Correlating Financial Time Series with Micro-Blogging Activity

Eduardo J. Ruiz, Vagelis Hristidis
Department of Computer Science & Engineering
University of California at Riverside
Riverside, California, USA
{eruiz009,vagelis}@cs.ucr.edu

Carlos Castillo, Aristides Gionis,
Alejandro Jaimes
Yahoo! Research Barcelona
Barcelona, Spain
{chato,gionis,ajaimes}@yahoo-inc.com

ABSTRACT

We study the problem of correlating micro-blogging activity with stock-market events, defined as changes in the price and traded volume of stocks. Specifically, we collect messages related to a number of companies, and we search for correlations between stock-market events for those companies and features extracted from the micro-blogging messages. The features we extract can be categorized in two groups. Features in the first group measure the overall activity in the micro-blogging platform, such as number of posts, number of re-posts, and so on. Features in the second group measure properties of an induced *interaction graph*, for instance, the number of connected components, statistics on the degree distribution, and other graph-based properties.

We present detailed experimental results measuring the correlation of the stock market events with these features, using Twitter as a data source. Our results show that the most correlated features are the number of connected components and the number of nodes of the interaction graph. The correlation is stronger with the traded volume than with the price of the stock. However, by using a simulator we show that even relatively small correlations between price and micro-blogging features can be exploited to drive a stock trading strategy that outperforms other baseline strategies.

Categories and Subject Descriptors

H.3.4 [**Information Systems Applications-Systems and Software**]: Information networks; J.4 [**Social and Behavioral Sciences**]: Economics

General Terms

Algorithms, Experimentation

Keywords

Social Networks, Financial Time Series, Micro-Blogging

WSDM'12, February 8–12, 2012, Seattle, Washingtion, USA.
Copyright 2012 ACM 978-1-4503-0747-5/12/02 ...$10.00.

1. INTRODUCTION

As the volume of data from online social networks increases, scientists are trying to find ways to understand and extract knowledge from this data. In this paper we study how the activity in a popular micro-blogging platform (Twitter) is correlated to time series from the financial domain, specifically stock prices and traded volume. We compute a large number of features extracted from postings ("*tweets*") related to certain publicly-traded companies. Our goal is to find out which of these features are more correlated with changes in the stock of the companies.

We start by carefully creating filters to select the relevant tweets for a company. We study various filtering approaches such as using the stock symbol, the company name or variations of the two. We also evaluate the effects of expanding this set of tweets by including closely related tweets.

Next, in order to enrich the feature-extraction mechanism, we represent the tweets during a time interval as an *interaction graph*, an example of which is shown in Figure 1. The nodes in this graph are tweets, users, URLs and hash-tags. The edges express relationships among the nodes, such as authorship, re-tweeting and referencing.

On these graphs, which we call *constrained subgraphs*, we define a large number of features, divided in two groups: activity-based and graph-based features. Activity-based features measure quantities such as the number of hashtags, the number of tweets, and so on. Graph-based features capture the link-structure of the graph. We then study how these features are correlated with the price and traded volume time-series of stock.

Our first key result is that the traded volume for a stock is correlated with the number of connected components in its constrained subgraph, as well as with the number of tweets in the graph. Intuitively, we expect that the traded volume is correlated with the number of tweets. Surprisingly, it is slightly more correlated with the number of connected components. On the other hand, the stock price is not strongly correlated with any of the features we extracted, but it is only slightly correlated with the number of connected components and even less with the number of nodes in the constrained subgraph. We found that other graph-based features, such as PageRank and degree, are effective for larger constrained graphs built around groups of stocks (e.g., financial indexes).

Clearly, finding a correlation with price change has wider implications than finding a correlation with traded volume. Therefore, we test how the slight correlation of the price with the micro-blogging features can be applied to guide a stock trader. Specifically, we create a stock trading simulation, and compare various trading strategies. The second key result of this paper is that by using the Twitter constrained subgraph features of the previous days, we can develop a trading strategy that is successful when compared against several baselines.

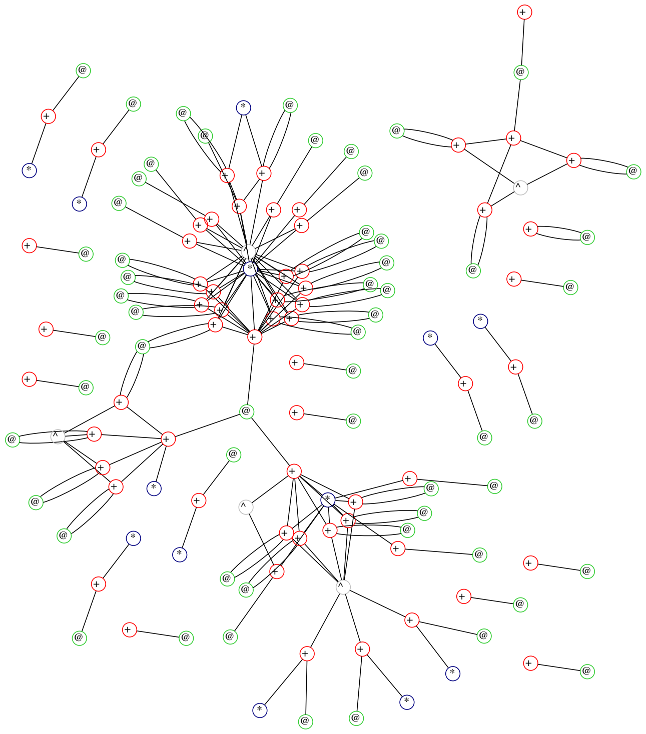


Figure 1: Example of a constrained subgraph for one day and one stock (YHOO). Tweets are presented with red color (+), users are presented with green (@), and URLs with blue (*). Light gray are the similarity nodes (∧)

Our main contributions can be summarized as follows:

- We compare alternative filtering methods to create graphs of postings about a company during a time interval (Section 2). We also present a suite of features that can be computed from these graphs (Section 3).
- We study the correlation of the proposed features with the time series of stock price and traded volume. We also show how these correlations can be stronger or weaker depending on financial indicators of companies, for instance, on their current level of debt (Section 4).
- We perform a study on the application of the correlation patterns found to guide a stock trading strategy. We show that it can lead to a strategy that is competitive when compared to other automatic trading strategies (Section 5).

Roadmap. In Section 2 we discuss the data used in our analysis and the preprocessing steps we performed in order to compute the features. A detailed description of the features we use is given in Section 3. In Section 4 we present correlation results between the proposed features for a company, and the financial time series for its stock, in terms of volume traded or change in price. In Section-5 we discuss how the correlations with price change can be used to develop a trading strategy via simulation. Finally, Section 6 outlines related work, while Section 7 presents our conclusions.

2. DATA PROCESSING

We start our presentation by describing the data used for our analysis, and the processing done in order to compute the features.

2.1 Data acquisition and pre-processing

Stock market data: We obtained stock data from Yahoo! Finance (http://finance.yahoo.com/) for 150 (randomly selected) companies in the S&P 500 index for the first half of 2010. For each stock we recorded the daily closing price and daily traded volume.

Then, we transformed the price series into its daily relative change, i.e., if the series for price is p_i, we used $\frac{p_i - p_{i-1}}{p_{i-1}}$. In the case of traded volume, we normalized by dividing the volume of each day by the mean traded volume observed for that company during the entire half of the year.

Twitter data: We set filters to obtain all the relevant tweets on the first half of 2010. By convention, Twitter in discussions about a stock usually include the stock symbol prefixed by a dollar sign (e.g., $MSFT for Microsoft Corp.). We use a series of regular expressions that find the name of the company, including the company ticker name and hash-tags associated with the company. The expressions were checked manually, looking at the output tweets, to remove those that extracted many spurious tweets. For example, the filter expression for Yahoo is: “#YHOO | $YHOO | #Yahoo”.

To refine this expression we randomly selected 30 tweets from each company, and re-wrote the extraction rules for those sets that had less that 50% of tweets related to the company. To be acceptable, tweets should be related to the company, e.g., mentioning their products or their financial situation. When we determined that a rule-based approach was not feasible, we removed the company from our dataset.

For instance, consider the companies with tickers YHOO, AAPL and APOL, for which the extraction rules had to be rewritten. The short name for *Yahoo* is used in many tweets that are related with the news service provided by the same company (Yahoo! News). In the second case, *Apple* is a common noun and is also used widely for spamming purposes (“Win a free iPad” scams). The last company, *Apollo*, is also the name of a deity in Greek mythology and it appears in many context that are unrelated to the stock.

2.2 Graph representation

We represent each collection of filtered tweets as a graph containing different entities the relationships among these entities.

Figure 2 shows the graph schema, which is also described in Table 1. The nodes in this graph are: the tweets themselves, the users who tweet or who are mentioned in the tweets, and the hash-tags and URLs included in the tweets. The relations in this graph are: re-tweets (between two tweets), authorship (between a tweet and its author), hash-tag presence (between a hash-tag and the tweets that contain it), URL presence, (between a URL and the tweets that contain it), etc.

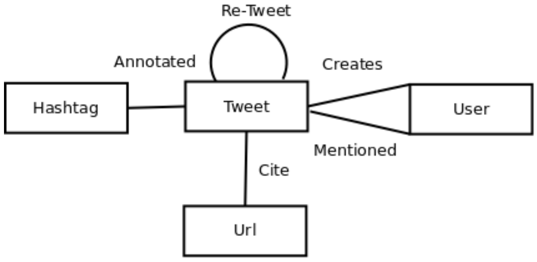


Figure 2: Graph Schema.

Table 1: Schemas.

Nodes	Schema and description
Tweet	(TweetId, Text, Company, Time) A microblog posting
User	(UserId, Name, #Followers, #Friends, Location, Time) A user that posts a tweet or is mentioned
Url	(Url, ExpandedUrl, Time) A URL included in a tweet
Hashtag	(Hashtag, Time) An annotation used in one tweet

Edges	Schema and description
Annotated	(TweetId, Hashtag, Timestamp) Relate a tweet with one hash-tag
Re-tweeted	(RTId, TweetId, Time) Represents the re-tweet action
Mentioned	(TweetId, UserId, Time) A explicit mention of another user
Cited	(TweetId,Url,Time) Connects a URL with tweets including it
Created	(TweetId, UserId, Time) Connects a tweet with its author

HASHTAG	2010-01-28	AAPL	#mkt
TWEET	2010-03-12	AAPL	1XX7XXXXX08
TWEET	2010-03-12	AAPL	1XXX1XXXX11
USER	2010-05-16	AAPL	1XX6XXX83
USER	2010-05-16	AAPL	1XX1XXX2
URL	2010-06-28	AAPL	http://bit.ly/bXXus
URL	2010-06-28	AAPL	http://bit.ly/bXXl3
USRMENTION	2010-06-15	AAPL	@CNNMoney

Figure 3: Example nodes (node type, timestamp, stock symbol, node identity) on the constrained graph of a company.

Additionally, nodes and edges have timestamps at a granularity of one day, corresponding to the granularity of our stock-market time series. Tweets are timestamped with the day they were posted. The rest of the nodes are timestamped with the day they were used for the first time in any tweet (i.e., for a user we set as a timestamp his first tweet). As every edge is incident on a tweet we use the timestamp of the tweet for the edge. For re-tweet edges we use the timestamp of the earliest tweet.

Figure 3 shows sample entries of events extracted for the company *Apple*. Each entry corresponds to a node in the constrained graph. For instance, the first line means the hash-tag #mkt was used on Jan 28 by some tweet related to Apple. The last line states that the Twitter account @CNNMoney was mentioned in some tweet related to Apple on June 15.

We are now ready to define the concept of *data graph.*

Definition [Data Graph] The *data graph* $G = (V,E)$ is a graph whose nodes and edges conform to the schemas in Tables 1.

Some statistics on our data graph are shown in Table 2.

We are interested in subgraphs constrained to a particular time interval and/or a particular company. A constrained subgraph, such as the one depicted in Figure 1, is a subgraph $G^c_{t1,t2}$ of G that only contains nodes with timestamps in time interval $[t1,t2]$, and is about company c. Our definition of constrained subgraph is the following.

Table 2: Data graph statistics for the normal and the expanded graph (which is described in Section 4.4)

	Normal	Expanded
Tweets	176 K	26.8 M
Nodes (Tweets+Users+URLs+Hashtags)	640 K	98.9 M
Edges	493 K	76.7 M
Compressed Size	48MB	1.4GB

Definition [Constrained Subgraph] Let G be a data graph. The *constrained subgraph* $G^c_{t1,t2} = (V,E)$ contains the nodes V of G that are either tweets with timestamps in interval $[t1,t2]$, or non-tweet nodes connected through an edge to the selected tweet nodes. All the edges E in G whose end-nodes are in V are added to $G^c_{t1,t2}$.

2.3 Graph post-processing

Most of the information that we include for each node and edge is straightforward to obtain from the Twitter stream. However, there are some data processing aspects that require special handling:

Mapping user names to IDs: The Twitter stream relates the tweets with internal user identifiers, while user mentions are expressed as user names. To match them, we use the Twitter API to resolve the user-id and user-name reference.

URL shortening: A tweet is constrained to 140 characters, so most URLs are shortened using a URL shortening service such as http://bit.ly/. The problem here is that a single URL can be referred to as several different short URLs. We solve this calling the interface of URL shortening services to get the original URLs.

Re-tweets: In the case of re-tweets, in most cases the original tweet of a re-tweet is referenced (by tweet-id). However, we found many cases where the reference to the original tweet is not present. To resolve those cases, instead of using just explicitly referenced re-tweets we augment the graph adding a new *similarity node* (see Figure 1) that links all similar tweets. We define two tweets to be similar if the Jaccard Distance between the bag of words for both tweets is greater than some value α. We set $\alpha = 0.8$ in our experiments, which is a conservative setting, meaning that tweets having this level of similarity are almost always re-tweets or minor variations of each other.

3. FEATURES

We extract two groups of features from the constrained subgraphs: activity features and graph features. Both are listed in Table 3. **Activity features** simply count the number of nodes of a particular type, such as number of tweets, number of users, number of hash-tags, etc. **Graph features** measure properties of the link structure of the graph. For scalability, feature computation is done using Map-Reduce (http://hadoop.apache.org/).

Feature normalization and seasonality: Most of the feature values are normalized to lie in the interval $[0,1]$. For example, if we consider all the constrained subgraphs within a k-days interval, we can normalize the number of tweets on such a subgraph by dividing by the maximum value across all such subgraphs. The same normalization strategy can be used for users and re-tweets. Other features like number of URLs, hash-tags, etc., are normalized using the number of tweets for the full day.

It is important to consider the effect of seasonality in this graph. The number of tweets is increasing (Twitter's user base grew during our observation period) and has a weekly seasonal effect. We normalize the feature values with a time-dependent normalization

Table 3: Features.

Activity features	Description
RTID	number of re-tweets in $G^c_{t1,t2}$
RTU	number of different users that have re-tweeted in $G^c_{t1,t2}$
TGEO	number of tweets with geo-location in $G^c_{t1,t2}$
TID	number of tweets in $G^c_{t1,t2}$
TUSM	number of tweets that mention any user in $G^c_{t1,t2}$
UFRN	average number of friends for user that posted in $G^c_{t1,t2}$
THTG	number of hash-tags used in all the tweets in $G^c_{t_1,t_2}$
TURL	number of tweets with URLs in $G^c_{t_1,t_2}$
UFLW	average number of followers for user that posted in $G^c_{t_1,t_2}$
UID	number different users that posted a tweet in $G^c_{t_1,t_2}$

Graph features	Description
NUM_NODES	number of nodes of $G^c_{t_1,t_2}$
NUM_EDGES	number of edges of $G^c_{t_1,t_2}$
NUM_CMP	number of connected components of $G^c_{t_1,t_2}$
MAX_DIST	maximum diameter for any component of $G^c_{t_1,t_2}$
PAGERANK	statistics on the page rank distribution for $G^c_{t_1,t_2}$ (AVG, STDV, QUARTILES, SKEWNESS, KURTOSIS)
COMPONENT	statistics on the connected component distribution for $G^c_{t_1,t_2}$ (same as above)
DEGREE	statistics on the node degree distribution for $G^c_{t_1,t_2}$ (same as above)

factor that considers seasonality. This factor is proportional to the total number of messages on each day.

4. TIME SERIES CORRELATION

In this section, we start by looking for correlations between the proposed features for a company, and the financial time series for its stock, in terms of volume traded or change in price. Next, we consider how this correlation changes under (*i*) an analysis isolating different types of companies, (*ii*) an analysis aggregating companies into an index, and (*iii*) changes to the filtering strategy.

4.1 Correlation with volume and price

We use the cross-correlation coefficient (CCF) to estimate how variables are related at different time lags. The CCF value at lag τ between two series X, Y, measures the correlation of the first series with respect to the second series shifted by an amount τ. This can be computed as

$$R(\tau) = \frac{\sum_i((X(i)-\mu_X)(Y(i-\tau)-\mu_Y))}{\sqrt{\sum_i(X(i)-\mu_X)^2}\sqrt{\sum_i(Y(i-\tau)-\mu_Y)^2}}$$

If we find a correlation at a negative lag, this means that the input features could be used to predict the outcome series. Tables 4 and 5 report the average cross-correlation values for traded volume and price respectively, for the 50 companies with most tweets in the observation period, at different lags. We only report the top 5 features for each case, i.e., those having the higher correlation at lag 0. Interestingly, the top features are similar in both lists.

Table 4 shows that the number of components (NUM-CMP) of the constrained sub-graph is the feature that has the best correlation with traded volume. Other good features for this objective are the number of tweets, the number of different users and the total number of nodes on each graph. We also see that there is a positive correlation at lag -1, meaning that these features have some predictive power on the value on the next day. On the other hand, Table 5 shows that the price change is not strongly correlated with any of the proposed features.

Table 4: Average correlation of traded volume and features.

	Lag [days]						
Feature	**-3**	**-2**	**-1**	**0**	**+1**	**+2**	**+3**
NUM-CMP	0.09	0.11	0.21	0.52	0.33	0.16	0.10
TID	0.09	0.10	0.19	0.49	0.31	0.15	0.09
UID	0.09	0.11	0.21	0.49	0.31	0.15	0.10
NUM-NODES	0.09	0.10	0.20	0.49	0.31	0.15	0.09
NUM-EDGES	0.09	0.09	0.18	0.45	0.29	0.14	0.09

Table 5: Average correlation of price and features.

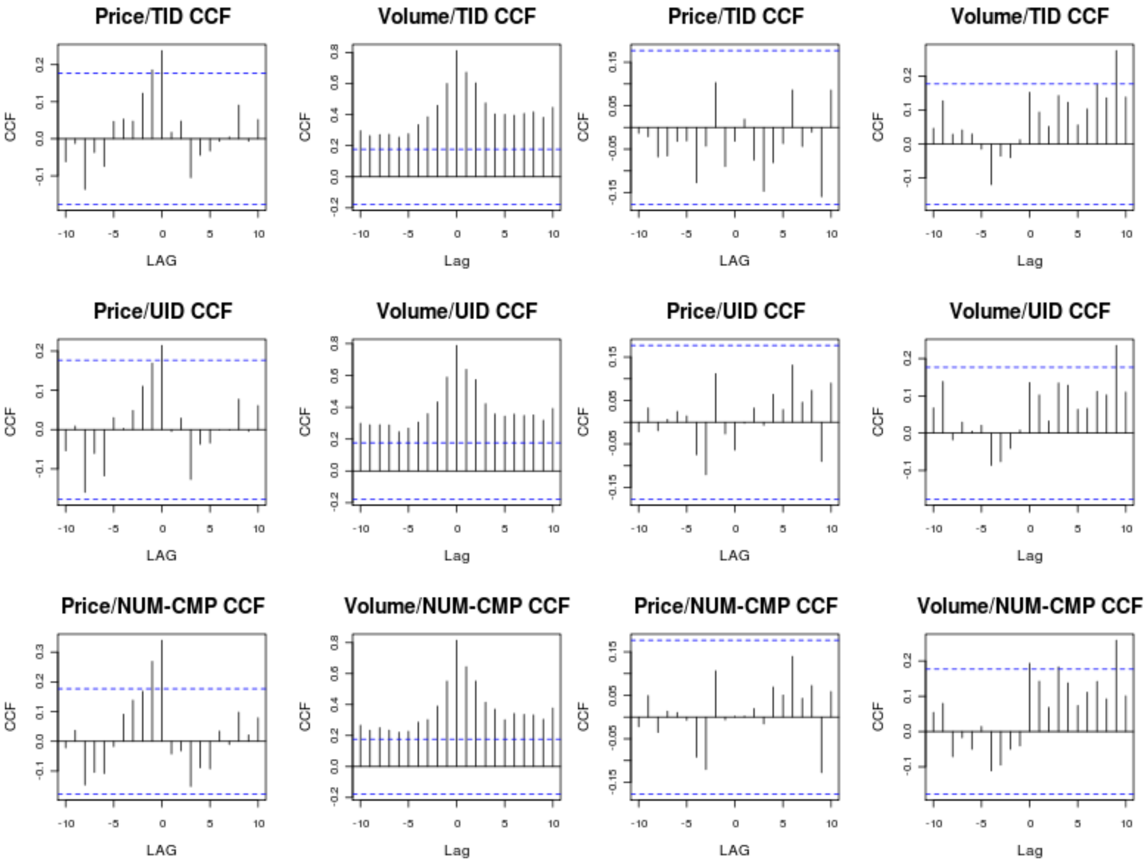
	Lag [days]						
Feature	**-3**	**-2**	**-1**	**0**	**+1**	**+2**	**+3**
NUM-CMP	0.08	0.09	0.10	0.13	0.07	0.07	0.07
NUM-NODES	0.07	0.09	0.10	0.11	0.08	0.07	0.07
TID	0.06	0.08	0.07	0.10	0.07	0.08	0.08
UID	0.07	0.08	0.08	0.10	0.07	0.08	0.07
NUM-EDGES	0.07	0.08	0.09	0.10	0.08	0.07	0.06

4.2 Separating companies by type

Figure 4 shows the cross-correlation coefficient (CCF) values for two selected companies (A.I.G. and Teradyne, Inc.) in our data-set. In Figure 4(a) we see a strong correlation of the stock volume with the four best features of Table 4. On the other hand, Figure 4(b) does not show this correlation.

The next question then is to find out factors that affect the correlation between micro-blogging activity and the companies' stock. We obtained a series of financial indicators for each company from Yahoo! Finance. For each such indicator, we separated the 50 companies in 3 quantiles.

The average correlation between NUM-CMP for each group is shown in Table 6, for the five financial indicators that exhibit the largest variance across their three groups. The "bounds" are the cut-off points of the quantiles. The table shows that the correlation is stronger for companies with low debt, regardless of whether their financial indicators are healthy or not. This could be related to stocks that are expected to surge or that may be candidates for short selling. The users' tweets also correlate better with the stocks for



(a) A.I.G. (AIG)

(b) Teradyne, Inc. (TER)

Figure 4: Correlations for two different companies.

Table 6: Average correlation of traded volumes for different companies according to several financial indicators. Financial indicators are discretized in 3 quantiles (low, medium, high) according to the bounds shown.

	Quantile		
Indicator and bounds	**Low**	**Medium**	**High**
Current Ratio (mrq) bounds: 1.34,2.39,9.41	0.42	0.62	0.52
Gross Profit (ttm) bounds: \$2B,\$9B,\$103B	0.59	0.54	0.42
Enterprise Value/EBITDA (ttm) bounds: 6.22,11.78,20.21	0.54	0.43	0.59
PEG Ratio (5 yr expected) bounds: 1.04,1.51,35.34	0.51	0.44	0.61
Float bounds: \$272MM,\$914MM,\$10B	0.61	0.46	0.48
Beta bounds: 0.98,1.34,3.95	0.47	0.51	0.58

companies having high beta and low float, again suggesting that Twitter activity seems to be better correlated with traded volume for companies whose finances fluctuate a lot.

4.3 Aggregating companies in an index

In Sections 4.1 and 4.2 we built single-stock constrained sub-graphs, which are often too small to reliably compute graph features like PageRank. In this section, we consider a stocks index I consisting of the $n = 20$ biggest (in terms of market capitalization) companies $c_1, \cdots, c_n$ in our dataset, and build index-based constrained sub-graphs.

We can define the index change for each date d as follows:

$$\mathrm{Idx}(I,d) = \sum_{c \in I} \mathrm{priceChange}(c,d) \cdot \mathrm{weight}(c)$$

where $\mathrm{priceChange}(c,d)$ is the difference between the open and close price for c and d, and the weight is the importance (market capitalization) of each company. In particular, as usually done in financial indexes, we define the importance for each company as:

$$\mathrm{weight}(c) = \frac{\mathrm{MarketCap}(c)}{\max_{c' \in I} \mathrm{MarketCap}(c')} .$$

We also define the index trade volume for a particular date as:

$$\mathrm{VolumeIdx}(I,d) = \sum_{c \in I} \mathrm{volumeTraded}(c,d) \cdot \mathrm{weight}(c) .$$

The index data graph considers the tweets that are posted in the first half of 2010. The graph has 108,702 nodes and 209,714 edges. We repeat the correlation experiments of Section 4.1. The results are shown in Tables 7 and 8. The key difference from Tables 4 and 5 is that in the larger index constrained graphs, graph centrality measures like PAGERANK and DEGREE get more reliable estimations and

Table 7: Correlation of traded volume and features, for a synthetic index of top 20 companies.

	Lag [days]						
Feature	**-3**	**-2**	**-1**	**0**	**+1**	**+2**	**+3**
NUM-CMP	0.00	0.12	0.20	0.24	0.15	0.26	0.21
P.RANK-AVG	0.03	0.15	0.20	0.24	0.12	0.16	0.14
TID	-0.01	0.14	0.17	0.23	0.19	0.27	0.21
UID	-0.03	0.11	0.15	0.22	0.20	0.26	0.23
NUM-EDGES	0.02	0.12	0.14	0.22	0.19	0.24	0.20

Table 8: Correlation of price change and features, for a synthetic index of top 20 companies.

	Lag [days]						
Feature	**-3**	**-2**	**-1**	**0**	**+1**	**+2**	**+3**
DEG.-STD	0.08	0.05	0.10	0.12	0.10	0.07	-0.04
DEG.-SKW	0.04	0.02	0.07	0.11	0.06	0.03	0.02
P.RANK-SKW	0.02	0.03	0.08	0.10	0.06	0.02	0.03
DEG.-KURT	0.02	-0.01	0.05	0.10	0.08	0.05	-0.00
P.RANK-STD	0.08	-0.01	0.12	0.09	0.04	-0.03	0.05

are shown to be more strongly correlated to both price and traded volume.

Another interesting observation is that the trading volume is less correlated (Table 7) than in the case of individual stocks (Table 4). We have observed that increases in the activity of some companies is often compensated by the inactivity of others, leading to more stable constrained graphs. In particular, we measured that the variance of the number of connected components, which was the best feature, is lower compared to the average variance for individual companies.

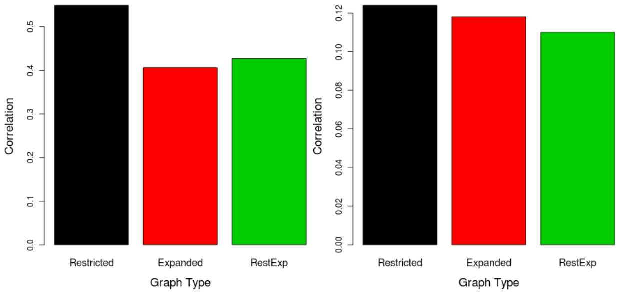
4.4 Modifying the filtering strategy

In Section 2.1 we presented one strategy for filtering the stream. The rationale behind this strategy is that we only focus on tweets related to the financial domain. However, we may end up filtering out some messages that are related to a company but do not mention it explicitly. We study more loose filtering strategies, and obtain negative results: indeed, we degrade the quality of the correlations. We consider the following strategies:

1. **Restricted Graph**: Presented in Section 2.1.
2. **Expanded Graph**: We consider all tweets that: contain the ticker preceded by the $ or # character, or contain the full name of the company, or the short name version after removing common suffixes (e.g., *inc* or *corp*), or the short name as a hash-tag. For instance, for Yahoo! the new expression is: "#YHOO | $YHOO | #Yahoo | Yahoo | Yahoo Inc".
3. **RestExp**: This combines the previous two strategies. We add to the restricted graph the tweets of the expanded graph that are reachable from the nodes of the restricted graph through a path (e.g., through a common author or a re-tweet).

Again we do some visual inspections in a small sample to see if the rules were related with the company. If a rule was very generic (ambiguous) we remove it. For example for the company *APOL* we remove the rules that use *APOLO* as short name. The expanded graph size is shown in Table 2; it has about 150 times more nodes and edges than the normal graph for the same period of time.

The restricted graph is more precise than the expanded graph, but has lower recall. The expanded graph is more noisy, as it may contain many tweets that are related to spam, common conversation (e.g., "I want to eat an apple") or tweets related to non-financial discussions (e.g., complaints about customer service).



(a) Traded volume.

(b) Price change.

Figure 5: Microblogging activity for different strategies, correlated with (a) traded volume (b) price change.

Figure 5 compares the volume and price correlation with the number of components (NUM-CMP), for the different filtering strategies. We observe that the restricted graph strategy has the best correlation, despite its smaller size. Using this expansion strategy of the graph, we add more noise than useful tweets.aaa

5. SIMULATION

In this section we study whether the correlation with price change can be used to develop a trading strategy in the stock market. We simulate daily trading [5, 22, 25] of stocks and try to predict the final price on each day of the simulation. We compare various trading strategies included simple regression models, augmented regression, random and fixed selection.

5.1 Strategies

We model an automated investor who buys and sells stock. The behavior model for this investor is the following:

1. The investor starts with an initial capital endowment C_0.
2. In the morning of every day t, she buys K different stocks using all of the available capital C_t. The investor uses various algorithms to select which stocks to buy and how many shares to buy from each of them. The companies in our simulated stock market are the same random selection from the S&P500 described in Section 2
3. The investor holds the stocks all day long.
4. She sells all the stocks at the closing time of day t. The amount she obtains will be her new capital C_{t+1} and will be used again in step 2. This process finishes on the last day of the simulation.
5. We compare the final capital against the initial investment. We plot the percent of money win or lossed each day against the original investment.

This simple simulation does not consider external effects like deficit of stocks, or the possibility of selling the stocks at the final price. Our aim is to determine if the proposed Twitter features have the potential of improving over other baseline strategies. The stock selection algorithms evaluated are the following:

Random: the investor selects K stocks at random each morning. To diversify the investment the amount of money invested in each stock is C_t/K (uniformly shared).

Fixed: the investor picks K stocks using a particular financial indicator (market capitalization, company size, total debt) and buys

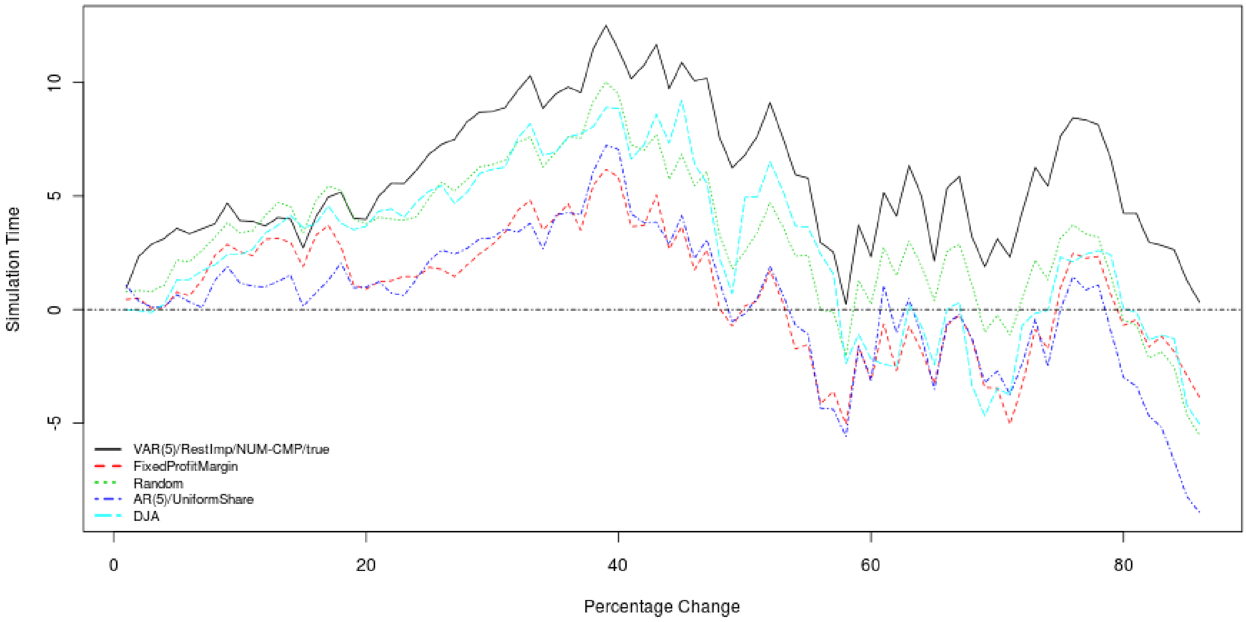


Figure 6: Simulation results for different trading strategies.

from the same companies every day. To diversify the investment the amount of money invested in each stock is C_t/K (uniformly shared).

Auto Regression: the investor buys the K stocks whose price changes will be larger, predicted using an auto-regression ($AR(s)$) model. This model predicts the price x_t of a company at time t using a linear combination of the price in the previous s days:

$$x_t = a_1 x_{t-1} + a_2 x_{t-1} \dots a_m x_{t-m} + c,$$

where each a_i are the parameters of model and c is a constant. Parameters are learned with simple linear regression on a provided training data of L samples.

To diversify the market we have two options: the first one is the uniform split which we already discussed and the second one weighs each stock using the predicted price change. This last strategy is consistent with a very simple heuristic used for the bin packing problem where we prefer those items with high price-weight ratio. In our case:

$$weight = \frac{\text{price difference}}{\text{open price}} .$$

Twitter-Augmented Regression: the investor buys the best K stocks that are predicted using a vector auto-regressive (VAR(s)) model. This model considers, in addition to the price of the previous days, a Twitter feature (e.g., number of components) as observed in the previous days. The model predicts the price of a stock at time t using a linear combination of the price in the previous s days and the values of the augmenting series (Twitter feature) in the same dates:

$$x_t = a_1 x_{t-1} + a_2 x_{t-2} \dots a_m x_{t-m} +$$
$$b_1 y_{t-1} + b_2 y_{t-2} \dots b_m y_{t-m} + c.$$

Training details are similar to the one discussed for the $AR(s)$ model. The strategies to diversify are also the same (uniform and weighted).

5.2 Results

We simulate a series of investments between March 1, 2010 and June 30, 2010. We use the data from January 1, 2010 to February 28, 2010 as training data for the regression models. We keep a window of $L = 35$ training examples and use the previous $s = 5$ days to train. Each model is trained using the Ordinary Least Squares (OLS) method. The initial capital is $C_0 = 10000\$$. The investor buys stock from $K = 10$ different companies every day in every strategy.

For the Twitter-Augmented Regression we use the following features: TID, UID, NUM-CMP, NUM-NODES,NUM-EDGES. We use all the graphs that were described in Section 4.4. We also try the weighed and uniform share options for both the the AR and VAR model.

Figure 6 shows the simulation results for the discussed period. We show the behavior of a sample of all trading techniques discussed: specifically, we show the best approach for each category (e.g., for Twitter-Augmented Regression we only show the behavior of the NUM-CMP feature). Our two baselines for the rest of the discussion are the random strategy and the Auto-Regression strategy. The average loss for the random strategy is -5.52% and the one for the AR models are -8.9% (Uniform) and -13.08% (Weighted). Only one of the fixed models (Profit Margin) have a better behavior than the default random (-3.8%). All the rest of the models that improve the baseline are VAR models. The best one uses the number of components on the RestExp graph with a uniform share for a 0.32% gain. The models obtained with the restricted graph and number of components average a -2.4% loss that is still better that the random model.

Figure 6 also includes the Dow Jones Index Average (DJA) for the same period. As we can see the behavior of all the strategies is consistent with this index's behavior. Our proposed strategy is the only ones that manages to obtain a profit during this period in which the Dow Jones fell −4.2% (Nasdaq (*NDQ*) also drops in a similar −4.7%). The best feature is again the number of components.

6. RELATED WORK

Microblogging data: In recent years several studies (e.g. [17, 19, 16]) have analyzed Twitter data to describe the different types of users, their behavior, the content of the tweets and the way that they are related to trends. In particular, [21] describe the relationship between tweets and trends in traditional news media, as well as query volume on major search engines. Our work is informed by this general knowledge about Twitter, but we focus our attention in a particular domain instead of attempting a general study.

The relations among users, entities and topics in Twitter have been described by a graph and exploited in previous work. For instance, [31] starts by identifying similar users based on their favorite topics and their social connections. Then, a modified version of PageRank is used to find the most influential authors on the Twitter graph. Yamaguchi et al. [32] extend the "Object Rank Algorithm" [1] to consider different types of vertices. These works only consider a fixed point in time and do not consider the changes on the graph structure over time.

News articles and the stock market: The literature relating news stories with financial events is vast. Here we outline some recent works on the subject. Hayo and Kutan [15] present a pure economic prediction model to study the effect of other markets on the Russian market. A variable in this model defines if the news were positive or negative in the past. Although the news classification is manual this study shows the importance of news on the market behavior.

Lavrenko et al. [23] present a model to predict the behavior of the stock of a company using news stories related to the company. The system builds a language model for positive and negative stories and predicts the future behavior checking the language model of the news that appear in the previous hours. Our work does not pretend to be a prediction model: we measure the correlation of the behavior of Twitter with the changes in the stock market. Schumaker and Chen [26] present a system that learns the importance of a news on the performance of a stock. Again, compared with this work we do not try to make a prediction but find a explanation of the change. Moreover, we use microblogging posts instead of news stories. DeChoudhury et al. [7] shows that discussions on blogpost are also correlated with the directions of the stock market.

Yi [33] presents a study to approximate the daily closing value of a stock using data from Twitter. This work also discusses several feature that can be used in the prediction and presents a model that can improve a simple moving average, reducing the error. Sprenger and Welpe [27] and Bollen et al. [3] also show the relations of the stock market or particular stocks with the sentiment of the tweets and how it can be used to improve the prediction. None of this work considers graph features.

Time series regression from Web data: The use of web data for predicting the behavior of a real time series is related with our work. Ginsberg et al. [12] present a method to approximate the cases of influenza of the US using the query log of a search engine. Corley et al. [6] makes a similar prediction using blog content. Other work [8] use search logs to predict the job market.

Hagedorn et al. [13] argue that while these prediction models are correct, they are not really competitive or add any information when compared with models that use domain knowledge. Their application on prediction of music, video games and movie hits shows that other, better-known and simple features are good enough.

Gayo et al. [11] have similar objections, and in addition show evidence of differences between the distribution of demographic characteristics of Twitter users from a country compared to the general population of that country. These differences are substantial and make it impossible to obtain a uniform random sampling of e.g. citizens voting in an ellection. Furthermore, Gayo [10] warns against jumping to conclusions too quickly when analyzing social media data, reminding that "just being large does not make such collections statistically representative of the population as a whole".

Data filtering and spam removal: The filtering of data can have an important effect in the performance of our method. This problem can be divided in two parts: finding related tweets and removing those that are spam.

In [14] the authors propose a supervised multi-classifier that can distinguish if an RSS feed is related with a particular company. We can use this kind of strategy to find better filters. Finin et al. [9] propose crowdsourcing strategies to annotate the entities that appear in the tweets. This knowledge could be used later to train Name Entity Recognizers that can be adapted to the particular twitter characteristics.

Our work could be extended by leveraging features used in the elimination of spam from tweets. Wang [30] and Benevenuto et al. [2] present features that can be used to detect spammers and how they differ from relevant users. Castillo et al. [4] go farther as they study the veracity of tweets for particular events. We think that our work can be improved if we utilize this knowledge to improve the filtering phase.

(Dynamic) graph features: The graph-based features we use are a subset of those present in previous work [18, 24]. We can extend our work by including more graph features. For instance, Kumar et al. [20] present a work on the evolution of social networks in the blogosphere. This work shows that there are changes on the structure that are related with changes in the real world. Other works [29, 28] propose algorithms to mine patterns on massive data graphs. In particular [28] shows how the structure of the graph changes over time.

7. CONCLUSIONS

We presented a framework to extract messages from Twitter about company stocks, and represent that information through graphs capturing different aspects of the conversation around those stocks.

We then used these time-constrained graphs to evaluate a wide range of features in terms of their degree of correlation to changes in stock price and traded volume. We show that the number of connected components of the constrained subgraph is generally the best feature in terms of correlation, especially in relation to traded volume. Graph centrality features like PageRank and average degree become effective for bigger graphs, which can be obtained for multi-company indexes.

Finally, we used simulation to show that these features are useful in order to improve a trading strategy in the stock market.

8. ACKNOWLEDGMENTS

Eduardo Ruiz and Vagelis Hristidis were supported in part by National Science Foundation grants OISE-0730065, IIS-0811922, IIS-0952347 and CNS-1126619. Carlos Castillo, Aristides Gionis and Alejandro Jaimes were partially supported by the Torres Quevedo Program of the Spanish Ministry of Science and Innovation, co-funded by the European Social Fund, and by the Spanish Centre for the Development of Industrial Technology under the CENIT program, project CEN-20101037, "Social Media"
(http://www.cenitsocialmedia.es/).

9. REFERENCES

[1] A. Balmin, V. Hristidis, and Y. Papakonstantinou. Objectrank: authority-based keyword search in databases. In *Proceedings of the 13th international conference on Very Large Data Bases*, pages 564–575, 2004.

[2] F. Benevenuto, G. Magno, T. Rodrigues, and V. Almeida. Detecting Spammers on Twitter. *Collaboration, Electronic messaging, Anti-Abuse and Spam Conference (CEAS)*, 2010.

[3] J. Bollen, H. Mao, and X.-J. Zeng. Twitter mood predicts the stock market. *Journal of Computational Science*, abs/1010.3003, 2010.

[4] C. Castillo, M. Mendoza, and B. Poblete. Information Credibility on Twitter. In *Proceedings of World Wide Web Conference (WWW)*, 2011.

[5] A.-S. Chen, M. T. Leung, and H. Daouk. Application of neural networks to an emerging financial market: forecasting and trading the taiwan stock index. *Computers & Operations Research*, 30(6):901 – 923, 2003.

[6] C. Corley, A. R. Mikler, K. P. Singh, and D. J. Cook. Monitoring influenza trends through mining social media. In *BIOCOMP*, pages 340–346, 2009.

[7] M. DeChoudhury, H. Sundaram, A. John, and D. D. Seligmann. Can blog communication dynamics be correlated with stock market activity? In *Proceedings of the 20th ACM conference on Hypertext and Hypermedia*, 2008.

[8] M. Ettredge, J. Gerdes, and G. Karuga. Using web-based search data to predict macroeconomic statistics. *Communications of the ACM*, 48:87–92, 2005.

[9] T. Finin, W. Murnane, A. Karandikar, N. Keller, J. Martineau, and M. Dredze. Annotating named entities in twitter data with crowdsourcing. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 80–88, 2010.

[10] D. Gayo-Avello. A warning against converting social media into the next literary digest. *Communications of the ACM*, 2011.

[11] D. Gayo-Avello, P. T. Metaxas, and E. Mustafaraj. Limits of electoral predictions using twitter. In *International AAAI Conference on Weblogs and Social Media (posters)*, 2011.

[12] J. Ginsberg, M. H. Mohebbi, R. S. Patel, L. Brammer, M. S. Smolinski, and L. Brilliant. Detecting influenza epidemics using search engine query data. *Nature*, 457(7232):1012–1014, 2009.

[13] S. Goel, J. M. Hofman, S. Lahaie, D. M. Pennock, and D. J. Watts. Whata can search predict? In *Proceedings of World Wide Web Conference (WWW)*, 2010.

[14] B. A. Hagedorn, M. Ciaramita, and J. Atserias. World knowledge in broad-coverage information filtering. In *Proceedings of the 30th annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 801–802, 2007.

[15] B. Hayo and A. M. Kutan. The impact of news, oil prices, and global market developments on russian financial markets. *The Economics of Transition*, 13(2):373–393, 2005.

[16] B. A. Huberman, D. M. Romero, and F. Wu. Social networks that matter: Twitter under the microscope. *First Monday*, 14(1), 2009.

[17] A. Java, X. Song, T. Finin, and B. Tseng. Why we twitter: understanding microblogging usage and communities. In *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, pages 56–65, 2007.

[18] J. M. Kleinberg, R. Kumar, P. Raghavan, S. Rajagopalan, and A. S. Tomkins. The web as a graph: measurements, models, and methods. In *Proceedings of the 5th annual international conference on Computing and combinatorics*, pages 1–17, 1999.

[19] B. Krishnamurthy, P. Gill, and M. Arlitt. A few chirps about twitter. In *Proceedings of the First Workshop on Online Social Networks*, pages 19–24, 2008.

[20] R. Kumar, J. Novak, P. Raghavan, and A. Tomkins. Structure and evolution of blogspace. *Communications of the ACM*, 47:35–39, December 2004.

[21] H. Kwak, C. Lee, H. Park, and S. Moon. What is twitter, a social network or a news media? In *Proceedings of the 19th international conference on World wide web*, pages 591–600, 2010.

[22] V. Lavrenko, M. Schmill, D. Lawrie, P. Ogilvie, D. Jensen, and J. Allan. Language models for financial news recommendation. In *Proceedings of the 9th international conference on Information and knowledge management*, pages 389–396, 2000.

[23] V. Lavrenko, M. Schmill, D. Lawrie, P. Ogilvie, D. Jensen, and J. Allan. Mining of concurrent text and time series. In *Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining Workshop on Text Mining*, pages 37–44, 2000.

[24] M. E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45(2):167–256, 2003.

[25] T. Preis and H. E. Stanley. Trend switching processes in financial markets. In *Econophysics Approaches to Large-Scale Business Data and Financial Crisis*, pages 3–26. 2010.

[26] R. P. Schumaker and H. Chen. Textual analysis of stock market prediction using breaking financial news: The azfin text system. *ACM Transaction Information Systems*, 27:12:1–12:19, 2009.

[27] T. O. Sprenger and I. M. Welpe. Tweets and trades: The information content of stock micrologs. Work in progress in Social Science Research Network.

[28] J. Sun, C. Faloutsos, S. Papadimitriou, and P. S. Yu. Graphscope: parameter-free mining of large time-evolving graphs. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 687–696, 2007.

[29] H. Tong, S. Papadimitriou, J. Sun, P. S. Yu, and C. Faloutsos. Colibri: fast mining of large static and dynamic graphs. In *Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 686–694, 2008.

[30] A. H. Wang. Don't follow me - spam detection in twitter. In *SECRYPT'10*, pages 142–151, 2010.

[31] J. Weng, E.-P. Lim, J. Jiang, and Q. He. Twitterrank: finding topic-sensitive influential twitterers. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining*, pages 261–270, 2010.

[32] Y. Yamaguchi, T. Takahashi, T. Amagasa, and H. Kitagawa. Turank: Twitter user ranking based on user-tweet graph analysis. In *Web Information Systems Engineering-WISE 2010*, pages 240–253. 2010.

[33] A. Yi. Stock market prediction based on public attentions: a social web mining approach. Master's thesis, University of Edinburgh, 2009.