Predicting and Interpolating State-level Polls using Twitter Textual Data

Nicholas Beauchamp
Northeastern University

Abstract

Spatially or temporally dense polling remains both difficult and expensive using existing survey methods. In response, there have been increasing efforts to approximate various survey measures using social media, but most of these approaches remain methodologically flawed. To remedy these flaws, this paper combines 1200 state-level polls during the 2012 presidential campaign with over 100 million state-located political Tweets; models the polls as a function of the Twitter text using a new linear regularization feature-selection method; and shows via out-of-sample testing that when properly modeled, the Twitter-based measures track and to some degree predict opinion polls, and can be extended to unpolled states and potentially sub-state regions and sub-day timescales. An examination of the most predictive textual features reveals the topics and events associated with opinion shifts, sheds light on more general theories of partisan difference in attention and information processing, and may be of use for real-time campaign strategy.

Replication Materials: The data, code, and any additional materials required to replicate all analyses in this article are available on the American Journal of Political Science Dataverse within the Harvard Dataverse Network, at: http://dx.doi.org/10.7910/DVN/RJAUNW

The author would like to thank for their invaluable comments and suggestions the participants at the MPSA, APSA, and New Directions in Text conference panels and the Harvard Applied Statistics and the NYU Social Media and Political Participation workshops where drafts of this paper were presented.

1 Introduction

State-level public opinion is at the heart of the US political process, determining not just gubernatorial and senatorial elections, but presidential elections via the electoral college. Yet despite this importance, time-dense state-level polling is rare, and even during presidential elections, is limited to a small handful of swing states. More generally, there is a strong ongoing need for survey data of all sorts that is regionally and temporally dense,[1] a demand that is rarely met given the expense. Given its current abundance, geographically and temporally located social media data would seem to be ideal, were it not for the manifest unrepresentativeness of those users. Many efforts have been made to show that nevertheless, social media data can track representative measures of public opinion, but as will be discussed below, most of these have significant flaws.

This paper attempts to remedy many of these flaws and show that the text of a sufficiently large collection of politically topical Twitter posts, identified down to the state-day level, can provide a method for (a) extrapolating vote intention in states that are poorly polled; (b) interpolating vote intention for unpolled days, and potentially for smaller time periods and sub-state regions; and (c) improving upon polling, even in well-polled states, for measuring quick changes in vote intention. This general approach can be extended to any other time series or time-series cross-sectional (TSCS) data – consumer sentiment, product sales, unemployment, etc – and offers significant improvements over previous approaches to estimating real-world survey data using social media data.

In addition to these practical applications, this approach allows us to extract from the social media data stream the textual features that are best predictive of the polling data, providing real-time substantive insight not just into what people are saying, but into the subset of what they say that correlates with important political behavior such

[1] Illustrated, for instance, by the rising popularity of multilevel regression and poststratification methods (Park, Gelman & Bafumi 2004, Lax & Phillips 2009, Ghitza & Gelman 2013).

as vote intention. This provides insights into the behavior and psychology of both social media users and the public more generally: the results here are consistent with existing theories of partisan differences in information use (Huckfeldt 1995, Kenski & Stroud 2006, Pennacchiotti & Popescu 2011, Wong et al. 2013), where left-leaning regions show more citation of external sources (URLs) and regional issues, while right-leaning regions show a higher degree of retweeting and national issues. These domain-specific results suggest that these methods may be useful not just for measuring opinion, but potentially for shifting vote-intention on the short time-scales necessary for modern campaigns.

The paper proceeds in seven stages: after this introduction, section 2 presents a brief examination of existing efforts to measure real-world trends using social media, both within politics and beyond. The main issue is that existing works have generally set the bar for success far too low, and that critique serves to define the alternative approach taken here. Section 3 describes the data preparation – how the Twitter and polling data are processed and combined. Section 4 describes the modeling approach, which allows us to model and predict vote intention as a function of Twitter textual features. In section 5 the model is tested, where out-of-sample validation shows that it does successfully allow one to track even very short-term polling changes using Twitter textual data, and shows that it outperforms a benchmark suite of standard machine-learning methods, and is the only approach tested here to out-perform the polls themselves in tracking opinion change. Finally, section 6 finishes with a brief descriptive discussion of the textual features that track inter-state, intra-state, and short-term variations in vote intention, which suggests that in this election at least, a more internally consistent and nationally-oriented Republican Twitter community may have been driving much of the cross-sectional results, while Republican concerns regarding the debates and Benghazi may have been driving much of the short-term temporal results.

2 Existing work in social media measurement

At this point there exists something of a minor industry dedicated to measuring public opinion using social media. And indeed, within it now exists an only slightly smaller industry dedicated to the critique of those purported measures (Lui, Metaxas & Mustafaraj 2011, Metaxas, Mustafaraj & Gayo-Avello 2011, Gayo-Avello, Metaxas & Mustafaraj 2011, Chung & Mustafaraj 2011, Jungherr, Jürgens & Schoen 2012, Gayo-Avello 2013). Gayo-Avello (2013) in particular serves as a useful meta-analysis of the existing efforts and their drawbacks, but although it provides a variety of criticisms, it is worth analyzing here a few of the more prominent attempts with an eye towards the general lessons we can draw about how it might better be done.

As we will see, there are four essential lessons to be learned about how to do social media prediction[2] scrupulously. In the interests of space, in the following discussion which of the lessons below are at issue is noted in brackets, using the following abbreviations:

1. Statistical testing [S]: Success must be measured statistically, not merely with descriptives or Mean Average Error; this requires an N large enough to support that, which also reduces issues of selection and desk-drawer bias.

2. Benchmarks [B]: Success must be measured relative to clear benchmarks, which can be previous election results, existing polls, or in the case of win prediction, default assumptions such as incumbency success.

3. Training [T]: Given the unrepresentativeness of Twitter users, purely a priori measures like candidate mentions or sentiment are unlikely to succeed without some sort of machine learning or model training, which in turn necessitates an abundant training set for the dependent variable (polls, earlier election results, etc).

4. Out of sample [O]: Given that models are being fit, validation must be carefully out of sample, ideally forward in time, with a careful specification of the in-sample model specifications, parameter fitting, or ensemble selection.

[2] "Prediction" is used in the out-of-sample sense here. That is, we are interested in predicting or measuring vote intention in states or days that are not polled, using text data that is collected during those times or in those locations. Apart from the single election that is roughly predicted from the data a few days before, there is no real prediction of truly future events, although today's text can give a peek at what will only be reported in tomorrow's polls.

Although early efforts to measure party success or electoral outcomes using weblogs were mainly unsuccessful (Jansen & Koop 2005, Albrecht, Lübcke & Hartig-Perschke 2007), the explosion of publicly available Twitter data changed the game abruptly in the last few years. The first prominent apparent successes in the political domain using Twitter are Tumasjan et al. (2010), which claimed to predict vote shares for German parties using Twitter party mentions, and O'Connor et al. (2010), which claimed to be able to match time series jobs sentiment measures using Twitter sentiment analysis. Tumasjan et al. (2010) in particular was immediately and thoroughly critiqued (Metaxas, Mustafaraj & Gayo-Avello 2011, Gayo-Avello, Metaxas & Mustafaraj 2011, Chung & Mustafaraj 2011, Jungherr, Jürgens & Schoen 2012), where Metaxas, Mustafaraj & Gayo-Avello (2011) argue that simple party mentions are highly subject to the vagaries of non-representative Twitter users (eg, had it been included in their analysis, the Pirate Party would have been predicted the overall winner of the German election) and the exact time frame chosen prior to the election [T, O]. These authors' efforts to replicate the results in Tumasjan et al. (2010), either with the same German data or in 6 US senate elections, do no better than chance [S, B], even for predicting the raw winners, let alone the vote percentages. Lui, Metaxas & Mustafaraj (2011) argue more generally against such crude count-based measures (eg, Google Trends), which are severely biased by the non-representative users [T, O]; such methods continue to be used (Skoric et al. 2012, Gaurav et al. 2013), but are generally plagued by small N, ad-hoc parameter settings, and presumably high selection bias (Lazer et al. 2014) [S, B, T, O].

The sentiment-based methods such as O'Connor et al. (2010) have not fared much better with time. Gayo-Avello, Metaxas & Mustafaraj (2011) attempt to replicate O'Connor et al. (2010) on the 2008 election, without success – and unsurprisingly, since O'Connor et al. (2010) grant that it doesn't actually work in their electoral test, just the jobs mea-

sure. But it doesn't really work on the jobs sentiment measure either, for two reasons: first, to match their "predicted" time series to the truth, they try a large number of different lags, and report success when they find that a subset of these tested lags produce high correlations between the two series; this is not truly an out-of-sample test, and is a particularly problematic when the two series happen to share a secular trend or are both concave or convex [O]. And second, as the authors mention in passing, the sentiment works when they use Tweets that contain the word "jobs", but not Tweets that contain the word "job;" they argue that this illustrates the importance of not stemming (which is true), but it also illustrates the danger of fishing and post-hoc model selection that is not truly out-of-sample [T, O]. This is especially problematic for sentiment methods, which remain ad hoc, language specific, and dependent on often-atheoretic word lists [T].

Perhaps in response to these manifest limitations, a more recent set of efforts has turned to supervised machine learning methods, which improve prediction by training algorithms on existing polls in order to better select and weight the features used to predict further polling. Bermingham & Smeaton (2011) employ a relatively small set of sentiment and frequency measures and train them via regression on polls prior to the election, which produces a decent match with party vote shares – but with an untestable N of 5 [S], and again, with the danger of fishing through parameter space for an ensemble of weights that works best [O]. Sang & Bos (2012) similarly reweight sentiment measures using polls, but provide no statistical test for the success of their predictions (N = 11) [S]. Ceron et al. (2014) do less training, but compare sentiment measures against polls on a rolling basis, yielding a half-dozen temporal measures per candidate; this is at least enough for a slight statistical test, and they appear to find that three of the seven candidates they examine have statistically significant matches between the Twitter series and the polls; whether that is more than we would expect by chance remains unanswered [S].

Two of the more scrupulous recent efforts are Livne et al. (2011) and Huberty (2013). Both use congressional elections to generate a larger N and are clear about their comparative benchmark (predicting electoral success based only on incumbency and party membership). Livne et al. (2011) find that a collection of features, including link-centrality and party-speech-centrality, appear to improve on the party + incumbency benchmark, but these features appear not be selected strictly out-of-sample [O], and in head-to-head competitions their accuracy is lower than simply picking the incumbent to win [B]. Huberty (2013) trains a SuperLearner ensemble on the 2010 election results, and then tests out-of-sample in two ways: against a held-back sample of 2010, and against 2012 results. This achieves only partial success: the SuperLearner improves on the incumbency benchmark for 2010, but not forward to 2012 [B]. The drawback of the first model is that it was not explicitly designed for election prediction (about which the authors are clear); the drawback of the second is that, while such ensembles can be very powerful in maximizing out-of-sample prediction, they are less successful when the test out-sample is unlike the training out-sample (eg, 2012 vs 2010) [O].

To sum up, in light of the four points raised above, a good test of a social media "prediction" must have a large enough out-sample for rigorous statistical testing; must be relative to reasonable benchmarks such as existing polling or incumbent success rates; will likely necessitate model fitting in-sample, and thus will require large quantities of the dependent variable in-sample; and should ideally be tested forward in time with all training done in-sample. The approach taken here meets all these criteria: we have a measure of the dependent variable (poll-measured state-level vote intention) and the independent variables (aggregate state-level Twitter word frequencies for 10,000 words) over 24 states and 2 months. This provides enough data to rigorously train and test the model out-of-sample. In addition, the text-based predictions are compared not just to the null (no

predictive ability whatsoever), but to rigorous benchmarks: first, to the prediction of poll-based opinion based on extrapolation from past polls; and second, to a series of standard machine-learning methods.

A unique advantage of this approach is that we need know nothing about the nature of Twitter users or political sentiment: the text features that correlate with truly representative public opinion (as measured by the polls) will be extracted and utilized for later text-based poll prediction, including extrapolation to unpolled states. The drawbacks are that we need plentiful and continuous training data, and we can only learn via post-hoc analysis of the extracted features exactly which signals in the Twitter stream are best matching and predicting proper polls – and even then, those interpretations must remain somewhat speculative.

3 Data preparation

Though individually fairly crude, tweets are produced at a sufficient rate[3] that they constitute an immensely rich data source in aggregate. Using Twitter's streaming API, every tweet containing any of a small set of political words[4] were collected beginning in June 2012 through June 2013. Twitter limits its basic feed to at most 1% of all tweets at a given time, but only for a few hours during the presidential debates was this ceiling hit, so for the most part the dataset constitutes every tweet containing these political words. The complete dataset amounts to about 200 million political tweets, but the for the present purposes, is limited to about 120 million political tweets between September 1, 2012 and

[3]Approximately 100 million on any given day during the collection period.

[4]Specifically: *obama, romney, pelosi, reid, biden, mcconnell, cantor, boehner, liberal, liberals, conservative, conservatives, republican, republicans, democrat, democrats, democratic, politics, political, president, election, voter, voters, poll, polls, mayor, governor, congress, congressional, representatives, senate, senator, rep., sen., (D), (R).* Note that in some cases these will generate false positives (ie, capture non-political tweets), but inspection suggests that the vast majority of collected tweets are in fact political; more importantly, this is not actually an issue for the supervised methods used here, which only utilize individual words that are actually correlated with variations in the polls.

election day.

Since the goal is measuring state-level opinion, the most challenging issue is identifying locations associated with each tweet. Although Twitter provides an automatic geocoding function, it is opt-in and very few users use it (1-3% at the time of these data). The "location" field on the other hand is free text, and thus consists of a lot of junk ("in a world of my own;" "la-la land;" etc) mixed in with actually informative text. The total dataset is far too numerous to use public location APIs, so instead a parser was constructed out a few lists of state names, abbreviations, and major cities, which appears to locate about 1/3 of all the tweets to a US state; manual validation of a small subset showed that few of these appear to be false positives. The located data thus amounts to about 40 million tweets – over a thousand for most state-day units, even for the low-population states.

To extract the textual features of tweets, the top 10,000 unigrams (including hashtags, urls, etc) were retained,[5] and for each state-day unit, the percentage of that unigram in that unit was calculated (eg, 0.02 for "obama" would mean that 2% of all words used in that day in that state were "obama," at least from among the top 10,000). Thus the 850 GB of raw JSON Twitter data is reduced to a mere 500 MB (compressed) dataset of 50 states x 67 days x 10,000 variables.

Turning now to the dependent variable, the poll data presents its own challenges. Since our motivating problem was the deficiency of dense state-level polling, we must do the best we can with what exists and use that to train and benchmark the method here and establish its feasibility for extrapolation to unpolled states and times. To that end, about 1,200 state-level polls during the 2012 campaign were collected from Pollster.com using their API, and converted to Obama vote share as a proportion of the two-party intended vote in that state on that day. Of course, the polling tends to focus on a certain subset of

[5]No stop words or stemming was used: america, american, and americans, for instance, are all different words with different meanings.

states, so only states with more than 15 polls during our two-month period were retained, leaving 24 states. Even with 15–60 polls per state, many if not most days remain unpolled for most states, and of course each poll is subject to the usual survey error. Thus for each state the collected polls were smoothed and interpolated across our 67-day period.[6] Figure 1 shows the original and smoothed polls for Ohio.

[Figure 1 about here.]

4 The testing procedure and models

Recall that the fundamental question is whether a time-series or TSCS dataset generated purely from Twitter data can track some real-world measure out-of-sample. The appeal of sentiment or frequency measures is that they are inherently out-of-sample (assuming no post-hoc manipulation of lags or sentiment specification). Given how poorly these approaches seem to work though, the alternative approach here is to fit a more complex model on past data to "predict" future polls (eg) using only Twitter data. Since the data are very high-dimensional, single-sample tests without out-of-sample validation will surely overfit the data by finding random features in the text that match the variation in the dependent variable. Thus the need for out-of-sample testing, which in turn requires large quantities of observations.

But within this general requirement for out-of-sample validation, there is a plausible hierarchy of progressively more stringent tests, each with its own substantive meaning. The

[6]Because the polls were collected at wildly varying intervals within each state, standard smoothers like cubic splines or loess tended to produce overly erratic sequences. The best method appeared to be a simple KNN smoother, where each day's value is simply the average of the 2-8 nearest polls, where that window varies depending on how often the state was polled: this tends to produce a time series that both shifts smoothly and retains enough temporal variation to be useful. These decisions were of course made prior to testing. However, afterwards a variety of different smoothers were examined, and the procedure generally works across various approaches, although the smoothest and noisiest series both work less well than the various partially smoothed series (see note 22 for more).

most basic test, akin to some of the benchmarks discussed in Section 2, is to fit a model on the polls and text prior to the election, and then use Twitter text alone just before election day to predict the state-level election results. This is a genuine out-of-sample test, and finding a set of textual features that genuinely correlate with the Obama vote across dozens of states and a wide range of variance in opinion is no mean feat. But once again, recalling from section 2 point [S] (statistics), our N of 24 is quite small, and in addition, recalling point [B] (benchmarks), we are extremely unlikely to do better than the polls themselves in predicting election returns, since they were designed to predict precisely that and tend to be run at the most dense shortly before the election. A significantly tougher test would be to predict vote shares in states outside of our 24-state training set – which would be out-of-sample not only in time, but also in space – although here the benchmarks are a little less clear, apart from coarse measures of accuracy such as R^2 or which states are correctly assigned to Obama vs Romney.

A better way to increase our N to allow proper testing is, in effect, to repeat the election prediction multiple times: ie, fit the text to the polls over some m days prior to day t, and then use the text on day t to "predict" the polls on day t. Thus to create a sufficiently large N, the 24 state predictions for each day t can be accumulated into a single data set: fit on the m days prior to t, predict t, then roll the window forward a day and repeat; stack all these predictions into a predicted TSCS dataset that can then be compared with the true poll measures.

In addition to a testing procedure, this is also precisely the approach we would take to create a dense interpolated rolling or real-time poll across all our states, giving us poll measures for states that were unpolled that day, and potentially measures that reflected today's events before they register in today's polls. But for this to be useful, we must be able to do a better job predicting today's polls using today's text than we could do simply

extrapolating yesterday's polls into today. To do this, we must be able to track variation not just across states, but within states over time, and do so better than a simple extrapolation from past polling data can do. To the degree that we are interested in within-state variation over time, we must isolate out the cross-sectional variation and see whether the within-state R^2 (for instance) is higher using the text than merely extrapolating from the polls alone. This is a very high benchmark, though the height of that bar depends in part on how clever we are in extrapolating from the polls themselves. Two straightforward benchmarks are tested here: a direct extrapolating from the past polling level into tomorrow,[7] and a somewhat more sophisticated extrapolation that utilizes a linear trend. There are of course more complex models one could use,[8] but already the bar here is considerably higher than most of what has come before in this domain.

Having laid out the general procedure, the next task is to specify how to best model the TSCS polls as a function of 10,000 features in order to generate our text-based poll predictions. Three established machine learning algorithms are tested, along with a fourth one developed here that is designed to suit the TSCS data structure. The algorithms tested here are three of the most successful and well-established high-dimensional methods currently in widespread use: random forests, support vector machines, and elastic nets.

Elastic net is a general-purpose feature selection algorithm that combines L1 (lasso) and L2 (ridge regression) regularization methods (Zou & Hastie 2005), and is well suited to high dimensional problems like these. It uses standard OLS regression methods along with shrinkage parameters (λ_1 and λ_2) to drive most of the feature coefficients (β coefficients)

[7] Which, note, produces a predicted series for each state that is not constant over time, since it is based on a rolling window which is shifting over time; indeed, this approach by itself accounts for about 18% of the within-state variation, as we will see.

[8] Although higher-order trends (quadratic, etc) were examined, they don't seem to offer any additional accuracy in predicting tomorrow's polls from past polls above the linear trends.

to 0:

$$\arg\min_{\boldsymbol{\beta}} \left\{ ||\mathbf{y} - \mathbf{X}\beta||^2 + \lambda_1 ||\boldsymbol{\beta}||_1 + \lambda_2 ||\boldsymbol{\beta}||^2 \right\}$$

Support vector machines (Cortes & Vapnik 1995), on the other hand, were originally designed for classification rather than continuous dependent variables, but work for the latter case as well. The basic idea is to find the best hyperplane $(\mathbf{w}, b)$ that separates the two classes of points, but this can be weighted when the observations are continuous. It is less suited to feature selection, but because spatial kernels can be directly chosen, it is quite flexible in fitting the separating hyperplane to complex non-linear data:

$$\arg\min_{\mathbf{w},b} \max_{\boldsymbol{\alpha}\geq 0} \left\{ \frac{1}{2}||\mathbf{w}||^2 - \sum_{i=1}^{n} \alpha_i [y_i(\mathbf{w}'\mathbf{x}_i - b) - 1] \right\}$$

Random forests (Breiman 2001) are especially well-suited to out-of-sample prediction, since they were designed to be trained via cross-validation. The end results is essentially a weighted set of flexible neighborhoods used to predict new values ($f(\mathbf{x})$); these neighborhoods are generated by repeated "trees" ($f_m(\mathbf{x})$) that cleave the space via cut-points (leaves L_t with cutpoints $\mathbf{k}$) and are then aggregated into the final forest:[9]

$$\begin{aligned} f(\mathbf{x}) &= \sum_{m=1}^{M} \frac{1}{M} f_m(\mathbf{x}) \\ f_m(\mathbf{x}) &= \sum_{t=1}^{T} w_t \mathbb{I}(\mathbf{x} \in L_t) \\ \mathbb{I}(\mathbf{x}) &= \underset{\mathbf{t} \text{ leaves, } \mathbf{k} \text{ cuts}}{\arg\min} ||\mathbf{y}_{tk} - \bar{y}_{tk}||^2 \end{aligned}$$

These three approaches cover feature selection, complex non-linear functions, and out-of-sample maximization. However, none of them are especially well-designed for temporal or TSCS data structures. As we will see, although these established methods do manage to

[9]For more details on these three approaches, see any machine learning textbook, such as Hastie et al. (2009).

leverage the textual information to a degree, they fail to improve upon the most stringent benchmark, where the future polls are predict from past polling data alone using fixed effects and time trends. That is, they fail to sufficiently utilize the textual information to actually improve upon the best text-less poll predictions. It should be said, though, that were the benchmarks lower – eg, were the standard, as in some of the papers cited in Section 2, only to predict polls better than chance – then all of these methods would pass with flying colors. It is only when the bar has been raised using the four criteria from Section 2 that these methods fail, and reveal that the text is not actually adding predictive power to the polls alone.

To better leverage the combination of TSCS data with the large number of textual features, the method here was designed to adapt a method similar to the L1 regularization in the Elastic Net method, but which allows us to directly incorporate fixed effects and time trends, essentially picking out the text features that are most predictive of polls over and above the state-level fixed effects and time trends based on past polls. This approach aggregates an ensemble of simple models (as in a random forest), where only a subset of these simple models are given a non-zero weight (as in L1 regularization) when taking the weighted average of the simple models. Essentially, each sub-model is a simple TSCS model that uses a single textual feature plus the fixed effects and time trends:

$$p_{jt} = \beta_j + \tau t + \beta_k w_{kjt} + \epsilon_{kjt}, \quad \text{for } k \text{ in } [1...10{,}000] \tag{1}$$

where p_{jt} is the mean poll-measured vote intention in state j on day t; w_{kjt} is the frequency of word k in state j on day t; β_j is a fixed effect for that state; τ is a time trend; and β_k is the effect of word k.[10] To avoid the sorts of over-fitting that even L1 regularization is

[10] A natural extension of this model might be to estimate coefficients that vary within states – β_{jk} – with shrinkage toward overall means β_k in proportion to state sample size. More generally, we might use the known demographics of each state to boost model fit using multilevel regression and poststratification

vulnerable to, each of these 10,000 models is estimated via a separate OLS regression, one for each word. Equation (1) shows the full model (M5, below); it can also be estimated with various subsets of the three components in Equation (1) (models M1 through M4 below).

To generate a new prediction, one simply averages the text-based predictions (β_k) over all the features we choose to retain, and adds back in the fixed effects and time trend:

$$\hat{p}_{j,t+1} = \hat{\beta}_j + \hat{\tau}(t+1) + \sum_{k \in \lambda(\sigma_{\beta_k})} \hat{\beta}_k w_{kj,t+1} \tag{2}$$

where the fixed effects are the mean values over the in-sample window, the time trend is the overall trend over that window, and the $\hat{\beta}_k w_{kj,t+1}$ are only those unigrams with $\hat{\beta}_k$ above some precision threshold λ. This threshold, which determines which subset of the features are retained for prediction purposes, is analogous to the λ_1 penalty in the elastic net, but is derived directly from the OLS p-values and sets all values below the threshold entirely to 0.[11] As well as drawing upon the ensemble literature, this approach is inspired by the multiple testing literature, where the false discovery rate (Benjamini & Hochberg 1995) correction essentially reweights the p-values when large numbers of tests are made, lowering the p-value threshold on what counts as a true positive, while allowing a certain percentage of false positives to also coexist.[12]

methods (MRP), but this must remain for later work.

[11] One additional nuance is that these three factors are reweighted (in-sample) before prediction by an additional regression of the three on p_{jt}; this does not affect R^2 but improves the mean absolute error, and imparts a degree of effective shrinkage on the retained β_k factors, increasing both out-of-sample performance and the similarity of this method to standard L1 regularization – particularly since an inspection of the coefficient profile plots for L1 regularization shows that for any given λ, most coefficients are either 0 or near their original values. Also, note that λ is the sole free parameter in the model apart from m (which is shared across all models), and both were selected using a brief coarse-grained search using only September data, out of consideration of the aforementioned dangers of over-fitting. Values of m tested were 1, 2, 3, or 4 weeks with three weeks selected; values for λ tested were 0.01, 0.001, 0.0001, and 0.00001, with 0.001 selected, suggesting a relatively low false positive rate given that between 50 and 500 features are usually retained out of the 10,000 depending on whether fixed effects and time controls are included. Post-hoc analysis shows that the results are robust to small variations in m or λ.

[12] The alternative is the familywise error rate (Dunn 1961), which sets a threshold such that there is, for instance, a 0.05 chance than even one positive is false, which is generally now considered too high a

In summary, this new model combines aspects of the more established methods – in particular, averaging ensembles of simple models with a threshold determining which subset of features are retained – while also better incorporating the specific time-series cross-sectional structure of the data in this particular domain. As we see in the next section, the result is that it is the only approach that actually manages to leverage the textual data to genuinely improve upon the poll-based predictions.

5 Results

To briefly recap, our fundamental question is whether the Twitter textual data can be used to predict polling variation and changes. But the deeper question raised in Section 2 is, better than what? What is our benchmark for success? Table 1 presents the results from our out-of-sample testing [O], where the upper area shows which factors are utilized by the model (textual features; state fixed effects; time trends) [T], and the lower area shows the results of the models measured in two ways: the mean absolute error between the correct and predicted results, and the R^2. By simple statistical measures of significance, all models do far better than chance alone [S], but this is a relatively low bar. More important as a measure of actual utility is the benchmark [B] set by models M2 and M4, which use only the state fixed effects and poll time trends (ie, no text) to predict the future polls. Doing better than these benchmarks is the true test of whether a model can leverage the textual signal to do genuinely useful predictive work.

Recall that the full dataset is the 9 weeks leading up to the election, and the predictions are based on a rolling window that fits each model on the three previous weeks and predicts day t's polls based on some combination of day t's Twitter text and the fixed state effects and time trend. When these one-day-ahead predictions are combined, we have an

standard, and unsuited to predictive tasks.

aggregated test set of 42 days x 24 states for all the models.

Model 1 estimates the pure text model using only the text term in Equation (1), without making any use of the fixed effects or time trend. The mean absolute error (MAE) over this pooled data set is about 2 percentage points – ie, most states are guessed out-of-sample within a couple points of their true values; the pooled R^2 is 0.77, and the average R^2 for each day is a bit higher at 0.82. This is a solid performance, and certainly answers the statistical significance question from point [S] – the p value from regressing the true on the text-predicted polls is $< 1 \times 10^{-10}$ (cluster-robust standard errors). But beyond the p value, what is our benchmark [B]? Is an MAE of around 2 any good?

[Table 1 about here.]

A clearly relevant benchmark, and one commonly used, is to examine the election results themselves and see whether the text alone (M1) can predict state-level outcomes. The left panel of Figure 2 shows the poll-based predictions on election eve versus the election results, and as we can see, beating even this basic benchmark will be quite difficult, since the polls alone essentially got no state outcome wrong. The right panel, however, shows that the text alone in fact does nearly as well, also getting almost none of the well-polled states wrong (triangles). But a much more stringent and interesting benchmark is whether the text model can be extended to unpolled states which have never been used for any training. And there, only two state outcomes (circles) are significantly wrong, although the percentage error between predictions and outcomes unsurprisingly increases.[13] Thus not only can the text come close to matching the poll-based election predictions, but in states with little to no polling, the text model trained on the well-polled states can effectively predict opinion in unpolled states, albeit with somewhat lesser precision.[14]

[13]Not shown is Maryland, with a predicted score of 1.25. To reduce these occasionally extreme effects, it may be more effective to also smooth the text predictions over a few days.

[14]By comparison, the off-the-shelf methods do considerably less well in predicting the electoral outcome

But of course for the most part, the outcomes of these unpolled states was never in doubt. That doesn't mean that there isn't great utility in measuring exact opinion levels rather than caring only about electoral outcomes. But it does mean that if we want to raise the bar still further, and determine whether the text-based approach can predict (or interpolate) polls better than the polls alone even in well-polled states, we will need to return to our rolling-window 24-state tests.

For M1, the within-state R^2 (ie, explaining the variation over time) is close to 0, illustrating again that this model is mainly picking up cross-sectional variation – useful for predicting un-polled states, but less useful for predicting forwards in time. If fact, if we simply use the mean Obama vote intention for each state over the past m days to predict vote intention in day $t+1$ (M2),[15] we explain most of the pooled R^2 and reduce the MAE greatly relative to M1. M2 in fact also explains 19% of the within-state variance over time.

If we combine the text features from M1 with the fixed effects in M2, the within-state R^2 nearly doubles (M3),[16] showing that we are now utilizing (different) text features to augment M2 and better track the temporal changes in polls. However, if we raise the poll-alone benchmark still higher, and add time trends to M2 to yield M4,[17] we again do better than the text-based M3, suggesting that although the text in M3 picks up the temporal shifts in polling, it does not do so as well as a simple poll-based time trend. M4, however, is a very high benchmark, higher than those that are used in almost any of the works discussed in Section 2.

in unpolled states using text data alone. For all 50 states, the M1 method gets 4 incorrect, or 2 not including edge cases, as can be seen in Figure 2. The next-best method is the elastic net ($\lambda_1 = 0.001$), which gets 8 states wrong, or 5 not including edge cases. The SVM does even less well, with 13 errors, or 8 not including edge cases. And the random forest does the least well, with 17 errors, or 16 not including edge cases. It is perhaps unsurprising that of the established methods, the elastic net does closest to the M1 method, since it is the method the M1-M5 approaches most resemble.

[15] M2 corresponds to solely the first right-hand-side term in Equation 1.

[16] M3 corresponds to the first and third right-hand-side terms in Equation 1.

[17] M4 corresponds to the first and second right-hand-side terms in Equation 1.

Nevertheless, the full model M5,[18] by combining text, fixed effects, and time trends, does manage to out-perform our best polls-alone benchmark M4, on both the mean average error measures and the R^2 measure, most importantly the within (temporal) R^2.[19] To illustrate the level of temporal accuracy, Figure 3 shows the predicted and (smoothed) truth for Ohio (using M5), a state that is predicted with about the median level of MAE; the text tracks the early October dip due (arguably) to the notorious first debate – perhaps with a bit more lag, but also with with much less volatility than the actual polls shown in Figure 1.

[Figure 2 about here.]

By comparison, the standard machine learning algorithms do less well with this prediction task, even when given state dummies and time counters as additional features. The Random Forest does reasonably well with cross-sectional variation, but less well with the all-important within-state variation. The SVM does less well than the Random Forest on either (illustrating a general weakness of SVM's for very high-dimensional data). The Elastic Net, interestingly, does better at either cross-sectional or within-state variance depending on the λ_1 level,[20] although either way it fails to surpass what can be done by extrapolating from the polls alone.[21] Only M5 manages to improve on the polls-only M4, mainly because it was purpose-built to best exploit the fixed effects and time-trends along with the high-dimensional textual information.[22]

[18] Equation 1 in full.

[19] Table 1 shows the R^2 using the interpolated polls, but the results are similar using only the real polls. For instance, M5 still out-performs M4 at $p < 0.05$.

[20] For all values of λ_1, the optimal setting for λ_2 was 0; ie, there was no useful L2 shrinkage here. For $\lambda_1 = 0.001$, many β_k are retained, as in M1; for $\lambda_1 = 0.1$, only a very few features are retained, as in M5. It is possible that "sweeping out" the fixed effects – ie, fitting the model on the residuals – would allow us to combine the elastic net results better with the TSCS structure.

[21] A Superlearner ensemble of these three methods was also tested, and offered only slight overall improvement, along with the usual cost in interpretability.

[22] These results are robust to variations in the method for smoothing the polls. A variety of fixed and variable smoothing windows were tested, ranging from the 1 to 8 nearest polls to the (missing) interpolated

[Figure 3 about here.]

One final important question is how well the model fit on the m days up through t continues to work for $t+2, t+3$, etc. That is, how long to these fitted models last? Again, the appeal of the sentiment-based approach is that the model should last as long as language itself remains relatively stable – if the sentiment-based approaches worked. In the present case, the cross-sectional fit works quite well over time: if we generate a new TSCS test set consisting of all the $t+2$ predictions over the rolling window, or another TSCS set consisting of $t+3$ predictions, etc, M1 retains its accuracy quite well over time, rarely falling below 0.90 for mean cross-sectional R^2. However, within-state R^2 quickly falls for all models, as shown in Figure 4; this drop is particularly notable after a week or so, although this drop is common to all the models tests, including the ones based only on past polling data.

[Figure 4 about here.]

We have seen that the text-based model M1 does a very good job of predicting poll levels across states, even when extended to unsampled states, and that the text-augmented model M5 does a better job of tracking poll variation than even a fairly careful extrapolation using past polls and trends can do. These results suggest that we now have a robust model that can extrapolate polls to unmeasured areas and finer time scales than currently exist. The final section examines to what these models can tell us about what is going on in public opinion and the campaign, and how that may affect vote intention.

day. But whatever the interpolation procedure, the accuracy rankings of the methods in Table 1 remains almost entirely the same for both the MAE and R^2 metrics. Perhaps the most stringent metric is using the within-R^2 using only the real (non-interpolated) polls; this is naturally lower than the interpolated within-R^2 shown in Table 1, but it is unbiased by the smoothing method. For M5, the real-value within-R^2 ranges from 0.12 to 0.16 depending on the smoothing method; for the elastic net ($\lambda_1 = 0.1$) it ranges from 0.02 to 0.03; for the SVM, from 0.01 to 0.02; and for the random forest, from 0.00 to 0.02. For most cases, the best performance is using the variable-window smoother employed in the main results (eg, Figure 1), even though alternative smoothers were not tested for out-of-sample predictive ability prior to analysis.

6 Textual content

In addition to allowing us to measure vote intention across states and time, the other benefit of these social media measures is that they provide direct insight not just into what Twitters users say when speaking about Obama, Romney, or other political topics, but into which words and topics are specifically associated with geographical or temporal variations in genuinely representative surveys of vote intention. Models 1, 3 and 5 each capture different subsets of features (geographical, long-term trends, and short-term events) that are associated with state-level measures of opinion;[23] the out-of-sample testing suggests that these correlations are not mere coincidence, but are picking out the aspects of Twitter speech that track forward in time with opinion change among representatively surveyed voters.

Looking first just at Twitter behavior as it coarsely correlates with vote intention, Figure 5 shows that political interest in both candidates has a local peak when states are most competitive (the 0.50 line), but in an interesting asymmetry, mentions for both candidates rise with increasing Obama vote share, flattening out somewhat as the state or time period becomes strongly pro-Obama. By themselves, these figures only paint a rough picture of the relationship between two specific words and vote intention, and of course we know nothing about the intentions or ideologies of the tweeters. In fact, these two words are not among the most predictive even for the most simple task of distinguishing cross-sectional (state level) differences in vote intention. Table 2 shows the most predictive features from models M1, M3, and M5. The first row shows the most significant[24] β_k from M1, ranked by most positive in sign (pro-Obama) and negative in sign (pro-Romney). As expected, many of these features are explicitly geographical, although further into these

[23] Recall that each of these models estimate a different set of β_k, depending on whether fixed effects and/or the time trend are included in the estimation of Equation (1).

[24] Which are also usually the largest in absolute value.

lists are many more substantial words and hashtags (about 500 features are retained for each run of M1). Notably though, the pro-Romney list has more explicitly political terms, a trend which continues through M3 and M5. In addition, by far the strongest term correlating with pro-Romney vote intention is "rt", indicating a retweet. Past work has suggested that Republican Twitter users tend to be more cohesive and retweet each other more often than on the left (Conover et al. 2012, Hoang et al. 2013), which may in turn serve to focus that community on a few more cohesive national political issues. By contrast, although not in the top 20, one of the highly correlated terms on the left is "http", most of which are links to pro-Obama external content; this in addition to "#socialmedia" and "#google" on the left again may suggest a Twitter population with more outward links to other websites, content, or social media, although to confirm this would require direct measures of the ideology of the tweeters.

[Figure 5 about here.]

In Model 3, the cross-sectional variation is mainly absorbed by the state fixed effects, leaving features associated with gradual trends over time. While Model 1 was dominated by the hashtags that correlate strongly with partisan regions or communities, now we see much more explicitly political topics and current events. Many of these are what we might expect: "cia" on the right (reflecting the Benghazi controversy) and the "#47percent" on the left. But there are also less expected elements, including four variants of "endorse" on the right, and a variety of numbers and "percent" on the left, many of which (upon direct inspection of some of the tweets) appear to be references to the polls themselves.[25] As with the Model 1 features, the terms here seem more internal and political (rt, endorsements)

[25]One could attempt to run a topic model on these aggregated state-day "documents," such as Supervised Latent Dirichlet Analysis (Blei & McAuliffe 2007), where the poll measures provide the supervision. However, with so few retained features, these methods perform far less well, from a human-interpretation point of view, than simply inspecting the feature lists with an expert eye.

on the right, and more external, informational, and geographical (http, polls) on the left; but confirming these impressions would require a multi-year, multi-campaign comparison.

[Table 2 about here.]

Finally, Model 5 – the one that gives us the features that genuinely improve upon the polling alone – moves even more towards the short-term events that are associated with temporal opinion shifts. The terms associated with Benghazi become much more prevalent on the right, while the terms on the left remain generally less informative, and may indeed be driving less of the predictive power than those on the right.

We can also connect these transient events to the timeline of the campaign, by plotting the T statistics for the top 20 β_k on either side over time, as our in-sample window rolls forward. Figure 6 shows the features from M5 associated with pro-Obama shifts (left) and those associated with pro-Romney shifts (center). For any given day, these are the features (words) that are most associated with changes in the polls and that have the greatest effect on predicting polling changes: they suggest how changes in Twitter content are both driven by, and predictive of, short-term changes in public opinion. Noted on each figure are the three debates, with a clearly discernible peak around the first debate for both, and arguably around the third for Romney. However, these effects are much weaker for Obama than Romney, suggesting that most of the predictive power for all of these models may come from the Romney side of the equation. To zoom in on those terms that are driving the greatest effect on the Romney side, the right panel in Figure 6 shows those features explicitly connected to Benghazi: embassy, embassies, controversy, slain, cia, intelligence. This quite clearly shows the two surges in predictive correlation at the first and third debates, suggesting that this issue may indeed have had a role in driving both Twitter attention and vote intention. Ultimately, however, the content analysis here amounts to a single case study, and is less dispositive than suggestive for future study of the interplay

between political events, opinion, and social media use.

[Figure 6 about here.]

7 Conclusion

We have seen that, correctly modeled, political tweets in sufficient quantity can indeed be used to measure, extrapolate, and interpolate properly representative polling variation, both across states and over time. A testing regime has been provided that satisfies most of the deficiencies of previous social media measures: the N is large, the tests statistically validated, the benchmark high, and the model carefully fit in-sample and tested out-of-sample and forward in time. The linear feature-selection model itself appears to work well, and better than powerful methods such as Random Forests, Support Vector Machines or Elastic Net (although as with any machine learning method, the results can presumably be further improved upon). This approach can serve as a model for a variety of social-media-based measures of true public opinion, even in domains where the training data is less abundant, although the need for at least some training data[26] remains an important constraint for exporting this procedure into highly under-surveyed domains.

Finally, in addition to validating the social media polling measure as a plausible tool that could be used with historical or real-time data, the textual features discussed in the previous section yield potential insights into large-scale geographical variation and short-term topical changes in vote intention. Beyond identifying the salient themes and events from 2012 – including the surprisingly evident influence of the debates – we discover what may be more general differences in online partisan behavior, with a more cohesive and

[26]One additional avenue for future exploration is just how much training data is necessary in order to be able to model temporal as well as cross-sectional variation. It may be that as few as a couple surveys per region per training set are needed, and preliminary tests suggest that the long-term viability of a fitted model can be increased at the cost of short-term accuracy, but the practical limits of this remain to be determined.

nationally-oriented right, and possibly a more regional and outward-looking left. These results provide not just a tool for generating survey-like data, but also a method for investigating how what people say and think reflects, and perhaps even affects, their vote intentions.

References

Albrecht, Steffen, Maren Lübcke & Rasco Hartig-Perschke. 2007. "Weblog campaigning in the German Bundestag election 2005." *Social Science Computer Review* 25(4):504–520.

Benjamini, Yoav & Yosef Hochberg. 1995. "Controlling the false discovery rate: a practical and powerful approach to multiple testing." *Journal of the Royal Statistical Society. Series B (Methodological)* pp. 289–300.

Bermingham, Adam & Alan F Smeaton. 2011. "On Using Twitter to Monitor Political Sentiment and Predict Election Results." *Sentiment Analysis where AI meets Psychology (SAAIP)* p. 2.

Blei, David M & Jon D McAuliffe. 2007. Supervised Topic Models. In *NIPS*. Vol. 7 pp. 121–128.

Breiman, Leo. 2001. "Random forests." *Machine learning* 45(1):5–32.

Ceron, Andrea, Luigi Curini, Stefano M Iacus & Giuseppe Porro. 2014. "Every tweet counts? How sentiment analysis of social media can improve our knowledge of citizens' political preferences with an application to Italy and France." *New Media & Society* 16(2):340–358.

Chung, Jessica Elan & Eni Mustafaraj. 2011. Can collective sentiment expressed on twitter predict political elections? In *AAAI*.

Conover, Michael D, Bruno Gonçalves, Alessandro Flammini & Filippo Menczer. 2012. "Partisan asymmetries in online political activity." *EPJ Data Science* 1(1):1–19.

Cortes, Corinna & Vladimir Vapnik. 1995. "Support vector machine." *Machine learning* 20(3):273–297.

Dunn, Olive Jean. 1961. "Multiple comparisons among means." *Journal of the American Statistical Association* 56(293):52–64.

Gaurav, Manish, Amit Srivastava, Anoop Kumar & Scott Miller. 2013. Leveraging candidate popularity on Twitter to predict election outcome. In *Proceedings of the 7th Workshop on Social Network Mining and Analysis*. ACM p. 7.

Gayo-Avello, Daniel. 2013. "A meta-analysis of state-of-the-art electoral prediction from Twitter data." *Social Science Computer Review* 31(6):649–679.

Gayo-Avello, Daniel, Panagiotis Takis Metaxas & Eni Mustafaraj. 2011. Limits of electoral predictions using twitter. In *ICWSM*.

Ghitza, Yair & Andrew Gelman. 2013. "Deep interactions with MRP: Election turnout and voting patterns among small electoral subgroups." *American Journal of Political Science* 57(3):762–776.

Hastie, Trevor, Robert Tibshirani, Jerome Friedman, T Hastie, J Friedman & R Tibshirani. 2009. *The elements of statistical learning*. Vol. 2 Springer.

Hoang, Tuan-Anh, William W Cohen, Ee-Peng Lim, Doug Pierce & David P Redlawsk. 2013. Politics, sharing and emotion in microblogs. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. ACM pp. 282–289.

Huberty, Mark Edward. 2013. Multi-cycle forecasting of congressional elections with social media. In *Proceedings of the 2nd workshop on Politics, Elections and Data*. ACM pp. 23–30.

Huckfeldt, R Robert. 1995. *Citizens, politics and social communication: Information and influence in an election campaign*. Cambridge University Press.

Jansen, Harold J & Royce Koop. 2005. "Pundits, Ideologues, and Ranters: The British Columbia Election Online." *Canadian Journal of Communication* 30(4).

Jungherr, Andreas, Pascal Jürgens & Harald Schoen. 2012. "Why the pirate party won the german election of 2009." *Social Science Computer Review* 30(2):229–234.

Kenski, Kate & Natalie Jomini Stroud. 2006. "Connections between Internet use and political efficacy, knowledge, and participation." *Journal of Broadcasting & Electronic Media* 50(2):173–192.

Lax, Jeffrey R & Justin H Phillips. 2009. "How should we estimate public opinion in the states?" *American Journal of Political Science* 53(1):107–121.

Lazer, David, Ryan Kennedy, Gary King & Alessandro Vespignani. 2014. "The parable of Google Flu: traps in big data analysis." *Science* 343(14 March).

Livne, Avishay, Matthew P Simmons, Eytan Adar & Lada A Adamic. 2011. The Party Is Over Here: Structure and Content in the 2010 Election. In *ICWSM*.

Lui, Catherine, Panagiotis T Metaxas & Eni Mustafaraj. 2011. On the predictability of the US elections through search volume activity. In *Proceedings of the IADIS International Conference on e-Society*.

Metaxas, Panagiotis Takis, Eni Mustafaraj & Daniel Gayo-Avello. 2011. How (not) to predict elections. In *Privacy, security, risk and trust (PASSAT), 2011 IEEE third international conference on and 2011 IEEE third international conference on social computing (SocialCom)*. IEEE pp. 165–171.

O'Connor, Brendan, Ramnath Balasubramanyan, Bryan R Routledge & Noah A Smith. 2010. "From Tweets to Polls: Linking Text Sentiment to Public Opinion Time Series." *ICWSM* 11:122–129.

Park, David K, Andrew Gelman & Joseph Bafumi. 2004. "Bayesian multilevel estimation with poststratification: state-level estimates from national polls." *Political Analysis* 12(4):375–385.

Pennacchiotti, Marco & Ana-Maria Popescu. 2011. "A Machine Learning Approach to Twitter User Classification." *ICWSM* 11:281–288.

Sang, Erik Tjong Kim & Johan Bos. 2012. Predicting the 2011 dutch senate election results with twitter. In *Proceedings of the Workshop on Semantic Analysis in Social Media*. Association for Computational Linguistics pp. 53–60.

Skoric, Marko, Nathaniel Poor, Palakorn Achananuparp, Ee-Peng Lim & Jing Jiang. 2012. Tweets and votes: A study of the 2011 singapore general election. In *System Science (HICSS), 2012 45th Hawaii International Conference on*. IEEE pp. 2583–2591.

Tumasjan, Andranik, Timm Oliver Sprenger, Philipp G Sandner & Isabell M Welpe. 2010. "Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment." *ICWSM* 10:178–185.

Wong, F, Chee Wei Tan, Soumya Sen & Mung Chiang. 2013. Media, pundits and the us presidential election: Quantifying political leanings from tweets. In *Proceedings of the International Conference on Weblogs and Social Media*.

Zou, Hui & Trevor Hastie. 2005. "Regularization and variable selection via the elastic net." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 67(2):301–320.

Table 1: Accuracy in matching out-of-sample text-predicted polls to true polls.

	M1	M2	M3	M4	M5	Random Forest	SVM	Elastic Net[c] $\lambda_1 =$ 0.001	Elastic Net[c] $\lambda_1 =$ 0.1
Twitter text	×		×		×	×	×	×	×
State fixed effects		×	×	×	×	×	×	×	×
Time trend				×	×	×	×	×	×
MAE (smoothed)[a]	1.91	0.60	0.53	0.54	**0.51**	1.53	3.53	0.88	3.76
MAE (real)[a]	2.16	1.38	1.32	1.30	**1.27**	1.81	2.76	1.53	3.21
R^2 Pooled[b]	0.77	0.98	0.98	0.98	**0.98**	0.90	0.19	0.95	0.01
R^2 Within[b]	0.03	0.19	0.36	0.37	**0.40**	0.09	0.07	0.08	0.22

Note: $N = 24$ states x 42 days = 1008. × = variable included in model. All variables in M1-M5 are significant at $p < 0.00001$ (cluster-robust standard errors). Best scores are in bold.

[a] MAE = mean absolute error (pct. points) between polls (real or smoothed) and predictions.

[b] R^2 Pooled = variance across all observations; Within = variance within states.

[c] Elastic net performance optimal at $\lambda_2 = 0$ for all λ_1.

Table 2: The text features most associated with pro-Obama and pro-Romney poll shifts.

	Pro-Obama	Pro-Romney
Cross-sectional (M1)	#ucwradio, #ny, #politics, ny, #hot, brooklyn, reuters, #business, ma, boston, cuomo, #google, york, #hitechcj, #socialmedia, #nytimes, scott, massachusetts, elizabeth, #boston, year, full	rt, socialists, indiana, #ccot, #dloesch, #ocra, montana, #dems, #insen, #patdollard, #theblaze, donnelly, #townhallcom, #jjauthor, #mo, mo, missouri, #lnyhbt, o, bjp
Long-term shifts (M3)	75, univision, eat, narrative, rich, plane, 46, million, #47percent, help, return, replaces, wtf, tan, percent, delusional, congress, blind, dependency, dinosaur, #mapoli, #billjryan	cia, endorsed, endorsing, convicted, pre, nervous, mother, name, flips, endorses, volunteer, #prolife, endorsement, niggas, #kimsfirst, repeats, skin, miami, reviews, tried
Transient events (M5)	million, 75, narrative, #truthteam2012, pi, 30, baldwin, leaking, anne, #nhpolitics, #nh, attacked, #paulryan, eat, area, tied, #socialmedia, tammy, sikh, walker, speaker, warming	flips, #tricianc1, embassy, striking, stir, lack, concession, espa, cia, embassies, context, slight, skin, intelligence, couples, feelings, rand, controversy, repeats, slain

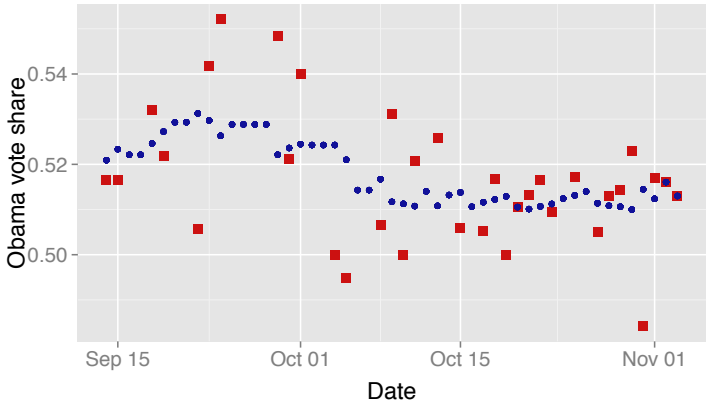

Figure 1: Ohio polls (boxes) and smoothed interpolated values (circles).

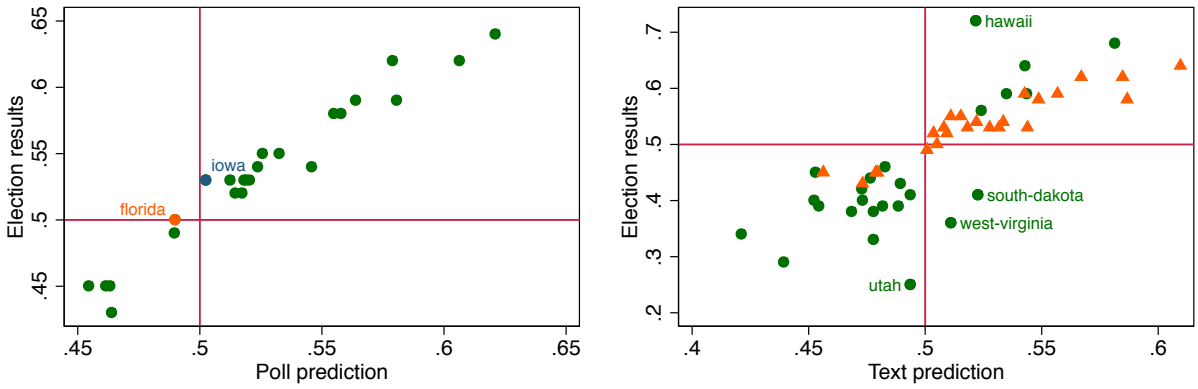

Figure 2: Left: Polls at 11/4/12 vs election results. Right: Pure text-based prediction on 11/4/12 from M1. Triangles: training states; circles: other states.

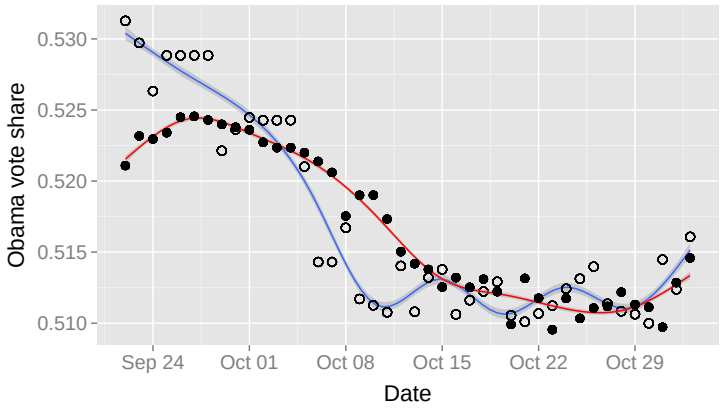

Figure 3: Predicted and actual polling for Ohio. Open circles: polls; filled circles: text-based predictions.

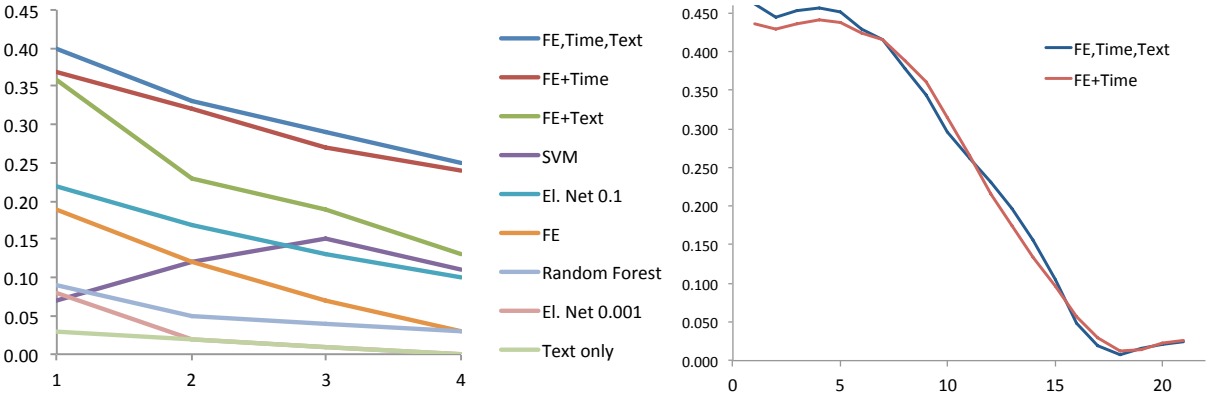

Figure 4: The accuracy of predicted polls as the text-based predictions are extended further from the fitting window (within-state R^2). Left: the decline of all models over 4 days. Right: the most effective models (M4 and M5) over 21 days.

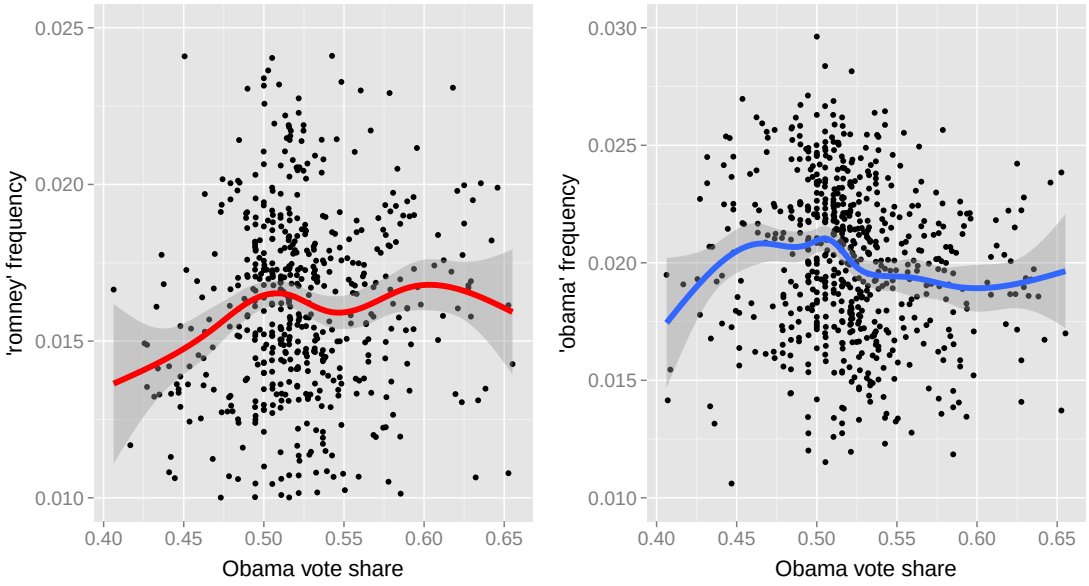

Figure 5: Candidate mentions: the frequency of "obama" and "romney" by intended Obama vote share (units: state-days, using only actual polls).

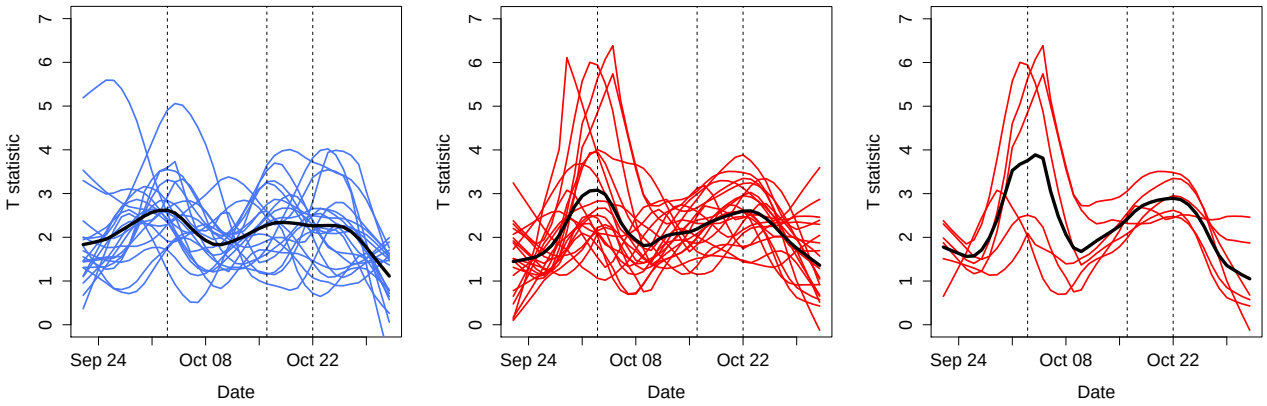

Figure 6: T statistics on the features most associated with pro-Obama changes (left) and pro-Romney changes (middle) in vote intention over time. Right: only those words associated with Benghazi: embassy, embassies, controversy, slain, cia, intelligence. The three debates are shown with dotted lines.