Identifying and Following Expert Investors in Stock Microblogs

[1]Roy Bar-Haim, [1]Elad Dinur, [1,2]Ronen Feldman, [1]Moshe Fresko and [1]Guy Goldstein
[1]Digital Trowel, Airport City, Israel
[2]School of Business Administration, The Hebrew University of Jerusalem, Jerusalem, Israel
{roy, moshe}@digitaltrowel.com, ronen.feldman@huji.ac.il

Abstract

Information published in online stock investment message boards, and more recently in stock microblogs, is considered highly valuable by many investors. Previous work focused on aggregation of sentiment from all users. However, in this work we show that it is beneficial to distinguish expert users from non-experts. We propose a general framework for identifying expert investors, and use it as a basis for several models that predict stock rise from stock microblogging messages (stock tweets). In particular, we present two methods that combine expert identification and per-user unsupervised learning. These methods were shown to achieve relatively high precision in predicting stock rise, and significantly outperform our baseline. In addition, our work provides an in-depth analysis of the content and potential usefulness of stock tweets.

1 Introduction

Online investment message boards such as Yahoo! Finance and Raging Bull allow investors to share trading ideas, advice and opinions on public companies. Recently, stock microblogging services such as StockTwits (which started as a filtering service over the Twitter platform) have become very popular. These forums are considered by many investors as highly valuable sources for making their trading decisions.

This work aims to mine useful investment information from messages published in stock microblogs. We shall henceforth refer to these messages as *stock tweets*. Ultimately, we would like to transform those tweets into buy and sell decisions. Given a set of stock-related messages, this process typically comprises two steps:

1. Classify each message as "bullish" (having a positive outlook on the stock), "bearish" (having a negative outlook on the stock), or neutral.
2. Make trading decisions based on these message classifications.

Previous work on stock investment forums and microblogs usually regarded the first step (message classification) as a sentiment analysis problem, and aligned *bullish* with positive sentiment and *bearish* with negative sentiment. Messages were classified by matching positive and negative terms from sentiment lexicons, learning from a hand-labeled set of messages, or some combination of the two (Das and Chen, 2007; Antweiler and Frank, 2004; Chua et al., 2009; Zhang and Skiena, 2010; Sprenger and Welpe, 2010). Trading decisions were made by aggregating the sentiment for a given stock over all the tweets, and picking stocks with strongest sentiment signal (buying the most bullish stocks and short-selling the most bearish ones).

Sentiment aggregation reflects the opinion of the investors community as a whole, but overlooks the variability in user expertise. Clearly, not all investors are born equal, and if we could tell experts from non-experts, we would reduce the noise in these forums and obtain high-quality signals to follow. This paper presents a framework for identifying experts in stock microblogs by monitoring their performance in a training period. We show that following the experts results in more precise predictions.

Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, pages 1310–1319,
Edinburgh, Scotland, UK, July 27–31, 2011.

Based on the expert identification framework, we experiment with different methods for deriving predictions from stock tweets. While previous work largely aligned bullishness with message sentiment, our in-depth content analysis of stock tweets (to be presented in section 2.2) suggests that this view is too simplistic. To start with, one important difference between bullishness/bearishness and positive/negative sentiment is that while the former represents belief about the future, the latter may also refer to the past or present. For example, a user reporting on making profit from a buying stock yesterday and selling it today is clearly positive about the stock, but does not express any prediction about its future performance. Furthermore, messages that do refer to the future differ considerably in their significance. A tweet reporting on buying a stock by the user conveys a much stronger bullishness signal than a tweet that merely expresses an opinion. Overall, it would seem that judging bullishness is far more elusive than judging sentiment.

We therefore propose and compare two alternative approaches that sidestep the complexities of assessing tweets bullishness. These two approaches can be viewed as representing two extremes. The first approach restricts our attention to the most explicit signals of bullishness and bearishness, namely, tweets that report actual buy and sell transactions performed by the user. In the second approach we learn directly the relation between tweets content and stock prices, following previous work on predicting stock price movement from factual sources such as news articles (Lavrenko et al., 2000; Koppel and Shtrimberg, 2004; Schumaker and Chen, 2010). This approach poses no restrictions on the tweets content and avoids any stipulated tweet classification. However, user-generated messages are largely subjective, and their correlation with the stock prices depends on user's expertise. This introduces much noise into the learning process. We show that by making the learning user-sensitive we can improve the results substantially. Overall, our work illustrates the feasibility of finding expert investors, and the utility of following them.

2 Stock Tweets

2.1 Stock Tweets Language

Stock tweets, as Twitter messages in general, are short textual messages of up to 140 characters. They are distinguished by having one or more references to stock symbols (tickers), prefixed by a dollar sign. For instance, the stock of *Apple, Inc.* is referenced as $AAPL. Two other noteworthy Twitter conventions that are also found in stock tweets are *hashtags*, user-defined labels starting with *'#'*, and references to other users, starting with *'@'*. Table 1 lists some examples of stock tweets.

As common with Twitter messages, stock tweets are typically abbreviated and ungrammatical utterances. The language is informal and includes many slang expressions, many of which are unique to the stock tweets community. Thus, many positive and negative expressions common to stock tweets are not found in standard sentiment lexicons. Their unique language and terminology often make stock tweets hard to understand for an outsider. Many words are abbreviated and appear in several non-standard forms. For example, the word *bought* may also appear as *bot* or *bght*, and *today* may appear as *2day*. Stock tweets also contain many sentiment expressions which may appear in many variations, e.g. *wow, woooow, wooooooooow* and so on. These characteristics make the analysis of stock tweets a particularly challenging task.

2.2 Content Analysis

A preliminary step of this research was an extensive data analysis, aimed to gain better understanding of the major types of content conveyed in stock tweets. First, we developed a taxonomy of tweet categories while reading a few thousands of tweets. Based on this taxonomy we then tagged a sample of 350 tweets to obtain statistics on the frequency of each category. The sample contained only tweets that mention exactly one ticker. The following types of tweets were considered irrelevant:

- Tweets that express question. These tweets were labeled as *Question*.
- Obscure tweets, e.g. *"$AAPL fat"*, tweets that contain insufficient information (e.g. *"http://url.com $AAPL"*) and tweets that seem

		Example	%
Fact	News	$KFRC: Deutsche Bank starts at Buy	14.3%
	Chart Pattern	$C (Citigroup Inc) $3.81 crossed its 2nd Pivot Point Support http://empirasign.com/s/x4c	10.9%
	Trade	bot back some $AXP this morning	12.9%
	Trade Outcome	Sold $CELG at 55.80 for day-trade, +0.90 (+1.6%)X	2.9%
Opinion	Speculation	thinking of hedging my shorts by buying some oil. thinking of buying as much $goog as i can in my IRA. but i need more doing, less thinking.	4.0%
	Chart Prediction	http://chart.ly/wsy5ny $GS - not looking good for this one - breaks this support line on volume will nibble a few short	12.9%
	Recommendation	$WFC if you have to own financials, WFC would be my choice. http://fsc.bz/448 #WORDEN	1.7%
	Sentiment	$ivn is rocking	8.6%
Question		$aapl breaking out but in this mkt should wait till close?	7.1%
Irrelevant		$CLNE follow Mr. Clean $$	24.9%

Table 1: Tweets categories and their relative frequencies

to contain no useful information (e.g *"Even Steve Jobs is wrong sometimes... $AAPL http://ow.ly/1Tw0Z"*). These tweets were labeled *Irrelevant*.

The rest of the tweets were classified into two major categories: *Facts* and *Opinions*.

Facts can be divided into four main subcategories:

1. *News:* such tweets are generally in the form of a tweeted headline describing news or a current event generally drawn from mass media. As such they are reliable but, since the information is available in far greater detail elsewhere, their added value is limited.

2. *Chart Pattern:* technical analysis aims to provide insight into trends and emerging patterns in a stock's price. These tweets describe patterns in the stock's chart without the inclusion of any predicted or projected movement, an important contrast to *Chart Prediction*, which is an opinion tweet described below. Chart pattern tweets, like news, are a condensed form of information already available through more in-depth sources and as such their added value is limited.

3. *Trade:* reports an actual purchase or sale of a stock by the user. We consider this as the most valuable form of tweet.

4. *Trade Outcome:* provides details of an "inverse trade", the secondary trade to exit the initial position along with the outcome of the overall trade (profit/loss). The value of these tweets is debatable since although they provide details of a trade, they generally describe the "exit" transaction. This creates a dilemma for analysts since traders will often exit not because of a perceived change in the stock's potential but as a result of many short-term trading activities. For this reason *trade outcome* provides a moderate insight into a user's position which should be viewed with some degree of caution.

Opinions can also be divided into four main subcategories:

1. *Speculation:* provides individual predictions of future events relating to a company or actions of the company. These are amongst the least reliable categories, as the individual user is typically unable to justify his or her insight into the predicted action.

2. *Chart Prediction:* describes a user's prediction of a future chart movement based on technical analysis of the stock's chart.

3. *Recommendation:* As with analyst recommendations, this category represents users who summarize their understanding and insight into

a stock with a simple and effective recommendation to take a certain course of action with regard to a particular share. *Recommendation* is the less determinate counterpart to *Trade*.

4. *Sentiment:* These tweets express pure sentiment toward the stock, rather than any factual content.

Table 1 shows examples for each of the tweet categories, as well as their relative frequency in the analyzed sample.

3 An Expert Finding Framework

In this section we present a general procedure for finding experts in stock microblogs. Based on this procedure, we will develop in the next sections several models for extracting reliable trading signals from tweets.

We assume that a stock tweet refers to exactly one stock, and therefore there is a one-to-one mapping between tweets and stocks. Other tweets are discarded. We define *expertise* as the ability to predict stock rise with high precision. Thus, a user is an *expert* if a high percentage of his or her *bullish* tweets is followed by a stock rise. In principle, we could analogously follow *bearish* tweets, and see if they are followed by a stock fall. However, bearish tweets are somewhat more difficult to interpret: for example, selling a share may indicate a negative outlook on the stock, but it may also result from other considerations, e.g. following a trading strategy that holds the stock for a fixed period (cf. the discussion on *Trade Outcome* tweets in the previous section).

We now describe a procedure that determines whether a user u is an expert. The procedure receives a training set $\mathcal{T}$ of tweets posted by u, where each tweet is annotated with its posting time. It is also given a classifier $\mathcal{C}$, which classifies each tweet as *bullish* or *not bullish* (either bearish or neutral).

The procedure first applies the classifier $\mathcal{C}$ to identify the bullish tweets in $\mathcal{T}$. It then determines the *correctness* of each bullish tweet. Given a tweet t, we observe the price change of the stock referenced by t over a one day period starting at the next trading day. The exact definition of mapping tweets to stock prices is given in section 5.1. A one-day holding period was chosen as it was found to perform well in previous works on tweet-based trading (Zhang and Skiena, 2010; Sprenger and Welpe, 2010), in particular for long positions (buy transactions). A bullish tweet is considered *correct* if it is followed by a stock rise, and as *incorrect* otherwise[1]. Given a set of tweets, we define its *precision* as the percentage of correct tweets in the set. Let C_u, I_u denote the number of correct and incorrect bullish tweets of user u, respectively. The precision of u's bullish tweets is therefore:

$$P_u = \frac{C_u}{C_u + I_u}$$

Let P_{bl} be the baseline precision. In this work we chose the baseline precision to be the proportion of tweets that are followed by a stock rise in the whole training set (including all the users). This represents the expected precision when picking tweets at random. Clearly, if $P_u \leq P_{bl}$ then u is not an expert. If $P_u > P_{bl}$, we apply the following statistical test to assess whether the difference is statistically significant. First, we compute the expected number of correct and incorrect transactions C_{bl}, I_{bl} according to the baseline:

$$C_{bl} = P_{bl} \times (C_u + I_u)$$

$$I_{bl} = (1 - P_{bl}) \times (C_u + I_u)$$

We then compare the observed counts (C_u, I_u) to the expected counts (C_{bl}, I_{bl}), using Pearson's Chi-square test. Since it is required for this test that C_{bl} and I_{bl} are at least 5, cases that do not meet this requirement are discarded. If the resulting p-value satisfies the required significance level α, then u is considered an expert. In this work we take $\alpha = 0.05$. Note that since the statistical test takes into account the number of observations, it will reject cases where the number of the observations is very small, even if the precision is very high. The output of the procedure is a classification of u as expert/non-expert, as well as the p-value (for experts). The expert finding procedure is summarized in Algorithm 1.

In the next two sections we propose several alternatives for the classifier $\mathcal{C}$.

[1]For about 1% of the tweets the stock price did not change in the next trading day. These tweets are also considered *correct* throughout this work.

Algorithm 1 Determine if a user u is an expert

Input: set of tweets $\mathcal{T}$ posted by u, bullishness classifier $\mathcal{C}$, baseline probability P_{bl}, significance level α

Output: NON-EXPERT/(EXPERT, p-value)

$\mathcal{T}_{bullish} \leftarrow$ tweets in $\mathcal{T}$ classified by $\mathcal{C}$ as *bullish*
$C_u \leftarrow 0$; $I_u \leftarrow 0$
for each $t \in \mathcal{T}_{bullish}$ **do**
  if t is followed by a stock rise **then**
    C_u++
  else
    I_u++
  end if
end for
$P_u = \frac{C_u}{C_u + I_u}$
if $P_u \leq P_{bl}$ **then**
  return NON-EXPERT
else
  $C_{bl} \leftarrow P_{bl} \times (C_u + I_u)$
  $I_{bl} \leftarrow (1 - P_{bl}) \times (C_u + I_u)$
  $p \leftarrow$ *ChiSquareTest*(C_u, I_u, C_{bl}, I_{bl})
  if $p > \alpha$ **then**
    return NON-EXPERT
  else
    return (EXPERT, p)
  end if
end if

4 Following Explicit Transactions

The first approach we attempt for classifying bullish (and bearish) tweets aims to identify only tweets that report buy and sell transactions (that is, tweets in the *Trade* category). According to our data analysis (reported in section 2.2), about 13% of the tweets belong to this category. There are two reasons to focus on these tweets. First, as we already noted, actual transactions are clearly the strongest signal of bulishness/bearishness. Second, the buy and sell actions are usually reported using a closed set of expressions, making these tweets relatively easy to identify. A few examples for buy and sell tweets are shown in Table 2.

While buy and sell transactions can be captured reasonably well by a relatively small set of patterns, the examples in Table 2 show that stock tweets have their unique language for reporting these transactions, which must be investigated in order to come by these patterns. Thus, in order to develop a classifier for these tweets, we created a training and test corpora as follows. Based on our preliminary analysis of several thousand tweets, we composed a vocabulary of keywords which trade tweets must include[2]. This vocabulary contained words such as *in, out, bot, bght, sld* and so on. Filtering out tweets that match none of the keywords removed two thirds of the tweets. Out of the remaining tweets, about 5700 tweets were tagged. The training set contains about 3700 tweets, 700 of which are transactions. The test set contains about 2000 tweets, 350 of which are transactions.

sell	sold sum $OMNI 2.14 +12%
buy	bot $MSPD for earnings testing new indicator as well.
sell	Out 1/2 $RIMM calls @ 1.84 (+0.81)
buy	added to $joez 2.56
buy	I picked up some $X JUL 50 Puts @ 3.20 for gap fill play about an hour ago.
buy	long $BIDU 74.01
buy	$$ Anxiously sitting at the bid on $CWCO @ 11.85 It seems the ask and I are at an impasse. 20 min of this so far. Who will budge? (not me)
buy	In 300 $GOOG @ 471.15.
sell	sold $THOR 41.84 for $400 the FreeFactory is rocking
sell	That was quick stopped out $ICE
sell	Initiated a short position in $NEM.

Table 2: Buy and sell tweets

Since the transaction tweets can be characterized by a closed set of recurring patterns, we developed a classifier that is based on a few dozens of manually composed pattern matching rules, formulated as regular expressions. The classifier works in three stages:

1. *Normalization:* The tweet is transformed into a canonical form. For example, user name

[2]That is, we did not come across any trade tweet that does not include at least one of the keywords in the large sample we analyzed, so we assume that such tweets are negligible.

Dataset	Transaction	P	R	F1
Train	Buy	94.0%	84.0%	0.89
	Sell	96.0%	83.0%	0.89
Test	Buy	85.0%	70.0%	0.77
	Sell	88.5%	79.0%	0.84

Table 3: Results for buy/sell transactition classifier. Precision (P), Recall (R), and F-measure (F1) are reported.

is transformed into USERNAME; ticker name is transformed into TICKER; *buy, buying, bought, bot, bght* are transformed into BUY, and so on.

2. *Matching:* Trying to match one of the buy/sell patterns in the normalized tweet.

3. *Filtering:* Filtering out tweets that match "disqualifying" patterns. The simplest examples are a tweet starting with an "if" or a tweet containing a question mark.

The results of the classifier on the train and test set are summarized in Table 3. The results show that our classifier identifies buy/sell transactions with a good precision and a reasonable recall.

5 Unsupervised Learning from Stock Prices

The drawback of the method presented in the previous section is that it only considers a small part of the available tweets. In this section we propose an alternative method, which considers all the available tweets, and does not require any tagged corpus of tweets. Instead, we use actual stock price movements as our labels.

5.1 Associating Tweets with Stock Prices

We used stock prices to label tweets as follows. Each tweet message has a time stamp (eastern time), indicating when it was published. Our policy is to buy in the opening price of the next trading day (P_B), and sell on the opening price of the following trading day (P_S). Tweets that are posted until 9:25 in the morning (market hours begin at 9:30) are associated with the same day, while those are posted after that time are associated with the next trading date.

5.2 Training

Given the buy and sell prices associated with each tweet, we construct positive and negative training examples as follows: positive examples are tweets where $\frac{P_S - P_B}{P_B} \geq 3\%$, and negative examples are tweets where $\frac{P_S - P_B}{P_B} \leq -3\%$.

We used the SVM-light package (Joachims, 1999), with the following features:

- The existence of the following elements in the message text:
  - Reference to a ticker
  - Reference to a user
  - URL
  - Number
  - Hashtag
  - Question mark
- The case-insensitive words in the message after dropping the above elements.
- The 3, 4, 5 letter prefixes of each word.
- The name of the user who authored the tweet, if it is a frequent user (at least 50 messages in the training data). Otherwise, the user name is taken to be "anonymous".
- Whether the stock price was up or down 1% or more in the previous trading day.
- 2, 3, 4-word expressions which are typical to tweets (that is, their relative frequency in tweets is much higher than in general news text).

6 Empirical Evaluation

In this section we focus on the empirical task of *tweet ranking*: ordering the tweets in the test set according to their likelihood to be followed by a stock rise. This is similar to the common IR task of ranking documents according to their relevance. A perfect ranking would place all the correct tweets before all the incorrect ones.

We present several ranking models that use the expert finding framework and the bullishness classification methods discussed in the previous sections as building blocks. The performance of these models is evaluated on the test set. By considering the

precision at various points along the list of ranked tweets, we can compare the precision-recall tradeoffs achieved by each model.

Before we discuss the ranking models and the empirical results, we describe the datasets used to train and test these models.

6.1 Datasets

Stock tweets were downloaded from the StockTwits website[3], during two periods: from April 25, 2010 to November 1, 2011, and from December 14, 2010 to February 3, 2011. A total of 700K tweets messages were downloaded. Tweets that do not contain exactly one stock ticker (traded in NYSE or NASDAQ) were filtered out. The remaining 340K tweets were divided as follows:

- *Development set:* April 25, 2010 to August 31, 2010: 124K messages
- *Held out set*: September 1, 2010 to November 1, 2010: 110K messages
- *Test set:* December 14, 2010 to February 3, 2011: 106K messages

We consider the union of the development and held out sets as our training set.

6.2 Ranking Models

6.2.1 Joint-All Model

This is our baseline model, as it does not attempt to identify experts. It learns a single SVM model as described in Section 5 from all the tweets in the training set. It then applies the SVM model to each tweet in the test set, and ranks them according to the SVM classification score.

6.2.2 Transaction Model

This model finds expert users in the training set (Algorithm 1), using the buy/sell classifier described in Section 4. Tweets classified as *buy* are considered *bullish*, and the rest are considered non-bullish. Expert users are ranked according to their p value (in ascending order). The same classifier is then applied to the tweets of the expert users in the test set. The tweets classified as bullish are ordered according to the ranking of their author (first all the bullish tweets of the highest-ranked expert user, then all the bullish tweets of the expert ranked second, and so on).

[3] stocktwits.com

6.2.3 Per-User Model

The *joint all* model suffers from the tweets of non-experts twice: at training time, these tweets introduce much noise into the training of the SVM model. At test time, we follow these unreliable tweets along with the more reliable tweets of the experts. The *per-user* model addresses both problems.

This model learns from the development set a separate SVM model $\mathcal{C}_u$ for each user u, based solely on the user's tweets. We then optimize the classification threshold of the learnt SVM model $\mathcal{C}_u$ as follows. Setting the threshold to θ results in a new classifier $\mathcal{C}_{u,\theta}$. Algorithm 1 is applied to u's tweets in the held-out set (denoted $\mathcal{H}_u$), using the classifier $\mathcal{C}_{u,\theta}$. For the ease of presentation, we define *ExpertPValue*($\mathcal{H}_u$, $\mathcal{C}_{u,\theta}$,P_{bl},α) as a function that calls Algorithm 1 with the given parameters, and returns the obtained p-value if u is an expert and 1 otherwise. We search exhaustively for the threshold $\hat{\theta}$ for which this function is minimized (in other words, the threshold that results in the best p-value). The threshold of $\mathcal{C}_u$ is then set to $\hat{\theta}$, and the user's p-value is set to the best p-value found. If u is a non-expert for all of the attempted θ values then u is discarded. Otherwise, u is identified as an expert.

The rest of the process is similar to the transaction model: the tweets of each expert u in the test set are classified using the optimized per-user classifier C_u. The final ranking is obtained by sorting the tweets that were classified as bullish according to the p-value of their author. The per-user ranking procedure is summarized in Algorithm 2.

6.2.4 Joint-Experts Model

The *joint experts* model makes use of the experts identified by the *per-user* model, and builds a single joint SVM model from the tweets of these users. This results in a model that is trained on more examples than in the previous per-user method, but unlike the *joint all* method, it learns only from high-quality users. As with the *joint all* model, test tweets are ranked according to the SVM's score. However, the model considers only the tweets of expert users in the test set.

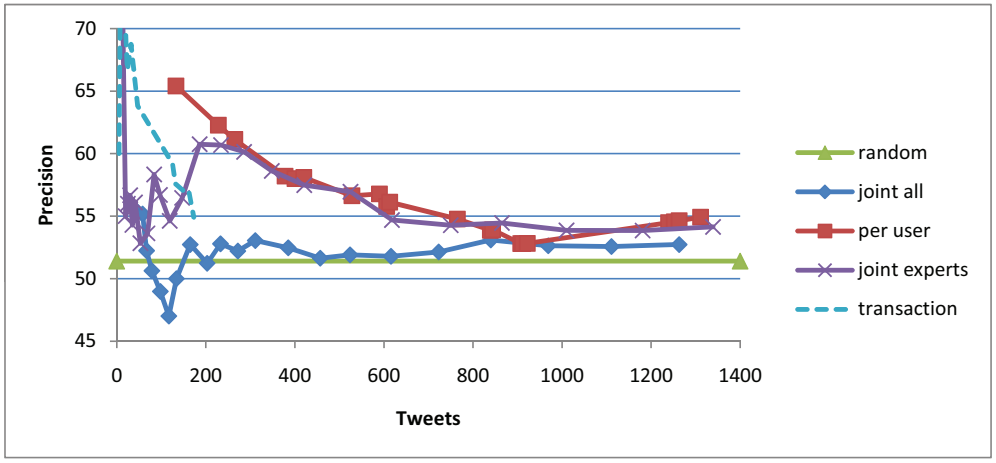

Figure 1: Empirical model comparison

Algorithm 2 Per-user ranking model

Input: dev. set $\mathcal{D}$, held-out set $\mathcal{H}$, test set $\mathcal{S}$, baseline probability P_{bl}, significance level α
Output: A ranked list $\mathcal{R}$ of tweets in $\mathcal{S}$

// Learning from the training set
$E \leftarrow \emptyset$ // set of expert users
for each user u **do**
  $\mathcal{D}_u \leftarrow u$'s tweets in $\mathcal{D}$
  $\mathcal{C}_u \leftarrow$ SVM classifier learnt from $\mathcal{D}_u$
  $\mathcal{H}_u \leftarrow u$'s tweets in $\mathcal{H}$
  $\hat{\theta} = \arg\min_\theta$ *ExpertPValue*($\mathcal{H}_u$, $\mathcal{C}_{u,\theta}$,P_{bl},α)
  $\mathcal{C}_u \leftarrow \mathcal{C}_{u,\hat{\theta}}$
  $p_u \leftarrow$*ExpertPValue*($\mathcal{H}_u$, $\mathcal{C}_{u,\hat{\theta}}$,$P_{bl}$,$\alpha$)
  if $p_u \leq \alpha$ **then**
    add u to E
  end if
end for

// Classifying and ranking the test set
for each user $u \in E$ **do**
  $\mathcal{S}_{bullish,u} \leftarrow u$'s tweets in $\mathcal{S}$ that were classified as *bullish* by $\mathcal{C}_u$
end for
$\mathcal{R} \leftarrow$ tweets in $\bigcup_u \mathcal{S}_{bullish,u}$ sorted by p_u
return $\mathcal{R}$

6.3 Results

Figure 1 summarizes the results obtained for the various models. Each model was used to rank the tweets according to the confidence that they predict a positive stock price movement. Each data point corresponds to the precision obtained for the first k tweets ranked by the model, and the results for varying k values illustrate the precision/recall tradeoff of the model. These data points were obtained as follows:

- For methods that learn a single SVM model (*joint all* and *joint experts*), the graph was obtained by decreasing the threshold of the SVM classifier, at fixed intervals of 0.05. For each threshold value, k is the number of tweets classified as bullish by the model.
- For methods that rank the users by their p value and order the tweets accordingly (*transaction* and *per user*), the i-th data point corresponds to the cumulative precision for the tweets classified as bullish by the first i users. For the *per user* method we show the cumulative results for the first 20 users. For the *transaction* method we show all the users that were identified as experts.

The *random* line is our baseline. It shows the expected results for randomly ordering the tweets in the test set. The expected precision at any point is equal to the percentage of tweets in the test set that were followed by a stock rise, which was found to be 51.4%.

We first consider the *joint all* method, which learns a single model from all the tweets. The only

Correct	Incorrect	P	p
87	46	65.4	0.001
142	86	62.3	0.001
162	103	61.1	0.002
220	158	58.2	0.008
232	168	58.0	0.008
244	176	58.1	0.006
299	229	56.6	0.016
335	255	56.8	0.009
338	268	55.8	0.031
344	269	56.1	0.019
419	346	54.8	0.062
452	387	53.9	0.152
455	389	53.9	0.145
479	428	52.8	0.395
481	430	52.8	0.398
487	435	52.8	0.388
675	564	54.5	0.030
683	569	54.6	0.026
690	573	54.6	0.022
720	591	54.9	0.011

Table 4: Per user model: cumulative results for first 20 users. The table lists the number of correct and incorrect tweets, the precision P and the significance level p.

per-user information available to this model is a feature fed to the SVM classifier, which, as we found, does not contribute to the results. Except for the first 58 tweets, which achieved precision of 55%, the precision quickly dropped to a level of around 52%, which is just a little better than the random baseline. Next, we consider the *transaction* configuration, which is based on detecting *buy* transactions. Only 10 users were found to be experts according to this method, and in the test period these users had a total of 173 tweets. These 173 tweets achieve good precision (57.1% for the first 161 tweets, and 54.9% for the first 173 tweets). However this method resulted in a low number of transactions. This happens because it is able to utilize only a small fraction of the tweets (explicit buy transactions).

Remarkably, *per user* and *joint experts*, the two methods which rely on identifying the experts via unsupervised learning are by far the best methods. Both models seem to have comparable performance, where the results of the *join experts* model are somewhat smoother, as expected. Table 4 shows cumulative results for the first 20 users in the per-user model. The results show that this model achieves good precision for a relatively large number of tweets, and for most of the data points reported in the table the results significantly outperform the baseline (as indicated by the p value). Overall, these results show the effectiveness of our methods for finding experts through unsupervised learning.

7 Related Work

A growing body of work aims at extracting sentiment and opinions from tweets, and exploit this information in a variety of application domains. Davidov et al. (2010) propose utilizing twitter hashtag and smileys to learn enhanced sentiment types. O'Connor et al. (2010) propose a sentiment detector based on Twitter data that may be used as a replacement for public opinion polls. Bollen et al. (2011) measure six different dimensions of public mood from a very large tweet collection, and show that some of these dimensions improve the predication of changes in the Dow Jones Industrial Average (DJIA).

Sentiment analysis of news articles and financial blogs and their application for stock prediction were the subject of several studies in recent years. Some of these works focus on document-level sentiment classification (Devitt and Ahmad, 2007; O'Hare et al., 2009). Other works also aimed at predicting stock movement (Lavrenko et al., 2000; Koppel and Shtrimberg, 2004; Schumaker and Chen, 2010). All these methods rely on predefined sentiment lexicons, manually classified training texts, or their combination. Lavrenko et al. (2000), Koppel and Shtrimberg (2004), and Schumaker and Chen (2010) exploit stock prices for training, and thus save the need in supervised learning.

Previous work on stock message boards include (Das and Chen, 2007; Antweiler and Frank, 2004; Chua et al., 2009). (Sprenger and Welpe, 2010) is, to the best of our knowledge, the first work to address specifically stock microblogs. All these works take a similar approach for classifying message bullishness: they train a classifier (Naïve Bayes, which Das and Chen combined with additional classifiers and a sentiment lexicon, and Chua et al. presented improvement for) on a collection of manually labeled messages (classified into *Buy, Sell, Hold*). Interestingly, Chua et al. made use of an Australian mes-

sage board (HotCopper), where, unlike most of the stock message boards, these labels are added by the message author. Another related work is (Zhang and Skiena, 2010), who apply lexicon-based sentiment analysis to several sources of news and blogs, including tweets. However, their data set does not include stock microblogs, but tweets mentioning the official company name.

Our work differs from previous work on stock messages in two vital aspects. Firstly, these works did not attempt to distinguish between experts and non-expert users, but aggregated the sentiment over all the users when studying the relation between sentiment and the stock market. Secondly, unlike these works, our best-performing methods are completely unsupervised, and require no manually tagged training data or sentiment lexicons.

8 Conclusion

This paper investigated the novel task of finding expert investors in online stock forums. In particular, we focused on stock microblogs. We proposed a framework for finding expert investors, and experimented with several methods for tweet classification using this framework. We found that combining our framework with user-specific unsupervised learning allows us to predict stock price movement with high precision, and the results were shown to be statistically significant. Our results illustrate the importance of distinguishing experts from non-experts. An additional contribution of this work is an in-depth analysis of stock tweets, which sheds light on their content and its potential utility.

In future work we plan to improve the features of the SVM classifier, and further investigate the usefulness of our approach for trading.

References

Werner Antweiler and Murray Z. Frank. 2004. Is all that talk just noise? the information content of internet stock message boards. *Journal of Finance*, 59(3):1259–1294.

Johan Bollen, Huina Mao, and Xiaojun Zeng. 2011. Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1 – 8.

Christopher Chua, Maria Milosavljevic, and James R. Curran. 2009. A sentiment detection engine for internet stock message boards. In *Proceedings of the Australasian Language Technology Association Workshop 2009*.

Sanjiv R. Das and Mike Y. Chen. 2007. Yahoo! for Amazon: Sentiment extraction from small talk on the Web. *Management Science*, 53(9):1375–1388.

Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010. Enhanced sentiment learning using twitter hashtags and smileys. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10. Association for Computational Linguistics.

Ann Devitt and Khurshid Ahmad. 2007. Sentiment polarity identification in financial news: A cohesion-based approach. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 984–991.

T. Joachims. 1999. Making large-scale SVM learning practical. In B. Scholkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*. MIT-Press.

Moshe Koppel and Itai Shtrimberg. 2004. Good news or bad news? Let the market decide. In *Proceedings of the AAAI Spring Symposium on Exploring Attitude and Affect in Text: Theories and Applications*.

Victor Lavrenko, Matt Schmill, Dawn Lawrie, Paul Ogilvie, David Jensen, and James Allan. 2000. Mining of concurrent text and time series. In *Proceedings of the 6th ACM SIGKDD Int'l Conference on Knowledge Discovery and Data Mining Workshop on Text Mining*.

Brendan O'Connor, Ramnath Balasubramanyan, Bryan R. Routledge, and Noah A. Smith. 2010. From tweets to polls: Linking text sentiment to public opinion time series. In *Proceedings of the International AAAI Conference on Weblogs and Social Media*.

Neil O'Hare, Michael Davy, Adam Bermingham, Paul Ferguson, Pvraic Sheridan, Cathal Gurrin, and Alan F Smeaton. 2009. Topic-dependent sentiment analysis of financial blogs. In *TSA'09 - 1st International CIKM Workshop on Topic-Sentiment Analysis for Mass Opinion Measurement*.

Robert P. Schumaker and Hsinchun Chen. 2010. A discrete stock price prediction engine based on financial news. *Computer*, 43:51–56.

Timm O. Sprenger and Isabell M. Welpe. 2010. Tweets and trades: The information content of stock microblogs. Technical report, TUM School of Management, December. working paper.

Wenbin Zhang and Steven Skiena. 2010. Trading strategies to exploit blog and news sentiment. In *ICWSM'10*.