Who Needs Polls?
Gauging Public Opinion from Twitter Data

David Cummings <davidjc>, Haruki Oh <harukioh>, Ningxuan Wang <nwang6>

I. INTRODUCTION

Twitter is a social network website where users post and read messages called tweets which by default are publicly available. Various aggregate analyses of tweets have been used to model things in many areas such as the stock market [1], earthquakes [2], and pandemics [3]. For this project, we generated various metrics from Twitter data to measure the presidential approval rating, economic confidence, and the generic Congressional ballot. Since precise public opinion is unknown, we used public polling as a proxy and worked to generate metrics that would correlate well with the polls. Our results indicate that different approaches are better for different topics.

II. DATA

Twitter Data: We have 7 months of Twitter data, downloaded from Stanford Large Network Data Collection made available by Jure Leskovec.[4] This data spans from mid-June to the end of December, 2009 and contains over 476 million tweets. This data set does not comprise all tweets made in that time frame, instead they are a random subset of publicly available tweets. Twitter gives its users 140 characters to say whatever is on their mind. There are a variety of Twitter-specific language phenomena: hash tags like #stanford or #beatcal mark the topic of the tweet such that it can be easily searched for found by other users; many tweets include URL links to other webpages; the prefix "RT" marks a tweet as a re-tweet, or a copy of a message some other user posted. In our tweet parsing, we treat hash tags as their own words, ignore URLs since resolving them and searching their content would take prohibitively long (and many links from 2009 are no longer valid), and leave re-tweets as they are, except for one variation in which we throw out all re-tweets to measure only original tweets.

Polling Data: We chose three public polls to model: the presidential approval rating, economic confidence, and the generic Congressional ballot. These three were selected based on their free availability, as well as the density of the data: presidential approval and economic confidence data were available on a daily basis, and generic Congressional ballot data was available on a weekly basis. Various opinion polls are available on a biweekly or monthly basis, but with only 7 months of Twitter data to model against, it would be difficult to conclusively judge correlation on so few data points. Further, since all three polls gauge public sentiment on topics Twitter users are likely to talk about and post, as opposed to more obscure topics, we expected that we would be able to generate better-quality results. Our data comes from the Gallup Organization and Rasmussen Reports, two prominent US polling firms which both make selections from their polling data available online for download and use.

For the presidential approval rating, we used Gallup's daily polling from June to December 2009. [5] Presidential approval polls are generally reported in categories of Strongly Approve, Approve, Disapprove, and Strongly Disapprove; however, the end result is often compiled into a single index called the presidential approval rating. This rating is calculated by subtracting the sum of disapproving respondents from the sum of approving respondents, and normalizing over the total of all respondents. If a plurality approves, the index will be positive (up to +100), and if a plurality disapproves, it will be negative (down to -100).

For economic confidence, we again used daily polling data from Gallup from June to December 2009.[6] The calculation of economic confidence is more complex than that of the presidential approval rating: two polls, rating respondents' opinions of current economic conditions and overall economic outlook, are combined and scaled to a range of -100

to 100, where -100 represents 100% negative current conditions and outlook, and +100 represents 100% positive current conditions and outlook. While the calculation is more complex, the goal of this index is simple: to model how well Americans feel about the economy, and thus predict the economic climate to come.

For the generic Congressional ballot, we used data from Rasmussen, since Gallup only provided monthly data and Rasmussen had weekly data points.[7] This poll asks the question, "If a Congressional race were held in your district today, for which party would you vote?" In the US two-party-dominated system, responses of "Democrats" and "Republicans" are strongly negatively correlated, so the data is often represented as the difference in the percentage of responses for each party. This gives a single index, again ranging from -100 to 100, representing how much Americans prefer one party to the other. We represent preference for Democrats with positive scores and preference for Republicans with negative scores.

Smoothing: Gallup's daily polling data (used for presidential approval and economic confidence) was smoothed over a period of 3 days using a sliding time window. This is standard industry practice, and allows for the suppression of noise in the data while maintaining the shape of the overall curve. We used two time window increments: one further smoothed this time window to 6 days by averaging pairs of values 3 days apart; and the other smoothed to 15 days, thus emulating the 5-7 and 15-day windows used in other research. Rasmussen's generic Congressional ballot polls were only conducted on a weekly basis, for a total of 28 data points within the range of our Twitter data, so we did not smooth this data.

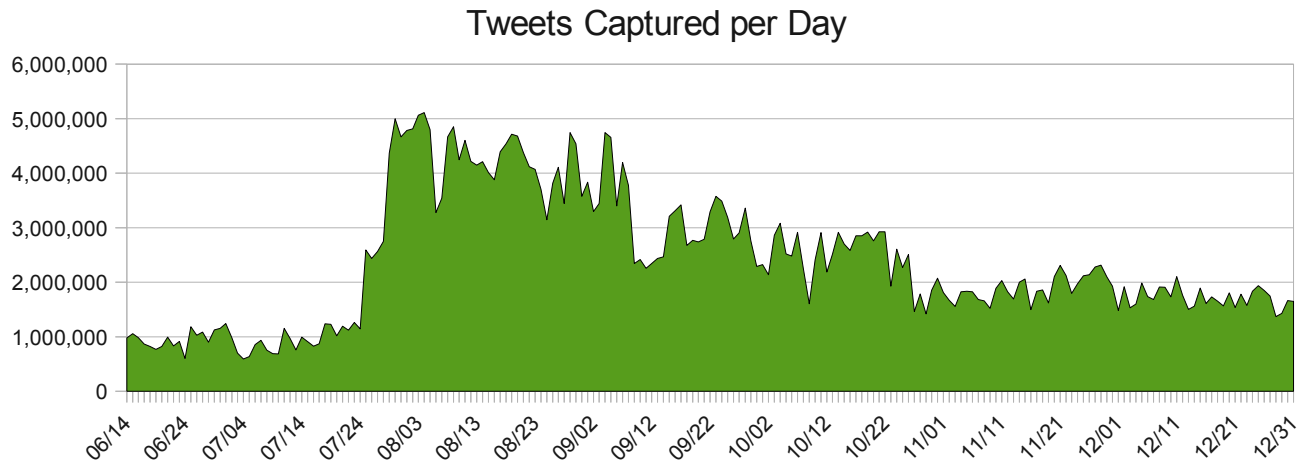
Caveats: Polls are not guaranteed to accurately represent public opinion, and indeed polls often diverge from reality due to systematic error. To cite one well-known example, pollsters who rely on conducting home phone interviews using numbers from a phonebook will never sample from the growing number of US citizens without a land-line phone. However, these polls are the best approximation of true public opinion that we can have, and so for the purposes of our research we treat it as the ground truth to be modeled. As another complicating factor, the overlap between the population of the US and the users of Twitter are not representative, much less comprehensive. Twitter users can even be international, or use languages other than English. Since Twitter users are primarily American, and foreign-language tweets represent a small fraction of our data, we did not expect these factors to significantly affect our approach.

III. METHODS

Volume: The first and simplest of our modeling methods is the volume metric. This represents the percentage of tweets that mention the topic or person in question over a set span of time. While crude, this metric does capture some intuitive information: if people are talking about a given person or group, then we have a general idea of their popularity, or at least their notoriety. (Distinguishing between the two is discussed further in the following sections.)

For the presidential approval rating, we used the string "obama"; for economic confidence, "job" or "economy"; and for generic Congressional ballot, "democrat" or "republican". As a simple form of stemming, matches were made with any word token containing these markers as a substring, such that tweets with the words "obamanomics", "jobs", and "democratic" counted toward the presidential approval, economic confidence, and Congressional ballot counts respectively. As a special case, since there are two opposing word markers for the generic Congressional ballot, we counted "democrat" as +1 and "republican" as -1, emulating the poll data.

To smooth the volume over time, we used a sliding time window to match the corresponding poll data (6, 7, or 15 days), took the number of all tweets mentioning the marker in the time frame, and divided by the total number of tweets in the time frame. At first, we thought we might rely on raw counts of tweets without normalizing, but a quick glance at the number of tweets captured per day (shown on the following page) demonstrates that it is highly variable, growing from about 1 million to



about 5 million in the course of a few weeks in late July, and declining gradually thereafter. Since the official statistics from Twitter show no such trends[8], this is likely an artifact of the methods used to capture the tweet stream, so we must normalize to compensate for this variability.

Generic Sentiment Classification: Our next approach seeks to measure the sentiment of Twitter users more directly by assigning positive, negative, or neutral labels to tweets about a given topic. We use the subjectivity lexicon contained in the University of Pittsburgh OpinionFinder project, available for free download and use. [9][10] This lexicon lists 6,885 unstemmed words along with their subjective polarities, such that the word "conceited" is strongly negative, "ironic" is weakly negative, "central" is neutral, "trendy" is weakly positive, "illuminating" is strongly positive, and so on. We reasoned that using such a dictionary of sentiment could help distinguish between cases where Twitter users are talking about something to complain or organize against it, and cases where they mention it in the form of praise or advocacy.

The implementation begins with parsing the subjectivity lexicon into a dictionary that maps each word to its subjective polarity. Then, given a tweet, every word that it consists of is looked up in the dictionary to determine its polarity. In this project, we assign 10 points to a strongly positive word, 5 to a weakly positive word, 0 to a neutral word, -5 to a weakly negative word, and -10 to a strongly negative word. Once all the words in that tweet have been scored, we sum up their points, and classify a tweet as positive if it has a positive score, negative if it has a negative score, and neutral if it has a score of zero. By assigning different values to strong and weak subjective sentiment, we can classify a sentence with one weak negative word and one strong positive word as positive, while allowing multiple negative words to override a single positive one. We experimented with using only the strong positive and negative words and ignoring the weak ones, but this resulted in worse performance due to the vast majority of tweets going unclassified, so we consistently used both together.

To process the Twitter data, we first filter using the method outlined in the previous section, choosing tweets with "obama" for presidential approval, "job" or "economy" for economic confidence, and "democrat" or "republican" for generic Congressional ballot. To aggregate data for each day, we count the total number of filtered tweets, tweets classified as positive, and tweets classified as negative. In exactly the same way that the economic confidence and generic Congressional ballot polls are calculated, we subtract the number of negative tweets from positive tweets and normalize over the number overall, resulting in a single value for each day in the range of -1 to 1. These values are then smoothed using the sliding time window method to match whichever poll we are attempting to model.

Language Model Sentiment Classification: The last method we implemented was domain-specific sentiment via language models trained on hand-classified data. We reasoned that we might have even better success modeling sentiment of tweets

specifically about Obama if we generated language models to classify sentiment of a tweet. To do this, we manually classified 3633 tweets into positive, negative, or neutral sentiment. For each sentiment category we trained a language model. Given a test tweet, its most likely sentiment classification is the sentiment whose language model gives the maximum likelihood of the tweet, where the prior $P(sent)$ is given by the ratio of tweets in each sentiment category to total tweets classified:

$$sent = argmax_{sent} P(tweet|sent)\, P(sent)$$

We classified 368 positive, 643 negative, and 2622 neutral tweets, but in the end only used 300 of each so that Negative and Neutral wouldn't be so highly favored: given more training data, these two would always generate higher probability than Positive, resulting in lopsided classification.

For this project we experimented with a variety of language models, but in the end we chose two to test against the polling data: first, a Laplace-smoothed unigram language model, and second, a linearly interpolated bigram language model. Justification for this choice is provided in the Results section below, which details our evaluation process.

As a brief description of how our language models work: All language models count the number of occurrences of tokens - unigrams for single words, bigrams for pairs, and trigrams for triples. In the simplest sense, the probability of a given token can be defined as the count of times the token appeared during training divided by the total number of tokens seen. Since this maximum likelihood model does not generalize well to data not seen in the training set, different language models use different smoothing methods to account for previously unseen tokens. Our Laplace-smoothed unigram language model makes the assumption that all words are seen at least once, and that all unseen words are distributed uniformly. Our implementation of the linearly-interpolated language model assigns the probability of a bigram token to be a linear combination of bigram and unigram language model, both using Good-Turing smoothing, which makes the assumption that the probability of an unknown word is equal to that of any word only seen once in training, and adjusts the probabilities of all other words to sum to one.

Many language models use validation data to optimize certain parameters, such as the mixing coefficient for linearly interpolated models. However, because our training set was already small (only 300 sentences in the case of positive tweets, after taking out 20% on which to test), we did not allocate any for validation and set the parameters manually instead.

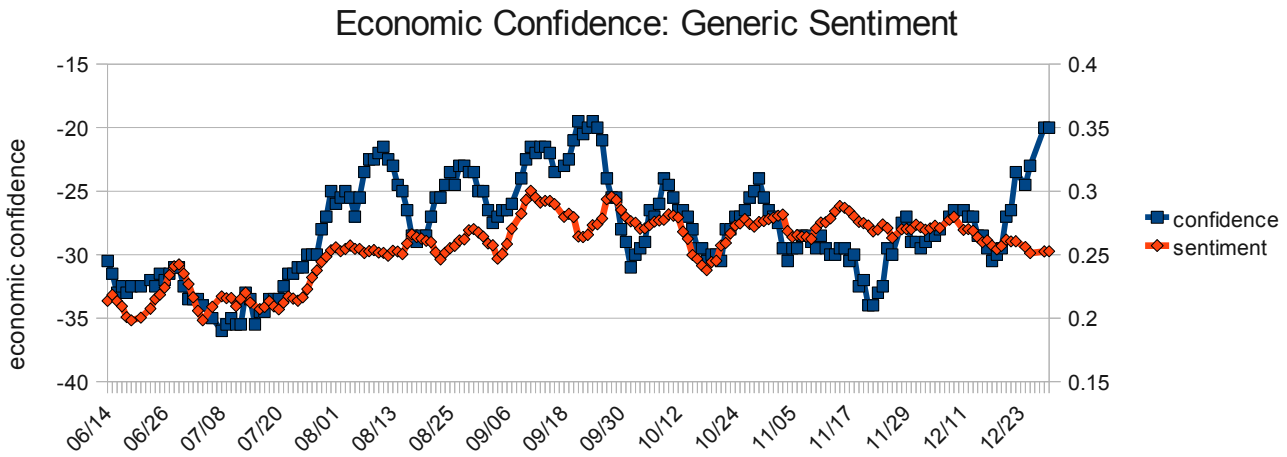
After we train the three language models, we classify the tweets and count the number of positively and negatively classified tweets for each day. Exactly the same as in the generic sentiment classification above, we take all tweets matching a given topic for each day, classify, take the difference between positive and negative counts, and normalize over the total number of tweets. This results in a single number for each day in the range of -1 to 1, which is then smoothed over a time window to match whichever poll we want to emulate.

IV. RESULTS

We measure the success of our methods in terms of correlation with the poll they are meant to emulate. The measure of correlation we use here is, strictly speaking, the Pearson product-moment correlation coefficient usually represented by r, calculated from the two data series X and Y by:

$$r = \frac{\Sigma_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\Sigma_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\Sigma_{i=1}^{n}(Y_i - \bar{Y})^2}}$$

A correlation of 1 implies that there is some linear function such that every data point in one series can be converted exactly into the corresponding data point in the other series, and a correlation of 0 implies that they have no linear relationship with one another. As a gold-standard baseline for two series measuring the same variable, we found 88.8% correlation between Gallup's and Rasmussen's presidential approval poll - of course we did not expect to have such high correlation from Twitter data to polls as between two equivalent polls, but it



sets a reasonable upper bound on expectations. Though correlation implies the existence of an optimal linear transformation of one data set into another, such that we could directly map Twitter-based scores to poll-equivalent data, we are not interested so much in this mapping as the comparison of the methods needed to produce well-correlated results.

Economic Confidence Index: The volume-based metric was not very effective for economic confidence; the two data sets actually had a moderate negative correlation at -36.3% for a 6-day smoothing window and -32.7% for a 15-day window. This made some sense, since people in economic trouble are more likely to be talking about looking for a job or commiserating about the state of the economy than those who have nothing to worry about. Looking at the data we filtered, we found that many sentences were not directly related to the economy or people's jobs, instead using idiomatic phrases like "Good job!" or "an inside job". However, rather than hand-picking certain phrases to exclude, we left the word-based filter as it was, reasoning that this more general approach would give results more applicable to other topics, and noting that most tweets were still germane.

When we applied the generic lexicon-based subjective polarity classification metric, the resulting correlation was much stronger, at 60.1% for a 6-day smoothing window, which further improved to 70.4% given a broader 15-day window. This result was not unprecedented, similar to the 70% range of correlation for economic confidence seen in O'Connor et al. [11] Our approach used a more nuanced view of sentiment analysis, distinguishing between strong and weak polarity words, but this does not appear to have given conclusively better performance.

A few representative examples illustrate both the successes and pitfalls of this approach:

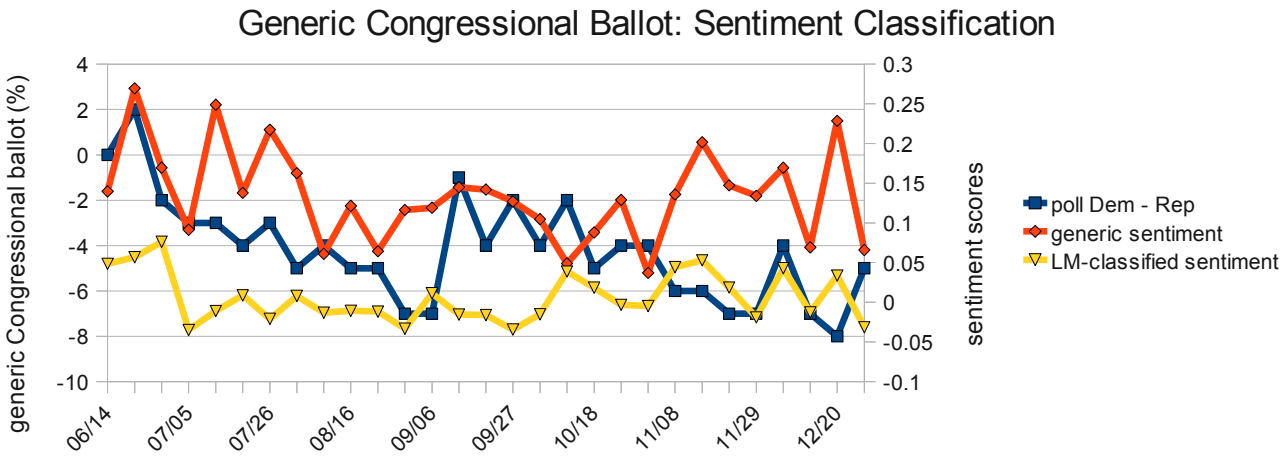
"My wife and I are barely making it now thanks to the collapse of the economy. if they make us pay for insurance we lose our house."
barely -5, collapse -10, lose -10: -25
(correctly classified as negative)

"Economy falls for 4th straight quarter.. http://digg.com/d3zF59" falls +5: +5
(incorrectly classified as positive)

"time to get ready and talk to someone about a job! fun fun" ready +5, fun +10, fun +10: +25
(correctly classified as positive)

"u aint got a job and aint got no mula...."
(incorrectly classified as neutral, since no words in lexicon)

These examples give some clues for future refinements to sentiment analysis applied to Twitter. Most tweets that have clear sentiment are correctly classified by this generic method, but a few fall between the cracks. The first problem is that sentiment in general may not map well to sentiment in a particular subcategory of interest. In the second example, "falls" is clearly a negative word in the

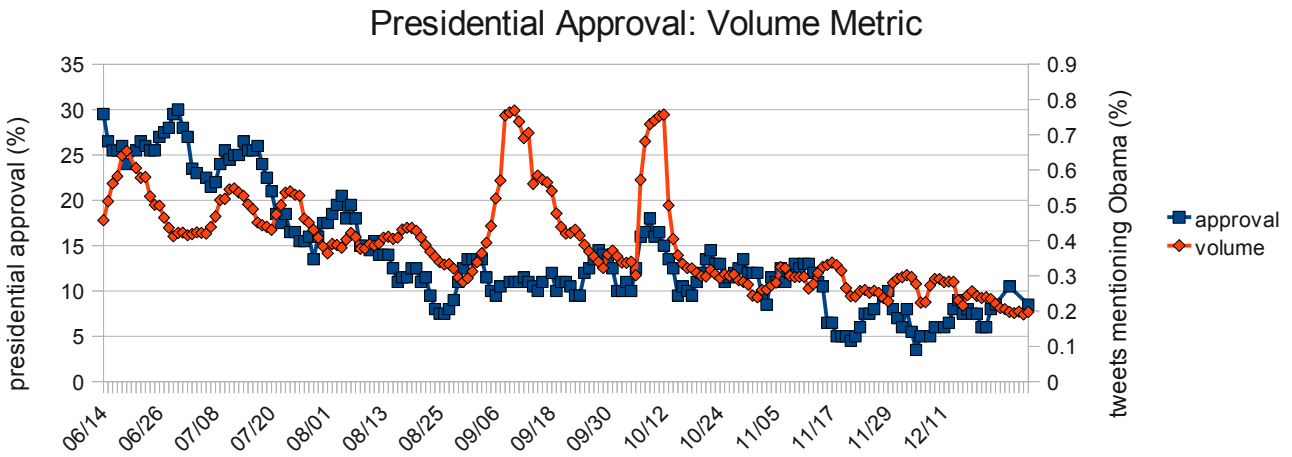


context of the economy, but the polarity lexicon lists it as a weak positive, and thus the sentence is misclassified. The second problem is that language on Twitter often diverges from standard written English. In the final example, we see unorthodox spelling that finds no matches in the lexicon, and is thus classified as neutral by default. To correct for this, one might imagine adding "ain't" and its variations as negative words in some more slang-aware polarity lexicon, or implementing some kind of spell check to search for word matches that users might have mistyped.

Generic Congressional Ballot: We tried various techniques for generic Congressional ballot polling data, but none succeeded in achieving a very high correlation. The first metric, volume, had similar performance to the jobs/economy metric above in that it was moderately negative, at -35.7% (smoothing window was set at 7 days, matching the weekly granularity of the polling data). This indicates that at times when people were talking about Democrats more, they actually preferred Republicans better, and vice versa. This matches the phenomenon seen when measuring economic confidence in that people often post on Twitter to complain or denounce the topic in question more often than to praise it.

Generating generic sentiment scores to compare against the generic Congressional ballot data did not produce nearly as good results as for economic confidence. The correlation did flip from negative to positive, as expected, but remained relatively low at only 21.5%. We tried various other methods, but none gave much stronger correlation: using strong sentiment words only (2.0%), mapping Democratic polling individually with Democratic sentiment (-8.8%), and mapping Republican polling with Republican sentiment (26.8%). Even using the language model sentiment classifier trained for the case of presidential approval only gave 20.9% correlation. This failure may be partly due to the granularity of the polls; we have only 28 data points which are smoothed over their respective time frames but not smoothed relative to one another, unlike the daily opinion polls used in the other two categories examined in this project. Further, compared to the other two polls in our data, there was less variation in the generic Congressional ballot over the time frame we measured, implying that the kind of broad trends seen in the other two might not exist in this sparser data set.

Presidential Approval Rating: The volume-based metric was surprisingly effective at modeling the presidential approval rating over the time frame of our Twitter data. With a smoothing window of 6 days, we have 52.4% correlation, and with 15 days, 61.0% correlation. For Obama, at least, it appears that being in the news correlates with popularity. Why this correlation is so different from the previous two polls is unclear. From our manual tweet-classifying efforts, it is abundantly clear that many users do mention Obama to vilify him or urge action against him, but apparently their tweets are more than counterbalanced by others who either support him or at least gossip about him when he is high in the public consciousness.



The peak in mid-September corresponds to the beginning of Obama's push for health care reform; the peak in mid-October corresponds to his winning the Nobel Peace Prize.

With the promising results from the volume metric in hand, we hoped that applying sentiment analysis might be able to generate even more highly correlated results. Unfortunately, in contrast to the results from the previous two categories, in this case generic sentiment actually produced a weaker, negative correlation: -24.5% for a 6-day window and -38.9% for a 15-day window. If generic sentiment worked effectively, then this result would mean that Obama was more popular when Twitter users expressed more negative sentiment about him and less popular when they were more positive. While this contrarian attitude may exist among Twitter users, our error analysis indicated that it was rather the generic sentiment classification that was failing to properly distinguish sentiment in this domain.

Some representative examples of poor performance on presidential approval:

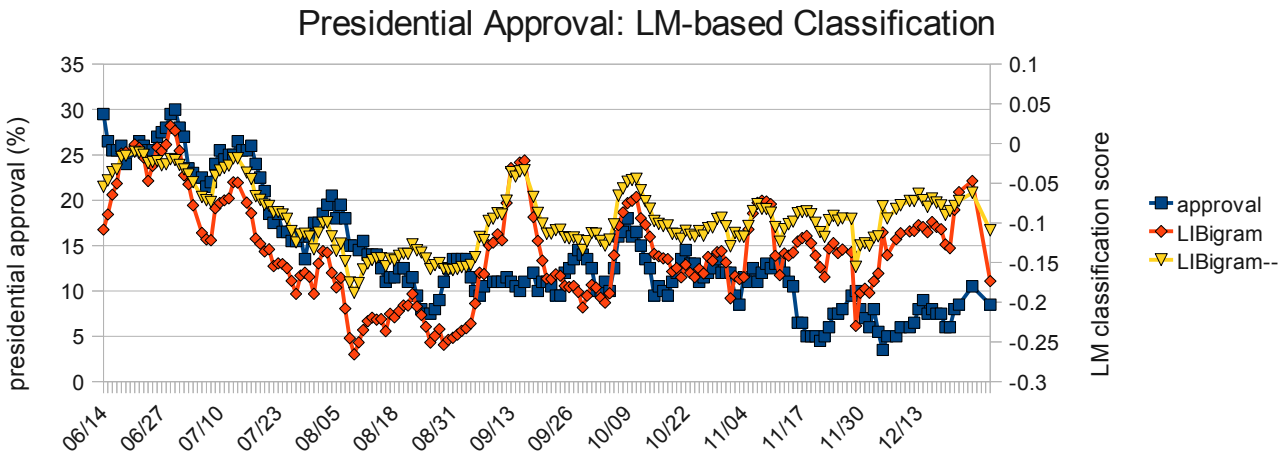
"Obama Thugs now changing terminology of the words used for Health Care so Americans will THINK it's a better plan. They Lie to us" care +10, better +10, lie -10**:** +10 (misclassified as positive)

"Obama calls for 'honest debate' on health care (AP) : AP - President Barack Obama is challenging his critics o.. http://bit.ly/XQbxE" honest +10, debate -5, care +10, challenging -10, critics -10: -5 (misclassified as negative)

"Please do your part! Email the Blue Dogs to defeat Obamacare with this easy form and RT this: http://ow.ly/irmB #tcot" please +10, defeat -10, easy +10: +10 (misclassified as positive)

"The Rick Joyner "ministry" is a fraud and a deception. Joyner is a political hack masquerading as a "minister" to attack Obama health care." fraud -10, deception -10, minister +10, attack -10, care +10: -10 (misclassified as negative)

In cases like the first, it is clear that though "thugs" is not in the lexicon, it represents a strongly negative word. Thus, we might imagine adding more words to the lexicon, perhaps more Internet-specific language. The second example shows an ambiguous tweet, which if anything might be classified as positive, but whose score happens to be slightly negative. We might want to classify such tweets as neutral, since they are not directly expressing an opinion about Obama but instead reporting on his actions. The third exhibits some language specific to the topic; "Obamacare" is a negative term for Obama's health care reform efforts, and "#tcot" (Top Conservatives On Twitter) is a hashtag that identifies the tweet sentiment as conservative - essentially, anti-Obama. The final example shows a tweet which is clearly negative, but is not directly negative about Obama. If we could somehow recognize that the tweet's language usage ("Obama health care" rather than "Obamacare" or some other term) actually implies positive sentiment toward Obama, then we could improve our classification.

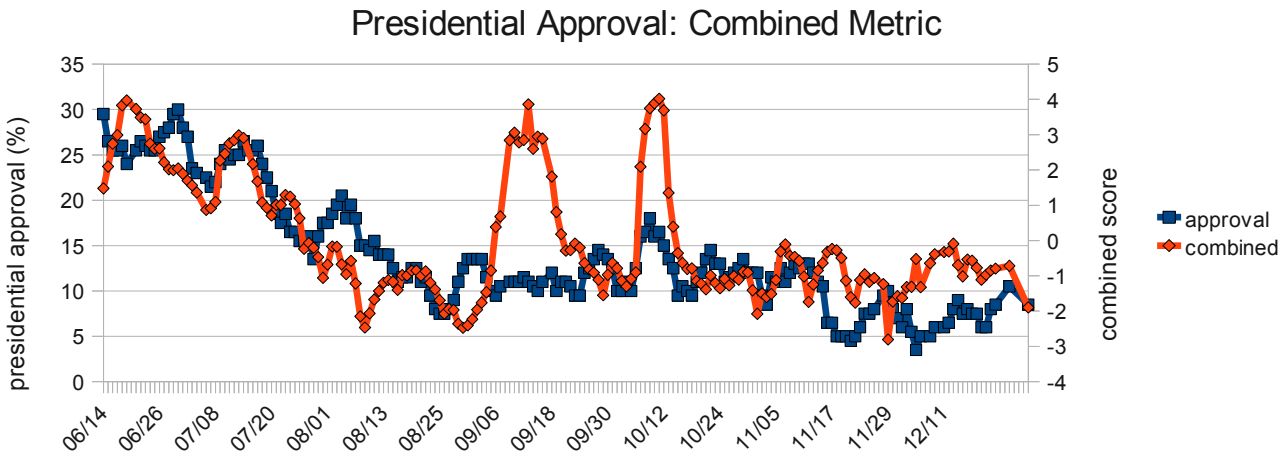


In the hope of solving all four of these issues, we set up language model classification as described in the previous section, training three language models from hand-classified tweets; one for positive, one for negative, and one (as a baseline reference) for neutral tweets. Splitting our labeled data into 80% training and 20% testing, we tested various different categories of language models to see their performance. Since we are focused on classifying positive and negative tweets, we show the precision (P) and recall (R) for both positive and negative categories, and finally the "F1 of F1" metric, simply the harmonic mean of the positive and negative F1 scores. The Laplace-smoothed unigram language model and linearly interpolated bigram language model performed the best, so we chose them to generate sentiment counts to measure against the polling data.

	Pos P	Pos R	Neg P	Neg R	F1▪F1
Generic Sentiment	0.157	0.652	0.226	0.315	0.258
Empirical Unigram	0.246	0.389	0.377	0.409	0.341
Laplace Unigram	0.224	0.598	0.368	0.441	0.360
Good-Tur. Unigram	0.059	0.181	0.235	0.465	0.139
Lin.Interp. Bigram	0.273	0.486	0.284	0.512	0.357
Lin.Interp. Trigram	0.231	0.431	0.305	0.512	0.337

A brief examination of the features extracted by the language models shows that most are closely related to the topic of Barack Obama's approval rating and the topics in the news over the 7 months in our data set. This was expected, since we only classified Obama-based data. For example, in terms of unigrams, the words "barackobama" and "president" were both approximately 10 times more likely in the positive language model than the negative one. Since "barackobama" is Obama's Twitter user name, it is understandable that statements of support are more likely to mention it, and using the title "president" is more indicative of respect than not. Alternately, the words "tcot" and "obamacare" (both discussed previously as negative markers) proved 8 and 6 times more likely in the negative model than the positive one, respectively. For bigrams, "president obama", "insurance reform", and "pres obama" all had higher probability in the positive language model, while "glenn beck", "nobel peace", and "hey obama" all had higher probability in the negative language model.

For the purposes of quick comparison, we focused on the 6-day smoothing window in our intermediate testing. The Laplace-smoothed unigram language model scores gave only 43.3% correlation with the polls, while the linearly interpolated bigram model (LIBigram) scores gave a higher correlation of 49.4%. This is understandable since the LIBigram model contains more information about word order than unigram; though the poor results from our trigram model show the peril of relying too much on higher n-grams, as the small amount of training data can lead to overfitting.



For further refinements to generate a single combined metric to emulate the presidential approval data, we re-ran the LIBigram model with all duplicate tweets and re-tweets removed (This was labeled "LIBigram--".) This was motivated by the large spikes seen in the unmodified LIBigram scores on certain days, when large numbers of Twitter users would re-tweet the same message or post identical tweets; we reasoned that removing these spikes would improve correlation. This gave the best performance yet, with 55.9% correlation on the 6-day window and 54.6% correlation on the 15-day window. Finally, to try to combine our best two metrics, volume and LIBigram--, we standardized both via the equation XYZ and took the sum. This final combined metric had 63.3% correlation on the 6-day window and 69.6% correlation on the 15-day window - our highest.

V. CONCLUSION AND FUTURE WORK

In this work, we have implemented a volume-based metric, a generic sentiment classification metric, and language-model based classification metric, all for predicting the movement of public opinion as measured by opinion polls such as presidential approval rating, economic confidence, and the generic Congressional ballot. Our generic sentiment classification model accurately replicated work by O'Connor *et al.* [12], and our new language-model based method improved on generic sentiment, generating scores for sentiment-based classification competitive with the volume metric. Finally, our combined volume and language-model based approach produced an index that had higher correlation than any other found, at 69.6%.

In terms of future work and improvements: Our approaches used very little data for training, counting analysis with little math, and relatively simple language model. Future improvements to our work should focus on further refining the classification accuracy. This could be done by mining more training data, perhaps using cloudsourcing such as Amazon Mechanical Turk, thus enabling the use of higher n-gram models or those requiring validation data. Further research might use mechanisms analogous to spell checking to correct abbreviations or misspelled words that are typed too quickly. Finally, to better assess the long-term effectiveness of our work, we could gather more Twitter data across a longer period of time.

VI. CITATIONS

[1] Johan Bollen, *et al.* "Twitter mood predicts the stock market." *Journal of Computational Science*, 2011.

[9][11][12] Brendan O'Connor, *et al.* "From Tweets to Polls: Linking Text Sentiment to Public Opinion Time Series." *Proceedings of the International AAAI Conference on Weblogs and Social Media* 2010

[3] Daniel M. Romero, *et al.* "Differences in the Mechanics of Information Diffusion Across Topics: Idioms, Political Hashtags, and Complex Contagion on Twitter." *WWW* 2011

[2] Takeshi Sakaki, et al. "Earthquake Shakes Twitter Users: Real-time Event Detection by Social Sensors." *WWW* 2010

Andranik Tumasjan, *et al.* "Election Forecasts With Twitter: How 140 Characters Reflect the Political Landscape." *Social Science Computer Review* 2010.

[10] Leonid Velikovic, *et al.* "The viability of web-derived polarity lexicons." *NAACL* 2010.

[8] Kevin Weil. "Measuring Tweets." http://blog.twitter.com/2010/02/measuring-tweets.html

DATA SOURCES

[4] SNAP Twitter data: http://snap.stanford.edu/data/twitter7.html

[5] Presidential approval polling data: http://www.gallup.com/poll/113980/Gallup-Daily-Obama-Job-Approval.aspx

[6] Economic confidence polling data: http://www.gallup.com/poll/122840/gallup-daily-economic-indexes.aspx

[7] Generic Congressional ballot polling data: http://www.rasmussenreports.com/public_content/politics/mood_of_america/generic_Congressional_ballot