# Tracking Multiple Social Media for Stock Market Event Prediction

Fang Jin[1], Wei Wang[2], Prithwish Chakraborty[2], Nathan Self[2], Feng Chen[3], and Naren Ramakrishnan[2]

1 Department of Computer Science, Texas Tech University
2 Discovery Analytics Center, Department of Computer Science, Virginia Tech
3 Department of Computer Science, University at Albany, SUNY

**Abstract.** The problem of modeling the continuously changing trends in finance markets and generating real-time, meaningful predictions about significant changes in those markets has drawn considerable interest from economists and data scientists alike. In addition to traditional market indicators, growth of varied social media has enabled economists to leverage micro- and real-time indicators about factors possibly influencing the market, such as public emotion, anticipations and behaviors. We propose several specific market related features that can be mined from varied sources such as news, Google search volumes and Twitter. We further investigate the correlation between these features and financial market fluctuations. In this paper, we present a Delta Naive Bayes (DNB) approach to generate prediction about financial markets. We present a detailed prospective analysis of prediction accuracy generated from multiple, combined sources with those generated from a single source. We find that multi-source predictions consistently outperform single-source predictions, even though with some limitations.

**Keywords:** Market prediction; Multiple social media; Features combination; Google trends; Twitter burst; News sentiment.

## 1 Introduction

Predictions concerning financial markets are complicated by their inherent volatility. Capturing signals of this volatility and providing proper estimates about 'market flips' is of prime interest to economists. This problem has attracted great interest from researchers in diverse disciplines such as economics, statistics and data science. Consequently this has led to a wide variety of methods aimed at modeling stock markets [11, 16, 17, 20, 23].

In most of the traditional approaches, researchers characterize the stock market by the historical records of prices and try to find signatures that indicate rising or falling prices based on this historical time series. However, such financial time series methods are generally incognizant of human indicators, such as public reaction, and have frequently been found wanting in their accuracy at predicting sudden, large changes in market value [4]. Recently, with the pervasive growth of social media [6, 14] which allow individuals to readily express their



P. Perner (Ed.): ICDM 2017, LNAI 10357, pp. 16–30, 2017.
DOI: 10.1007/978-3-319-62701-4_2

sentiments [21], views and concerns, real-time mining of such factors has become possible. Furthermore, different aspects of public sentiment can now be extracted by analyzing multiple social networks. In this paper, we collect and analyze global search trend data from Google, archived news articles from Bloomberg News and relevant tweets from Twitter. Using unsupervised methods, we extract features from these publicly available data sources. Using these features, we design a set of experiments to investigate the correlations between human behavior and market fluctuations in South American markets. With this analysis, we propose models that predict large changes (events) in market value using the most informative extracted factors. To be specific, given these three data sources at day $d$ and historical stock prices for a market, our proposed models attempt to predict the stock market value of at least day $d + 1$.

The key contributions of this paper are:

- We propose a systematic analysis of Google Search Trends, Bloomberg News and Twitter to gather information about market trends and quantify these social media trends.
- We identify burst features from Twitter and further group burst features into burst events. We also investigate and find the correlations of these burst events with market trends.
- We present Delta Naive Bayes Model to predict finance market fluctuations by fusing multiple social media sources. Though there have been earlier attempts at investigating combinations of sources for finance related applications [12, 19], most of the work has focused on surveying datasets. In that respect, to the best our knowledge, our work is the first not only to compare sources but also to propose predictive models that leverage multiple sources.
- Finally, we present our findings about the underlying cross-correlation among market indicators extracted from multiple social media sources and extensively analyze the information from each data source.

## 2 Related Work

In this section we review a few works related to the fields of financial time series analysis, modeling of financial markets and extraction of features from online data streams.

*Financial Time Series Analysis* Financial time series analysis has been one of the most popular approaches to market modeling. Generalized Autoregressive Conditional Heteroskedasticity (GARCH) models [5] have been widely applied in the financial domain since the 1980s. Clustering algorithms have been applied to redescribe time series [11] and identify temporally correlated stocks [1], methods which we have employed in our data processing. More recently, [17] proposed information dissipation length as a leading indicator to measure global instability.

*Correlation with Social Media* With the recent development and prevalence of big data platforms [25, 24], adopting data mining tasks on online social networks has been shown to produce state-of-the-art results. There have been a number of exploratory analyses correlating social media with stock markets. [16] found a correlation between transaction volumes of top companies and Google search volumes of those companies' names. [12] investigated the correlation of search query volume and the Dow Jones Industrial Average (DJIA) and found that a higher search volume of certain finance terms indicates lower DJIA prices. Further, [15] found that trading strategies based on the volumes of 98 keywords from Google Search Trends outperformed random investment with respect to overall turnover. Recently, [13] study the "co-movements" between stock prices and news articles for stock market prediction. [4] calculated public sentiment from Twitter and found the "calm" sentiment curve has an especially strong correlation with DJIA values. [20] found that the number of connected components in a constrained subgraph within time-constrained graphs has high correlation with traded volume. In this paper, we build on the research that suggests that aggregate search volume and sudden changes in sentiment across social networks are correlated with financial market performance by combining these factors in a unified prediction framework.

*Feature Fusion* With respect to fusion methods, [10] proposed the popular Kalman Filter approach for linear filtering and prediction problems which measures multiple sequential sensors to estimate a system's dynamic states. The naive Bayes classifier has been recognized as an effective model which can estimate class labels for multi-dimensional features based on maximum posterior probabilities. Hidden Markov models have been applied successfully for temporal pattern recognition in areas such as sequential images [27]. In our experiment, we employ and revise the classic naive Bayes model, for feature fusion and finance prediction.

## 3 Feature Extraction

The first step in our method is to extract features from social medias. As shown in Figure 1, each data source requires processing to become a time series which we use for prediction. For articles from Bloomberg News, we use topic modeling, natural language processing and sentiment analysis to estimate values for economic indicators. For Google data, we employ Lasso regression to determine which terms from our custom set of finance related terms are the most informative. For Twitter, we chain "burst features" into "burst events" and consider the ones with the greatest hotness degree.

### 3.1 Google Search Trends

We consider Google search volumes a source for surrogate information about public feelings towards certain stock indices and concerns about financial futures. These dynamic trends are potentially important factors which can influence a

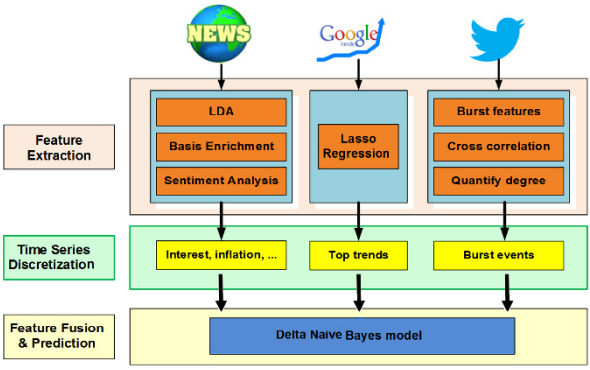



**Fig. 1.** Overall System Framework.

finance market analysis framework. Several authors have used Google search volumes to predict stock value changes [12, 15, 16]. However, since their approaches compute search volumes for keywords and phrases using cumulative returns over a fixed period of time, they are prone to miss the dynamic contribution of keywords over a search cycle as the importance of those words changes. To find the most informative terms for each cycle, we use L1-regularization, or Lasso [22] to filter our dictionary of finance terms. This approach makes our method robust enough to easily incorporate expert feedback by adding keywords as needed.

Formally, for each of our $K$ keywords we can compute its search volume, $x^k(t)$, at time $t$. We propose that this volume influences future stock prices, $P(t+1)$, at time $t+1$. We compare the change in market price, $\Delta P(t+1)$, at time $t+1$ with the change in search volume of the keywords at time $t$, $\Delta x^k(t)$. We express this correlation as a linear equation as shown in Equation 1. $a_k$ is each term's weight.

$$\Delta P(t+1) = \sum_{k=1}^{K} a_k \Delta x^k(t) \tag{1}$$

Weights are computed for each term using Equation 2 where the best keywords are the ones with highest, non-zero weights. We fix $\lambda$ from empirical evaluations based on the accuracy of the final predictions.[1]

$$\begin{aligned} a &= (a_1, a_2, \ldots, a_K) \\ &= argmin_a(\textstyle\sum_t(\Delta P(t) - \sum_{k=1}^{K} a_k \Delta x^k(t))^2 + \lambda \sum_k |a_k|) \end{aligned} \tag{2}$$

Since Google search volumes are reported only weekly, they can mask the general tendency of public queries. We address this problem by considering the $z-score(4)$ of the Google search volumes for each week as the representative value for every day of that week. For market prices, which change every day, we use $z-score(30)$ of each day's closing price as the value for that day. A $z-score$

[1] For the results reported in this paper we use $\lambda = 0.8$.

is defined by:

$$z - score(n) = (X - \mathcal{M})/\Sigma \tag{3}$$

where $X$ is the 1-day difference, $\mathcal{M}$ is the trailing $n$-day moving average of 1-day differences, and $\Sigma$ is the standard deviation of those trailing $n$-day moving 1-day differences. This step aligns the weekly Google Search Trends data with daily stock price data. It also ensures that variances are measured over roughly the same amount of time: about one month. By building functions that connect Google trends with financial market values, we can see how changes in search volume reflect market changes and furthermore select the most informative features for prediction.

### 3.2 Mining Opinion from News Articles

To make these predictions, we are interested in economic features such as interest rates, inflation rates, GDP, consumer confidence, and foreign investment. Unfortunately, official values for these features are not immediately available which makes them unusable for real-time economic forecasting. Instead of relying on values from state-run or research institutions, we employ natural language and statistical processing on economic news stories published online by Bloomberg News and consider the processed content a surrogate for the actual, unavailable data.

We collected 401,923 Bloomberg articles (freely available using their API), from April 2010 to June 2013. To filter this collection to only articles about financial news, we employ the popular Latent Dirichlet Allocation [2] topic modeling algorithm to identify interesting topics. We randomly chose 25,041 articles as a training set, fixed the number of topics at 30 and computed a set of topics. We then manually selected the topics most relevant to finance and economics based on an analysis of the top 30 keywords of each topic. Any document in the corpus composed of any of the selected topics is considered relevant for financial prediction.[2]

Next, we use a suite of natural language processing tools developed by Basis Technology to perform lemmatization, sentence boundary detection, part of speech tagging and noun phrase detection on each article. From there, we use a custom country level dictionary to bin articles by country. This natural language processing allows us to detect passages that refer to a particular country since articles may discuss multiple countries. Finally, we use a customized economic feature dictionary along with noun phrase detection outputs to calculate sentiment scores for each feature for each country. These sentiment scores serve as inputs for our economic predictions.

### 3.3 Twitter Burst Detection

With the explosion in social media networking, Twitter has quickly evolved as an invaluable source for reflecting social movements and exhibiting the complex emotions of individuals. Research [4, 20] shows that financial markets are

[2] A single document can comprise of at most five topics.

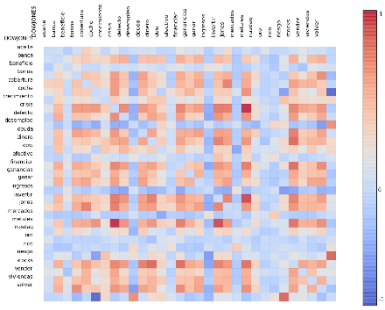


**Fig. 2.** Feature cross correlation from Twitter burst events.

closely tied with social movements and, furthermore, driven by human emotions. Using Twitter, we can detect the most recent dramatic societal movements in real-time which we represent as Twitter bursts [7]. Analyzing the correlation between these bursts and market turbulence, we investigate how they correlate with financial markets. Our entire framework for the Twitter burst detection involves the following steps:

1. Build the network of burst features: For a given window of time, we employ TF-IDF to identify burst features and assign each feature with a burst score.
2. Cluster burst features into burst events: Each pair of nodes that are correlated are connected by an edge with weight equal to their correlation score.
3. Calculate surrogate data: We use Twitter volume and sentiment scoring to measure each burst event's influence.

**Identifying Burst Features** Since we are interested in predictions for country-wide stock markets, we need to determine from which country a tweet was generated. Though tweets can contain user supplied latitudes and longitudes, most tweets do not have this data and must be geolocated by other means. To circumvent this, we applied another geo-enriching algorithm to our dataset by using the geocoding tool described in [18]. Once we have a set of tweets for a country of interest we calculate the 'burst score' for each of its terms over time. Each term's burst score is calculated using 'term frequency-inverse document frequency'(TF-IDF). At given time $t$, when a term's $|z - score(30)|$ is greater than a threshold, it was labeled as a burst feature.

**Grouping into Burst Events** After calculating burst scores for each term, we have a set of burst feature vectors denoted by $B = \{b_0, b_1, \ldots, b_l\}$. We intend to build a graph, $G = (B, E, C)$, such that nodes are connected by an edge in $E$ if there exists a strong enough correlation $C$ between the two nodes [26]. We use Equation 4 to calculate correlation scores where $f^*$ denotes the complex conjugate of $f$ and $m$ is the time lag. The results can be seen in Figure 2.

$$(f * g)[n] = \sum_{m=-\infty}^{\infty} f^*[m]g[n+m] \tag{4}$$

We only consider vectors whose cross-correlation score is greater than a predefined threshold, 0.1 for our purposes, as highly correlated. We create an edge, $e$, between highly correlated vectors with correlation score $c$. Note that because only highly correlated vectors (burst features) are connected by edges, grouping burst features into burst events is transformed into the problem of detecting communities in Twitter networks with each community as a burst event composed of a set of burst features. We employ the Louvain method [3] for community detection in Twitter networks.

**Measure Event *Hotness* Degree** We consider an event that takes place over a short period of time with high Twitter volume to be *hot*. For each event $E$, we take the list of burst features $f_i$ and their associated Twitter volumes and calculate the event's hotness degree. Combining public reaction to this event, we label an event with optimistic and pessimistic quantifiers. Mathematically, for a topic with $m$ burst events, Twitter volume $v_i$ and negative sentiment accounts $s_{ni}$, its hotness degree, $H_e(t)$, at time slot $t$ can be computed according to Equation 5.

$$H_e(t) = \sum_{i=1}^{m} \frac{v_i * s_{ni}}{t} \tag{5}$$

## 4 Ensemble and Prediction

Google search volume is generally considered to be a weather vane of consumer confidence and public expectations about economic situations, which can serve as a background detector for market predictions [15, 16]. Internet articles from companies like Bloomberg News can be a good source for insider opinions, expert views, important event reports and situation analysis, factors containing more information and relatively more professional and accurate signals compared to other social media sources [8]. Finally, Twitter is one of the more popular social media networks for capturing emerging social movements. Especially for breaking news, Twitter moves much faster than traditional media [9]. To make full use of these sources, we construct a feature fusion framework to combine the features extracted from each: news factors ($F_n$), Google search volume factors ($F_g$), and Twitter event factors ($F_t$). We expect to obtain an improved performance by combining these three sources in an intelligent way.

### 4.1 Feature Fusion

Since our observation features come from three different sources, we consider them to be independent of each other. As shown in Figure 3, we denote $Y = \{y_n\}$ as the label of the historical stock price class, where $y_n$ represents the stock price class on day $n$. We also denote the observation features from Google Search Trends, Bloomberg News and Twitter as $F_g$, $F_n$ and $F_t$, respectively. We apply a strategy for combining these features, which can be expressed as $x_n = (F_g, F_n, F_t)_n$, where $x_n$ is the combination of the three feature observed on day $n-1$. In addition, normalization is required before feature fusion since $F_g$, $F_n$ and $F_t$ might be of different scales.

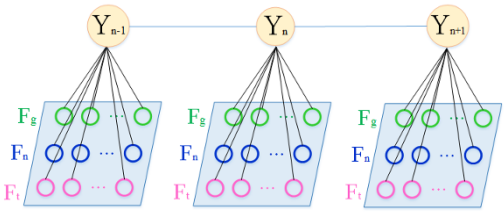



**Fig. 3.** Feature fusion: $Y_n$ is the state of the market. $F_n$ is news factor. $F_t$ is Twitter factor. $F_g$ is Google trends factor.

### 4.2 Stock Discretization

To ensure that periods of high volatility and changes that occur over many weeks are captured accurately, we calculate $z-scores$ over a time window of 30 days. Based on $z-score$ thresholds, we define stock status as following. When $z-score(30) \leq -1$, we call it as **'plunge'** status, when $z-score(30) \geq 1$, we call it as **'surge'** status, otherwise, when $|z-score(30)| < 1$, we call it **'flat'** status.

### 4.3 Delta Naive Bayes Model

Naive Bayes is a well known classifier which, assuming conditional independence of features, computes the posterior probability of labels for an input feature set. Unfortunately, the basic naive Bayes model is suboptimal for financial market predictions because of the highly skewed class distribution. In our training process, we found that 78.6% of our features were assigned to the same class. The result is reasonable considering the stock market values do not often have large, abrupt value changes.

**Algorithm 1** Delta Naive Bayes Model

**INPUTS:** historical dataset $\{(x_{n-L}, y_{n-L}), ..., (x_n, y_n)\}$, training window size $L$, and probabilty threshold $t$, and most recent observation $\{x_{n+1}\}$

**OUTPUTS:** $\hat{y}_{n+1}$

1. Using Naive Bayes Model, compute conditional probability matrix $P(Y|X)$ from $s_0 = \{(x_{n-L}, y_{n-L}), ..., (x_{n-1}, y_{n-1})\}$;
2. Using Naive Bayes Model, compute conditional probability matrix $Q(Y|X)$ from $s_1 = \{(x_{n-L+1}, y_{n-L+1}), ..., (x_n, y_n)\}$;
3. Compute $\Delta P(Y|X)$=abs$(Q(Y|X) - P(Y|X))$;
4. $\bar{p}_{max} = \max_{y_j} \Delta P(Y = y_j|X = x_{n+1})$;
5. Return $\hat{y}_{n+1} = \arg\max_{y_j} \Delta P(Y = y_j|X = x_{n+1})$ when $\bar{p}_{max} > t$; otherwise return $\hat{y}_{n+1}$ as **'flat'**.

To overcome this disadvantage, we propose the Delta Naive Bayes Model in Algorithm 1. Unlike the basic naive Bayes model, this classifier is based on the

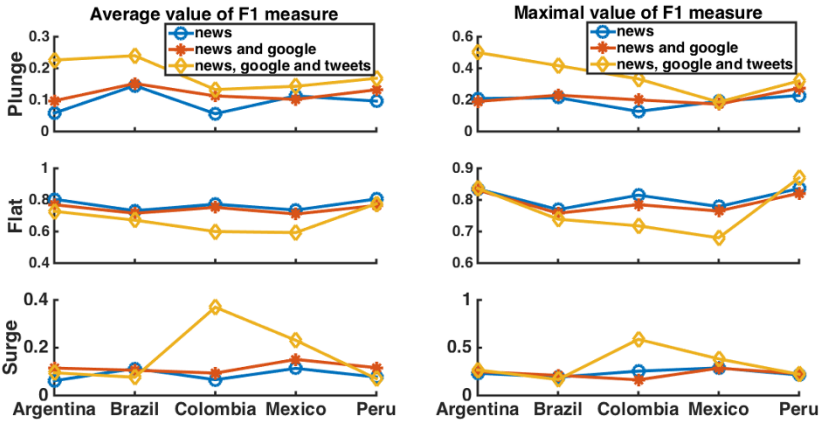



**Fig. 4.** Performance with single news, two sources and three sources respectively on multiple countries. The upper row shows the plunge status prediction performance, with the left one present average $F_1$ measure, and the right one present maximal $F_1$ measure. The middle row shows the flat status performance. The bottom row shows the surge status performance.

incremental change of the posterior probability instead of the posterior probability itself. For the $n^{th}$ time step in the discretized series, we compute $P(Y|X)$ and $Q(Y|X)$ on the historical datasets $s_0 = \{(x_{n-L}, y_{n-L}), \ldots, (x_{n-1}, y_{n-1})\}$ and $s_1 = \{(x_{n-L+1}, y_{n-L+1}), \ldots, (x_n, y_n)\}$, respectively. The difference between $s_0$ and $s_1$ is represented by $\Delta P(Y|X) = Q(Y|X) - P(Y|X)$. For a new observation $x_{n+1}$, we don't use $P(Y|X)$ or $Q(Y|X)$ directly as the classifier. Instead, we first check the maximal value of the conditional probability, $\bar{p}_{max} = \max_{y_j} \Delta P(Y = y_j|X = x_{n+1})$. If $\bar{p}_{max}$ is larger than some predefined threshold $t$, $y_{n+1}$ will be predicted as:

$$\hat{y}_{n+1} = \arg\max_{y_j} \Delta P(Y = y_j|X = x_{n+1}).$$

Otherwise, $y_{n+1}$ will be predicted as **'flat'** state.

As we can see, the only difference between dataset $s_0$ and $s_1$ is that the point $(x_{n-L}, y_{n-L})$ is replaced with the most recent point $(x_n, y_n)$. Hence the difference of the two conditional probability matrices $P(Y|X)$ and $Q(Y|X)$ should be quite small. This helps explain why the basic naive Bayes model nearly always assigns the same class. In addition, the influence of the most recent point $(x_n, y_n)$ on our prediction is attenuated by the other $L-1$ points. It is important to note that using $L$ points helps us avoid the effects of a few noisy data points. The choice of $L$ is also important. Using a low value reduces computation complexity of computing the conditional probability matrix $P(Y|X)$ but risks greater interference from noisy points. On the other hand, using a high value for $L$ leads to better masking of interference but greater computational complexity.

**Table 1.** Delta Naive Bayes model performance on Colombia stock index COLCAP. N denotes source of news, G represents Google trends source and T means Tweets source. $t$ shows transition threshold, $p$ represents precision and $r$ denotes recall.

| Source | t | Plunge | | | Flat | | | Surge | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | $p$ | $r$ | $F_1$ | $p$ | $r$ | $F_1$ | $p$ | $r$ | $F_1$ |
| N | 0.02 | 0.147 | 0.074 | 0.097 | 0.69 | 0.78 | 0.731 | 0.146 | 0.132 | 0.138 |
| | 0.04 | 0.165 | 0.048 | 0.069 | 0.683 | 0.873 | 0.764 | 0.072 | 0.045 | 0.052 |
| | 0.06 | 0.133 | 0.028 | 0.044 | 0.686 | 0.909 | 0.78 | 0.088 | 0.035 | 0.047 |
| | 0.08 | 0.145 | 0.023 | 0.038 | 0.688 | 0.93 | 0.79 | 0.105 | 0.035 | 0.049 |
| | 0.1 | 0.195 | 0.018 | 0.033 | 0.691 | 0.949 | 0.799 | 0.112 | 0.03 | 0.043 |
| NG | 0.02 | 0.156 | 0.097 | 0.118 | 0.697 | 0.742 | 0.718 | 0.142 | 0.153 | 0.147 |
| | 0.04 | 0.178 | 0.097 | 0.124 | 0.702 | 0.817 | 0.754 | 0.138 | 0.094 | 0.11 |
| | 0.06 | 0.17 | 0.083 | 0.109 | 0.692 | 0.839 | 0.757 | 0.104 | 0.063 | 0.075 |
| | 0.08 | 0.186 | 0.083 | 0.112 | 0.688 | 0.856 | 0.762 | 0.117 | 0.054 | 0.07 |
| | 0.1 | 0.19 | 0.071 | 0.1 | 0.688 | 0.881 | 0.772 | 0.149 | 0.049 | 0.068 |
| NGT | 0.02 | 0.135 | 0.169 | 0.149 | 0.582 | 0.499 | 0.535 | 0.325 | 0.418 | 0.366 |
| | 0.04 | 0.134 | 0.146 | 0.139 | 0.593 | 0.594 | 0.588 | 0.379 | 0.418 | 0.394 |
| | 0.06 | 0.128 | 0.124 | 0.123 | 0.583 | 0.63 | 0.6 | 0.405 | 0.418 | 0.407 |
| | 0.08 | 0.15 | 0.124 | 0.134 | 0.587 | 0.681 | 0.627 | 0.374 | 0.358 | 0.36 |
| | 0.1 | 0.175 | 0.091 | 0.118 | 0.584 | 0.74 | 0.65 | 0.355 | 0.302 | 0.322 |

## 5 Empirical Evaluation

Our goal is to predict significant fluctuations in the value of a stock exchange. For this challenge, we focus on South American countries including Argentina, Brazil, Colombia, Mexico, and Peru. We first describe our evaluation criteria then present a detailed performance analysis.

### 5.1 Evaluation Criteria

In the absence of a standard performance matrix, we define an evaluation criteria based on 'hit rate' as follows. For a single day there are only three status: plunge or flat or surge. To account for the accuracy of each status, only when the prediction status is exactly the same, we consider the prediction to be correct.

Recall, precision and $F_1$ measure are widely used metric employed in classification problem. A formal definition of recall is $r = \frac{TP}{TP+FP}$, precision is $p = \frac{TP}{TP+FN}$ and $F_1$ measure is $F_1 = \frac{2rp}{r+p}$, where TP denotes true positive, FP shows false positive, and FN means false negative. While it is fairly straightforward to compute precision and recall for a binary classification problem, it can be quite confusing as to how to compute these values for a multi-class classification problem. In our case, we are 3-class multi classification problem. For an individual class, the false positives are those instances which were classified as that class, but in fact aren't, and the true negatives are those instances which are not that class, and were indeed classified as not belonging to that class (regardless of whether they were correctly classified).

### 5.2 Performance Analysis

Our experiment dataset is from January 1, 2012 to July 31, 2013. We set training window size $L$ vary from 40 to 200 days, to test the efforts in shaping quality distribution. We present an exhaustive evaluation of Delta Naive Bayes model against multiple aspects as follows:

**How do our model perform for the 5 countries of interest and how well does the prediction for plunge, flat and surge status?** We plot the overall prediction performance for each country regard the three stock status in Figure 4. We can see while predict plunge status, Brazil and Argentina achieve prominent performance. When predicting surge status, Colombia is superior than the rest countries, with $F_1$ measure as high as 0.6. While predicting the flat status, Mexico and Peru exhibit outstanding and stable performance.

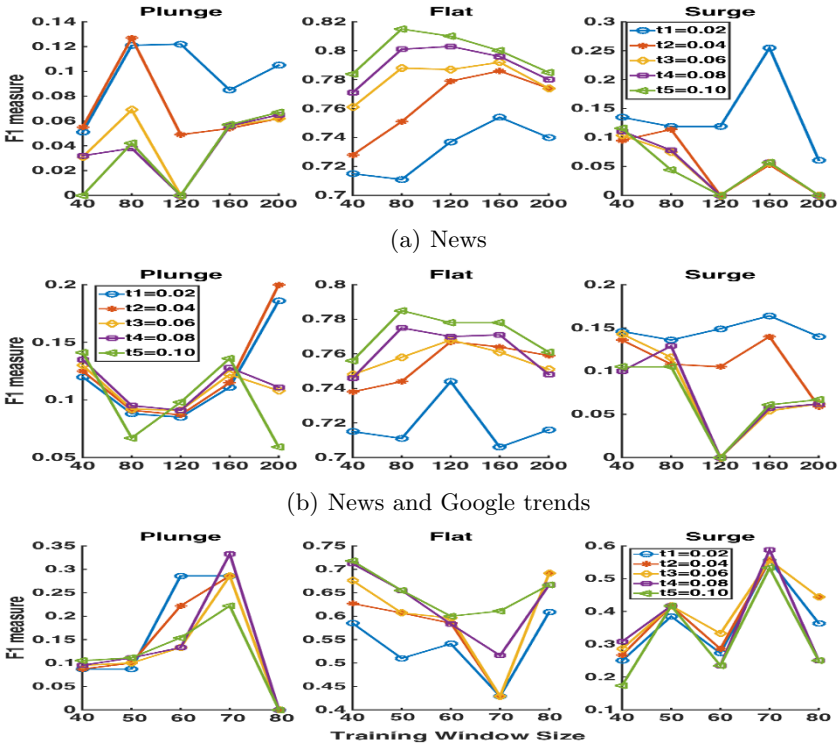
**Single source or multiple sources, which performance is better at capturing surge/plunge status?** As we know, most of the status of stock market is flat status, which makes it even challenging to predict the surge or plunge status. In our experiment, we pay special attention to surge and plunge status prediction, which is shown as upper row and bottom row, respectively in Figure 4. We can see, three sources prediction persistently outperforms two sources prediction, while two sources prediction is greater or equal to single feature prediction moderately. Another clear evidence can be found in Table 1, where shows detailed prediction performance of Colombia, in recall, precision and $F_1$ measure. We can see as for the surge status, the combined three-sources prediction $F_1$ measure can be as high as 0.407, while the two-sources prediction maximal $F_1$ measure is 0.147, and single news prediction maximal score as low as 0.138.

**How does the transition threshold $t$ help shape our quality distribution, in recall and precision?** For Flat state, a larger $t$ often generates a better $F_1$. While Plunge and Surge state, a larger $t$ is often accompanied by a lower $F_1$. So the selection of $t$ is a tradeoff among plunge, surge and flat $F_1 measure$.

**How does training size window $L$ vary with $F_1$ measure scores?** We find that given transition threshold, different training size window lead to different $F_1$ measure depending on fused sources, as can be seen in Figure 5(a). In the case of Colombia, as the three sources prediction, the best performance happens at training window size of 70, both in the plunge status and surge status. However, the relationship are not always consistent across different fusing sources. Thus a general statement about the efficacy of the training window sizes would be hasty since even though the combined sources may show some trends as discussed above, the relationships may be different for a particular example.

### 5.3 Case Study

In this section we further explore the importance of sources by looking at a few specific examples as case studies.




(a) News

(b) News and Google trends

(c) News, Google trends and Tweets

**Fig. 5.** Prediction performance of Colombia, with various threshold and training window size. The X axis denotes training window size and Y axis shows $F_1$ measure. (a)Prediction using single source of news. (b)Prediction using news and Google trends. (3)Prediction using news, Google trends and Tweets.

**Google Search Trends** Google Search Trends of a specific country trends to have a high correlation with country level stock index. We plot four examples of Google Search Trends(GST) terms with country level stock index in Figure 6. We can see the correlation score between GST and stock index can be as high as close to 0.8. Google Search Trends also proved to be a good leading indicator. Figure 7(a) shows the correlation of the tracked Google search trends feature with the $z - score(30)$ of IBOV stock index. As we can see, GST showed an appreciable peak just days before the actual peak on March 27, 2013.

**News Opinion** News opinions were found to be great indicators and for some cases are quite good at detecting sudden changes in the stock market. As can be seen in Figure 7(b), with the help of a series of news articles from May 9, 2012, including "*Brazil Bulls Capitulate as State Intervention Spurs Outflows*," the news mining approach successfully predicted that the IBOV index would be

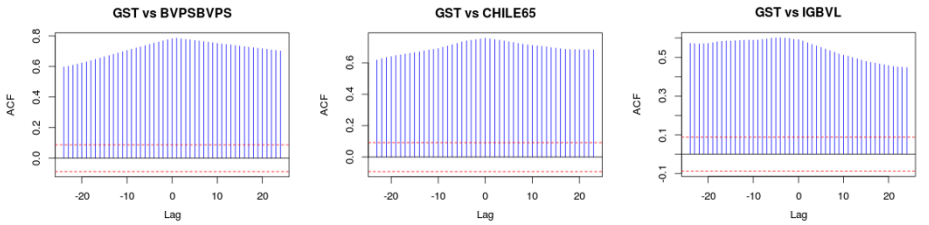
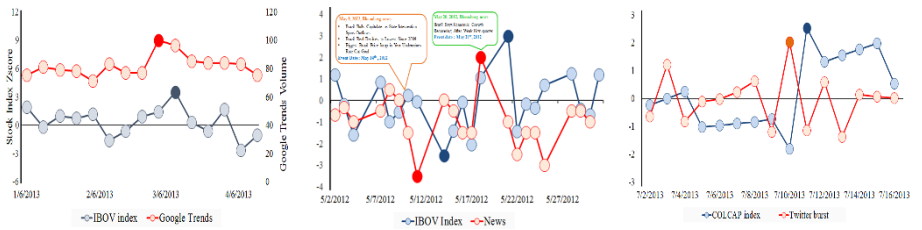



**Fig. 6.** Google Search Trends (GST) correlation with stock indexes. The Y-axis shows the correlation score, and the X-axis shows the lag time (day).




(a) Google Trends (b) News (c) Twitter Burst

**Fig. 7.** Case Study examples. Blue lines represent actual stock market values. Red lines represent predicted values. Dark circles indicate a high-sigma event. (a) shows Google Search Trends volumes for selected terms spiking on March 1, 2013, days before an actual high sigma event in $z - score(30)$ values for Brazil's IBOV index on March 5. (b) shows IBOV values against predicted Bloomberg News values. The boxes list publications from May 9, 2012, that helped generate a predicted downward value event on May 14 and from May 20 that were used in the prediction of the uptick on May 21. (c) shows values for Twitter bursts surging on July 10, 2013, before a high sigma event in $z - score(30)$ values for Colombia's COLCAP index on July 12.

adversely affected. Similarly, the approach was also able to predict an uptick on May 21 based on the news entitled "*Brazil Says Economic Growth Recovering After Weak First Quarter*," published on May 20. The key takeaway from a few such other examples was that our news mining method is very effective for market prediction, especially when high quality news is available.

**Twitter Burst Events** At the same time our case studies indicate that Twitter is also an invaluable source of information. News moves through Twitter much faster than traditional news media. Figure 7(c) shows an example highlighting how Twitter bursts are leading indicators of stock movements and, although noisy, if properly processed can be an invaluable real-time indicator for financial markets. In our experiments, the burst event detection method has been able to capture such extraordinary movements in Twitter chatter and effectively detect bursts of market-influencing events.

## 6 Discussion

In this paper, we propose a novel approach to anticipate financial market fluctuations by combining multiple social media sources. We employ language modeling, topic clustering and sentiment analysis to extract opinions from news articles, Lasso regression on Google search volume data to select the most informative features, and burst detection and event grouping techniques on Twitter data to identify market indicators from Twitter data. Finally, we present the results using Delta Naive Bayes model. In the process, we also demonstrate that the combination of multiple data sources achieves better prediction performance than any of these media sources alone.

## 7 Acknowledgment

Supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center (DoI/NBC) contract number D12PC000337, the US Government is authorized to reproduce and distribute reprints of this work for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBC, or the US Government.

## References

1. Basalto, N., Bellotti, R., De Carlo, F., Facchi, P., Pascazio, S.: Clustering stock market companies via chaotic map synchronization. Physica A: Statistical Mechanics and its Applications 345(1), 196–206 (2005)
2. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. JMLR 3, 993–1022 (2003)
3. Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment 2008(10), P10008 (2008)
4. Bollen, J., Mao, H., Zeng, X.: Twitter mood predicts the stock market. Computational Science 2(1), 1–8 (2011)
5. Bollerslev, T.: Generalized autoregressive conditional heteroskedasticity. Journal of econometrics 31(3), 307–327 (1986)
6. He, W., Guo, L., Shen, J., Akula, V.: Social media-based forecasting: A case study of tweets and stock prices in the financial services industry. Journal of Organizational and End User Computing (JOEUC) 28(2), 74–91 (2016)
7. Jin, F., Khandpur, R.P., Self, N., Dougherty, E., Guo, S., Chen, F., Prakash, B.A., Ramakrishnan, N.: Modeling mass protest adoption in social network communities using geometric brownian motion. In: Proc. KDD'14. pp. 1660–1669. ACM (2014)
8. Jin, F., Self, N., Saraf, P., Butler, P., Wang, W., Ramakrishnan, N.: Forex-foreteller: Currency trend modeling using news articles. In: Proc. KDD'13 Demo Track. pp. 1470–1473. ACM (2013)

9. Jin, F., Wang, W., Zhao, L., Dougherty, E., Cao, Y., Lu, C.T., Ramakrishnan, N.: Misinformation propagation in the age of twitter. Computer 47(12), 90–94 (2014)
10. Kalman, R.: A new approach to linear filtering and prediction problems. Journal of basic Engineering 82(1), 35–45 (1960)
11. Lavrenko, V., Schmill, M., Lawrie, D., Ogilvie, P., Jensen, D., Allan, J.: Mining of concurrent text and time series. In: KDD'00 Workshop. pp. 37–44 (2000)
12. Mao, H., Counts, S., Bollen, J.: Predicting financial markets: Comparing survey, news, twitter and search engine data. Quantitative Finance Papers 1112(1051) (2011)
13. Ming, F., Wong, F., Liu, Z., Chiang, M.: Stock market prediction from wsj: Text mining via sparse matrix factorization. In: Data Mining (ICDM), 2014 IEEE International Conference on. pp. 430–439. IEEE (2014)
14. Piñeiro-Chousa, J., Vizcaíno-González, M., Pérez-Pico, A.M.: Influence of social media over the stock market. Psychology & Marketing 34(1), 101–108 (2017)
15. Preis, T., Moat, H.S., Stanley, H.E.: Quantifying trading behavior in financial markets using Google Trends. Scientific reports 3 (2013)
16. Preis, T., Reith, D., Stanley, H.E.: Complex dynamics of our economic life on different scales: insights from search engine query data. Phil Trans Math Phys Eng Sci 368(1933), 5707–5719 (2010)
17. Quax, R., Kandhai, D., Sloot, P.M.: Information dissipation as an early-warning signal for the Lehman Brothers collapse in financial time series. Scientific reports 3 (2013)
18. Ramakrishnan, N., Butler, P., Muthiah, S., Self, N., Khandpur, R., Saraf, P., Wang, W., Cadena, J., Vullikanti, A., Korkmaz, G., et al.: 'beating the news' with embers: forecasting civil unrest using open source indicators. In: Proc. KDD'14. pp. 1799–1808. ACM (2014)
19. Rao, T., Srivastava, S.: Modeling movements in oil, gold, forex and market indices using search volume index and twitter sentiments. In: Proc. WebSci'13. pp. 336–345 (2013)
20. Ruiz, E.J., Hristidis, V., Castillo, C., Gionis, A., Jaimes, A.: Correlating financial time series with micro-blogging activity. In: Proc. WSDM'12. pp. 513–522 (2012)
21. Sul, H.K., Dennis, A.R., Yuan, L.I.: Trading on twitter: Using social media sentiment to predict stock returns. Decision Sciences (2016)
22. Tibshirani, R.: Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society. Series B (Methodological) pp. 267–288 (1996)
23. Veiga, A., Jiao, P., Walther, A.: Social media, news media and the stock market. News Media and the Stock Market (March 29, 2016) (2016)
24. Wang, J., Yao, Y., Mao, Y., Sheng, B., Mi, N.: Omo: Optimize mapreduce overlap with a good start (reduce) and a good finish (map). In: IPCCC'15
25. Wang, J., Yao, Y., Mao, Y., Sheng, B., Mi, N.: Fresh: Fair and efficient slot configuration and scheduling for hadoop clusters. In: CLOUD'14. pp. 761–768. IEEE (2014)
26. Weng, J., Lee, B.S.: Event Detection in Twitter. In: Proc. ICWSM'11 (2011)
27. Yamato, J., Ohya, J., Ishii, K.: Recognizing human action in time-sequential images using hidden Markov model. In: Proc. CVPR '92. pp. 379–385 (1992)