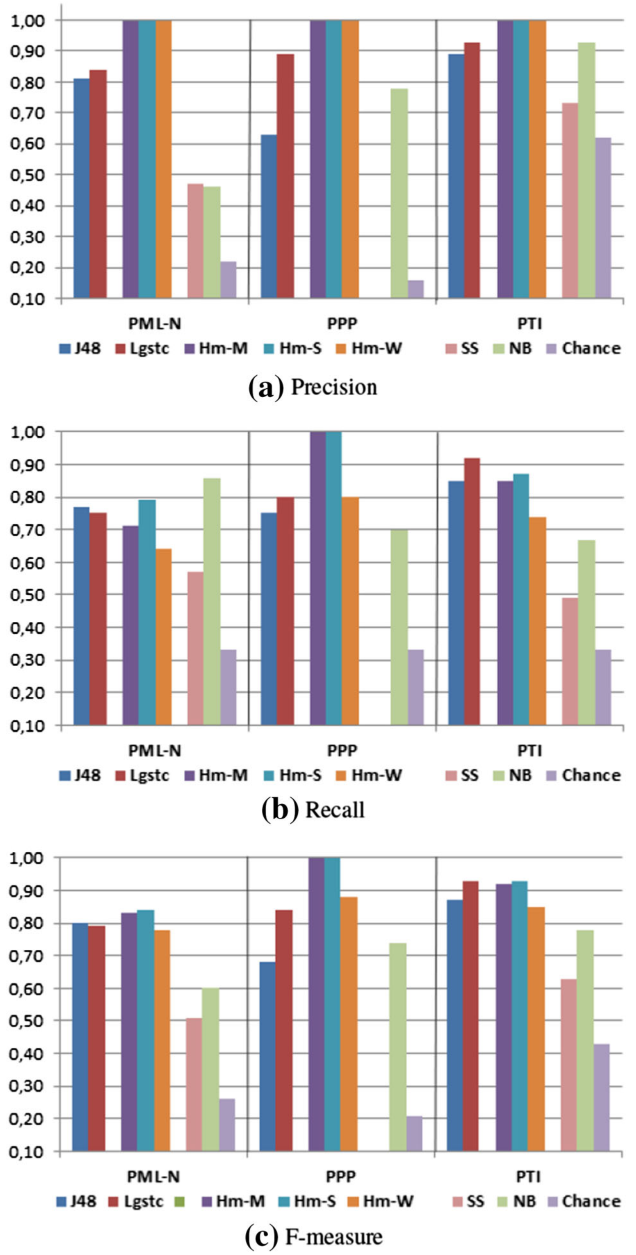
election (cf. Sect. 5), which would have required an extensive time and effort cleaning the collected data. Instead, we used Twitter search directly and obtained accounts whose owners explicitly stated (through a posting), on the election day or shortly after, who they had voted for. As in the case with the Albertan election, we asked three annotators to evaluate such tweets and identify the ground truth partisan identifications of accounts with seemingly genuine statements of support. The annotators achieved high agreement, or more accurately 0.83 in terms of Fleiss' kappa (Fleiss et al. 1971).

As it can be seen in Table 7, for our test set, we collected 63 accounts and extracted 43,120 interactions (3.5 times as much as for the test set for Albertan election, cf. Table 4). Notice that we have no MQM supporters in the test set. In fact, there were five accounts whose owners expressed their support for MQM, but, unfortunately, these accounts did not produce interactions during the election campaign, and we could not include them.

7.3 Experiments and results

We conducted a set of experiments to estimate the general accuracy of the method across the elections and to test the performance of the method on the Pakistani election on a per-party basis. The experimental setup and all the baselines were identical to the Albertan election (cf. Sect. 5). As for prediction models, we used the same top ten ranked features (cf. Table 3), making an adjustment for language-dependent features. Specifically, we enriched sentiment lexicons with Urdu[12] and Roman-Urdu lexicons. The latter was obtained by automated transliteration[13] of the Urdu lexicon.

Let us begin with the analysis of the performance of the method on the data covering the Pakistani election. Figure 6a–c shows a comparison of our method to the other approaches when evaluated on a per-party basis in terms of precision, recall, and *F*-measure, respectively. It can be seen that the annotators were perfectly precise, achieving a 100 % precision for all parties. The Lgstc classifier achieved higher precision than a decision tree-based J48 for all parties. In general, both classifiers outperformed sentiment classification approach (SS) and the chance baselines, and in the case of the PTI party, Lgstc and document classification baseline (NB) performed equally, achieving 93 % precision. Interestingly, a SS could not correctly identify a single voter of the PPP party, i.e., showed 0 % precision. Consequently, for this party, SS showed 0 % recall, and the *F*-measure could not be computed.



(a) Precision

(b) Recall

(c) F-measure

Fig. 6 Preference prediction of our classifiers (J48 and Lgstc), *Hm-M* majority vote of annotators, *Hm-S* strongest annotator, *Hm-W* weakest annotator, *SS* SentiStrength, *NB* Naive Bayes, and Chance for Pakistani election

In terms of recall (cf. Fig. 6), our classifiers show better performance by having higher recall than both the human

[12] http://chaoticity.com/urdusentimentlexicon.

[13] http://www.ijunoon.com/transliteration/urdu-to-roman.

Table 8 General accuracy of the method on a per-election basis

Method	Albertan election	Pakistani election
J48	0.61	0.78
Lgstc	**0.68**	0.86
Human majority	0.63	0.84
Human strongest	0.63	**0.87**
Human weakest	0.58	0.73
Sentiment classification	0.48	0.43
Document classification	0.61	0.72
Chance	0.25	0.33

majority and the weakest annotator for the PML-N party. Also, Lgstc shows similar performance as that of the weakest annotator for PPP and outperforms all the annotators for PTI. With the exception of the PML-N party for which NB achieves a 86 % recall, our classifiers consistently outperform all the baselines for every party. Of our classifiers, Lgstc achieved the highest recall of 92 % for the PTI party.

Finally, in terms of *F*-measure, our classifiers perform better than all the baselines, and for the PML-N party both classifiers outperform the weakest annotator. Also, for PTI Lgstc outperforms all of the human baselines achieving 93 % *F*-measure.

To compare the results across elections and to evaluate their consistency, we plotted in Table 8 the performance of the methods for both elections side-by-side, in terms of the general accuracy, i.e., the percentage of all test users for whom the correct preference was predicted. Comparing the absolute performance numbers, we can see that all methods except sentiment classification perform better in the Pakistani election. This has to do with the change in population; voters in the Pakistani election often seemed decided and were more explicit in their support of a party or a leader, whereas this was less obvious in the Albertan election. This argument is supported by the number of undecided voters. According to some estimates[14], 18 % of Albertan voters were still undecided even 5 days before the election day, a figure which was surprising to many. Only 11 % of Pakistani voters reported having no preference towards any party[15]. Also, there was more discussion of topics from party platforms in the Albertan election (compared to the Pakistani election), and this could have added another twist. Next, we draw conclusions based on the relative performance of our method with respect to other approaches.

It can be seen that for both elections, our classifiers perform well above the chance and the sentiment classification approach. Also, for both elections, the Lgstc outperforms the document classification baseline, and J48 shows equal performance to this baseline in case of the Albertan election (AE) and gains 6 % improvement over it in case of the Pakistani election (PE). For both elections, logistic regression classifier performs better than J48. Moreover, in both cases, the advantage is almost identical; for AE, Lgstc gains 7 % over J48, and for the PE, the gain is 8 %. We conclude that the method shows consistent performance with respect to machine-based baselines. We notice less striking consistency when comparing the performances in relation to the human baselines. J48 gains 3 % accuracy over the weakest human baseline for the AE, and for the PE, the advantage is 5 %. On the other hand, Lgstc outperforms the strongest annotator by 5 % in case of AE, and in case of PE, the classifier falls short by 1 %. Taking into account the fact that the inter-annotator agreement was lower for AE (0.68) than that for PE (0.83), the performance of the method with respect to the human baselines can be considered consistent. Thus, overall we conclude that our method displays consistent performance when applied to different elections.

8 Conclusions

We studied the problem of predicting political preference of users on the Twitter network. We showed that the generated content and the behavior of users during the campaign contain useful knowledge that can be used for predicting the political preference of users. In addition, we showed that the predicted preference changes over time and that these changes co-occur with campaign-related events. We also compared the preference change of silent users to that of vocal users, and found that vocal users are more reluctant to change their preferences during the pre-election campaign. To address the concern of a possible dependence of the approach to a specific election, we re-evaluated the method on a different election, and showed that the performance on the new data remained consistent with the originally obtained results. Also, almost 1-year-long temporal gap between the elections suggests that our method is robust with respect to the ever-changing behavior of Twitter users.

There is a number of possibilities for future research. First, the process of a party profile generation can be further automated using various seed set extension techniques (Conover et al. 2011a, b).

Second, when faced with the issue of language dependency of some of the features, we had to make necessary adjustments. To make the method robust with respect to

[14] http://www.edmontonjournal.com/news/6477884/story.html.

[15] http://lgdata.s3-website-us-east-1.amazonaws.com/docs/35/711395/survey.

this issue, we plan to refrain from using such features or replacing them. We have already ran initial experiments and tried to apply the collaborative filtering techniques.

Third, investigation of patterns of strategic voting still remains an open and interesting research question. For the Albertan election campaign, strategic voting was a widely discussed issue, and we found numerous evidence of it on Twitter, while developing our method, e.g., *just voted PC lesser of two evils but i still feel like i need a shower. #abvote #yyc #yeg*. Also, at some point in the campaign special trends started to emerge. These trends used hashtags like #nowrp, #strategicvotes, #strategicvoting, etc. We believe studying the behavior of users engaged in these discussions can help improve our models, leading to more accurate preference predictions.

Finally, with elections happening virtually everywhere and everytime, it would be very interesting to address the automated election identification problem by analyzing local and global Twitter trends. An approach capable of detecting election-related tweets can be combined with the method described in this paper to attempt a completely automated approach to the prediction of political preferences of Twitter users.

References

Benevenuto F, Magno G, Rodrigues T, Almeida V (2010) Detecting spammers on twitter. In: Proceedings of Collaboration, Electronic messaging, Anti-Abuse and Spam Conference (CEAS)

Choy M, Cheong MLF, Laik MN, Shung KP (2011) A sentiment analysis of Singapore presidential election 2011 using twitter data with census correction. CoRR abs/1108.5520

Conover M, Gonçalves B, Ratkiewicz J, Flammini A, Menczer F (2011a) Predicting the political alignment of Twitter users. In: Proceedings of SocialCom/PASSAT Conference, pp 192–199

Conover M, Ratkiewicz J, Francisco M, Gonçalves B, Menczer F, Flammini A (2011b) Political polarization on Twitter. In: Proceedings of the ICWSM Conference

Fleiss J et al (1971) Measuring nominal scale agreement among many raters. Psychol Bull 76(5):378–382

Golbeck J, Hansen DL (2011) Computing political preference among twitter followers. In: Proceedings of HCI Conference, pp 1105–1108

Hu M, Liu B (2004) Mining and summarizing customer reviews. In: Proceedings of KDD Conference, pp 168–177

Kucuktunc O, Cambazoglu B, Weber I, Ferhatosmanoglu H (2012) A large-scale sentiment analysis for Yahoo! answers. In: Proceedings of WSDM Conference, pp 633–642

Livne A, Simmons MP, Adar E, Adamic LA (2011) The party is over here: structure and content in the 2010 election. In: Proceedings of the ICWSM Conference

Makazhanov A, Rafiei D (2013) Predicting political preference of Twitter users. In: Proceedings of the ASONAM Conference, pp 298–305

Marchetti-Bowick M, Chambers N (2012) Learning for microblogs with distant supervision: political forecasting with Twitter. In: Proceedings of the EACL Conference, pp 603–612

Metaxas P, Mustafaraj E, Gayo-Avello D (2011) How (not) to predict elections. In: Proceedings of the SocialCom/PASSAT Conference, pp 165–171

Mustafaraj E, Finn S, Whitlock C, Metaxas PT (2011) Vocal minority versus silent majority: discovering the opionions of the long tail. In: SocialCom/PASSAT, pp 103–110

O'Connor B, Balasubramanyan R, Routledge BR, Smith NA (2010) From tweets to polls: linking text sentiment to public opinion time series. In: Proceedings of the ICWSM Conference

Ratkiewicz J, Conover M, Meiss M, Gonçalves B, Flammini A, Menczer F (2011) Detecting and tracking political abuse in social media. In: Proceedings of the ICWSM Conference

Sepehri HR, Makazhanov A, Rafiei D, Barbosa D (2012) Leveraging editor collaboration patterns in Wikipedia. In: Proceedings of the HT Conference, pp 13–22

Sparks D (2010) Birds of a feather tweet together: Partisan structure in online social networks. In: Presented at the 2010 meeting of the Midwest Political Science Association

Thelwall M, Buckley K, Paltoglou G, Cai D, Kappas A (2010) Sentiment strength detection in short informal text. J Am Soc Inf Sci Technol 61(12):2544–2558

Thelwall M, Buckley K, Paltoglou G (2011) Sentiment in Twitter events. J Am Soc Inf Sci Technol 62(2):406–418

Tumasjan A, Sprenger TO, Sandner PG, Welpe IM (2010) Predicting elections with Twitter: What 140 characters reveal about political sentiment. In: Proceedings of the ICWSM Conference

Wang H, Can D, Kazemzadeh A, Bar F, Narayanan S (2012) A system for real-time Twitter sentiment analysis of 2012 US Presidential election cycle. In: ACL (System Demonstrations), pp 115–120