Draft Tech Report
Using News Articles to Predict Stock Price Movements *

Győző Gidófalvi and Charles Elkan
Department of Computer Science and Engineering 0114
University of California, San Diego
La Jolla, California 92037-0114
{*gyozo,elkan*}*@cs.ucsd.edu*

March 26, 2003

Abstract

In many application areas, a constant stream of text documents is available for analysis. How to use documents to predict or recognize automatically threats or opportunities is an open research question. For example, how can email messages streaming in and out of an organization be measured to predict the risk of a network attack? In this paper, we investigate an analogous problem for which training information is available publicly. The problem is to extract indicators from financial news articles that predict short-term changes in stock prices.

Each news article in a training set is placed in one of three classes ("up," "down," or "normal") according to the price change of the associated stock in a time interval surrounding the publication of the article. The three classes are defined relative to the volatility of the stock and the change in a relevant index. A naive Bayesian text classifier is trained to recognize which class each news article belongs to. Given a future article, the trained classifier potentially predicts the price change of the associated stock.

We test the performance of trained classifiers using various evaluation metrics. We find that the highest predictiveness is achieved over the 20 minute period before the publication of each article. In other words, news mostly influences stock prices in the 20 minutes prior to its publication. Some market participants have access to news before it is public. With this access, our trained classifier predicts price changes in a way that can be used for profitable trading. In particular, using a minimal risk trading policy and investing \$1000 at each trade, the average profit per trade is \$1.0063 with a standard error of $\pm$\$0.2979. A bootstrap analysis shows that this profit is statistically significant.

1 Introduction

According to the efficient market hypothesis in financial markets profit opportunities are exploited as soon as they arise, hence stock prices follow a random walk and are impossible to predict [6]. However, as was pointed out in earlier research ([2], [3] and [5]), in the financial market setting the task is rather to generate profitable action signals (buy and sell) than to accurately predict future values of a time series. Technical analysis tries to predict future prices based on past prices, whereas fundamental analysis tries to base predictions on factors in the real economy (e.g.: inflation, changes in the company, demand for products or services of the company). As financial textual data (news articles) became available on the web a new source of indicators appeared, which potentially could contain useful information for fundamental analysis. The objective of this project is to analyze and extract such information, and derive numerical indicators from financial text.

This paper is organized as follows. In section 2 we describe the task at hand, give a high-level system overview, and describe the data set that we use in this project. In this section we also describe some of the challenges we encounter during the gathering of the data, and give some preprocessing steps that we use to overcome these challenges. In section 3 we describe our particular method for deriving numerical

*This material is based on work sponsored by the United States Airforce and supported by the Air Force Research Laboratory under Contract F30602-02-C-0046.

indicators from textual data. In section 4 we describe the experimental design along with some methods for evaluating predictions that have some economic value associated with them. In section 5 we show and discuss the results of the experiments we perform.

2 Task, data and system overview

Indicators can be of two types: those derived from textual data (news articles), and those derived from numerical data (stock prices). We choose to focus our research on indicators derived from textual data. We obtain indicators derived from news articles by learning a naïve Bayesian text classifier for higher-level, relative price movements of stocks. We then use this trained naïve Bayesian classifier to compute the probability for every new, stock-specific news article that the article belongs to a class representing a particular movement type. Using various performance metrics we evaluate these predictions and analyze the naïve Bayesian text classifiers that we learn. This process is depicted in Figure 1 below.

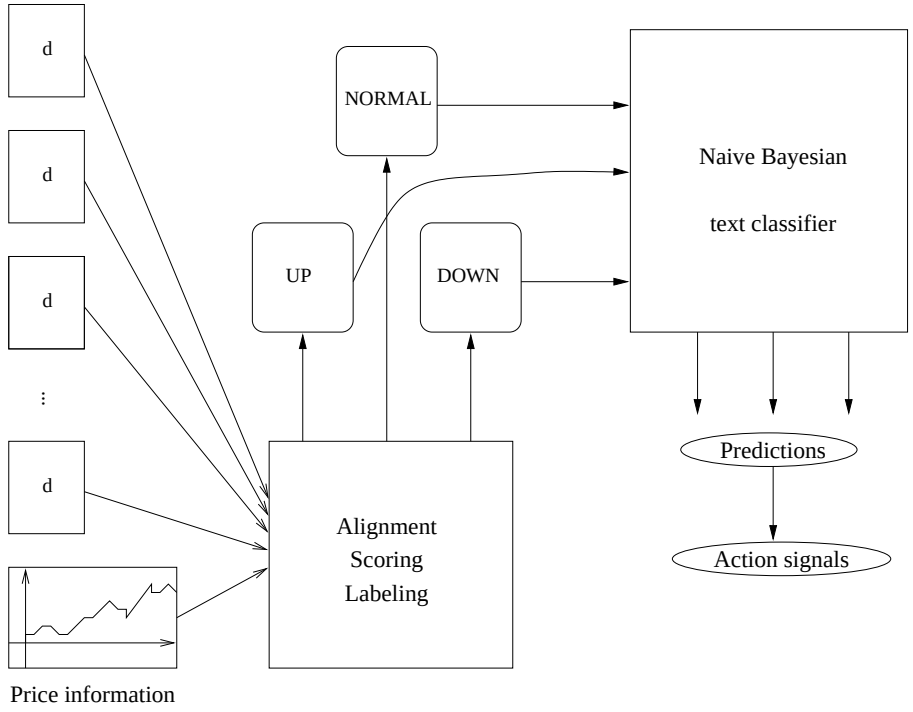

Figure 1: A high-level overview of the system architecture. Stock specific, timestamped news articles, marked as *d*s, are aligned to corresponding price information, and are scored based on the relative performance of the stock during the period of alignment. Thresholding on these scores we obtain 3 classes of news articles: UP, DOWN, and NORMAL. The naïve Bayesian classifier learns a predictive model for each of these classes. This classifier upon receiving a novel news article returns the posterior probabilities of the 3 classes which then can be used to generate action signals like BUY, SELL, NO ACTION.

Since profit opportunities in the stock market are present for an extremely short time, the frequency of the availability of the information is essential to profitable trading strategies. We have gathered price information and news articles on a minute-by-minute basis for the DJI index and its top 30 components for a period of little less than 8 months from 7/26/2001 to 3/16/2002. Over this period we obtained a total of 25087. Since price information is only available on official trading days from 9:30 AM to 4:30 PM, we discard all news articles from our data set that could have an ambiguous effect. Thus after discarding news articles that were retrieved during non-business hours we were left with a total of 13372 news articles.

A closer look at the database reveled that the information gathered is incomplete. Defining a gap as any time period for which we do not have price information for more than 1 minute, we find that on average there are 270 gaps for the index and each of the 30 components. These gaps are most likely

results of networking problems and system issues, most of which have already been resolved. Fortunately the gaps for individual components and the index are more or less aligned.

We estimate that with the exception of a few large gaps (approximately 4 with a total length of 30 days) if one discards 30 minutes before the the gap start and 30 minutes after the gap start, including the gap interval we roughly lose 11 days of business trading time. Since we are interested in predicting the effects of news articles within a very short time period around the time of appearance of the news article, it is essential that for each news article we use we have complete price information (no gaps) during that time period. Thus we discarded news articles from our set for which we did not have sufficient information within a $[-30, +30]$ time interval centered around the appearance of the news article, leaving us with a total of 12437 news articles.

To overcome the slight misalignment of price information due to gaps we take the following steps. Since price information about the index is essential in evaluating the relative performance of individual stocks we discard all price information about individual stocks where we do not have price information about the index for the corresponding time. In case of missing stock price information we substitute the missing information with the next available price information for that stock. More precisely, if at time t we have price information for the index but do not have price information for the stock, we look for the next available price information for that stock that has a timestamps which is greater than t and is the smallest among all the timestamps for the stock.

3 Indicators derived from textual data

As was mentioned in the previous section, we derive a set of indicators from textual data using a naïve Bayesian text classifier. This derivation can be divided into the following subparts:

- Definition of movement classes,
- Alignment of news articles
- Scoring of news articles
- Labeling of news articles
- Training a naïve Bayesian text classifier for the movement classes, and using posterior probabilities for each news article as indicators.

Unlike in the usual text classification framework in our task the news articles are initially unlabeled. Defining classes and obtaining labels for training examples is crucial for any classification task. The following subsections describe in detail our approach to accomplish this task.

3.1 Aligning of news articles

According to our initial hypothesis news articles contain information that have an effect on stock prices. To evaluate this possible effect of a news article, for each article we define a time interval that we call the *window of influence*. The window of influence of a news article is the time period throughout which that news article might have an effect on the price of the stock. The window of influence of a particular article d with a time stamp t can be characterized by lower boundary offset and an upper boundary offset from t. An offset is negative if $t + offset$ is prior to t. As an example figure 2 shows on the left an alignment with $[-20, +30]$ offsets and on the right an alignment with $[+10, +30]$ offsets.

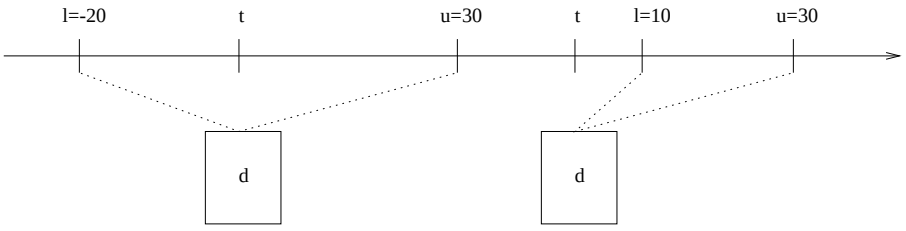

Figure 2: Illustration of alignments with different offsets. l and u are abbreviations for lower and upper boundary offsets and are measured in minutes.

3.2 Scoring of news articles

One commonly used method to evaluate the performance of a particular stock is based on the volatility of the stock, which is known as the β-value. In short, this β-value describes the behavior of the stock relative to some index, and is calculated using a linear regression on the data points (Δindex-price, Δstock-price).

The previously mentioned gaps in the data posed several difficulties when calculating the β-values of the individual stocks. We perform linear regression on the change in price of the index and the change in price of the stock, where the change is calculated on an hourly basis.

By plotting the changes in the index prices vs the changes in the stock prices we find a few outliers. We call a data point an outlier when either the absolute change in index price or the absolute change in the stock price is larger then 2 percent. Based on the examination of a few stocks that most of these outliers are due to off-market trading. During off-market hours non-processional traders conduct trades that influence the stock price. Thus on two consecutive official trading days it is quite common that the closing prices from the previous day differ by a substantial amount from the opening prices of the following official trading day. We find several "outliers," which in this case should not be considered outliers, around the tragic events of September 11. Finally, we find outliers at the boundaries of larger gaps in the data. When calculating the volatility of stocks we discard all of the true outliers.

Assuming a linear model the obtained β-value describe the following characteristic of the stock. A stock with a β-value of 1 has the characteristic that whenever the percent change for the index price is δ the percent change for the stock price is expected to be δ as well. Similarly a stock with a β-value of 2 has the characteristic that whenever the percent change in the index price is δ the percent change in the stock price is expected to be 2δ. In other words, stocks with a β-value greater than 1 are considered to be relatively volatile, while stocks with a β-value less then 1 are considered to be more stable.

To eliminate the effects of the exponential change of stock prices we calculate the change on a log scale according to the following formula:

$$\Delta price(u, v) = ln \frac{price(v)}{price(u)}.$$

We define m, the *movement* of a stock in the time interval $[u, v]$, as follows:

$$m(u, v) = \frac{\Delta p_s(u, v)}{\beta} - \Delta p_i(u, v),$$

where $\Delta p_s(u, v)$ and $\Delta p_i(u, v)$ are the changes in price for the stock and the index in the time interval $[u, v]$. Using this movement measure a news article d with timestamps t when aligned with offsets $[u, v]$ receives a score $m(t + l, t + u)$.

3.2.1 Labeling news articles

The movement of zero at a particular time and for a particular window of influence means that the price change of the stock at that time is normal. Similarly, a movement greater than or less then zero means that the price change of the stock is *relatively* better or worse than the expected. We emphasize the word relatively, since according to our scoring method a news article may receive a positive score even if the change in the stock price is negative. Our measure of movement is a relative one, which is not only based on the change in stock price, but is also based on the change in the index price and out expectation of the stock's behavior to this change. We define the movement classes: upward movement (UP), downward movement ($DOWN$), and expected or normal movement ($NORMAL$), according to the following rules:

$$mc(m) = \begin{cases} UP & m > \rho_{positive} \\ DOWN & m < \rho_{negative} \\ UP & m > \text{otherwise} \end{cases}$$

Although this rule for labeling news articles may seem simple, the task at hand is highly non-trivial. As we pointed out earlier assigning correct labels to news articles is essential to our classification task. To demonstrate the importance and difficulty of the labeling process consider a case depicted in figure 3. Let us assume that the true distribution of the news articles in the three classes is as shown above the axis. Here we assume that each class has an associated language usage, which are distinct from the others. In particular we would expect that words like *lost*, *shortfall*, or *bankruptcy* would occur more

frequently in news articles that correspond to downward movements in price, whereas worlds like *incept*, *propel*, or *peak* would occur more frequently in news articles that correspond to upward movements in price. Setting our threshold values $\rho_{negative}$ and $\rho_{positive}$ to the values shown in figure 3, results in a non-optimal or incorrect labeling. Some news articles that discuss downward movement in price of a certain stock are labeled as NORMAL in our model, and some articles that discuss expected movement in price are labeled as UP in our model.

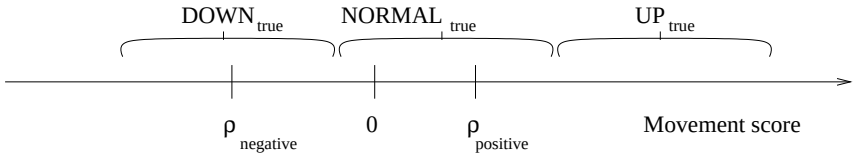

Figure 3: Implications of non-optimal labeling of news articles. Incorrect settings of threshold values $\rho_{negative}$ and $\rho_{positive}$ results in mislabeling news articles that should belong to the NORMAL class.

Yet another possible source for confusion between distinct languages for the classes could be due to the reappearance of similar or identical news articles. Similar or identical news articles can reappear with a time shift due to different sources reporting similar things, or due to corrections by the same source. This clearly could cause confusion in the predictive models that we are trying to learn, since news articles describing some negative news (should be assigned to class DOWN) might be assigned to class UP or NORMAL because of the time shift of the publication of the news article. This situation is depicted on figure 4.

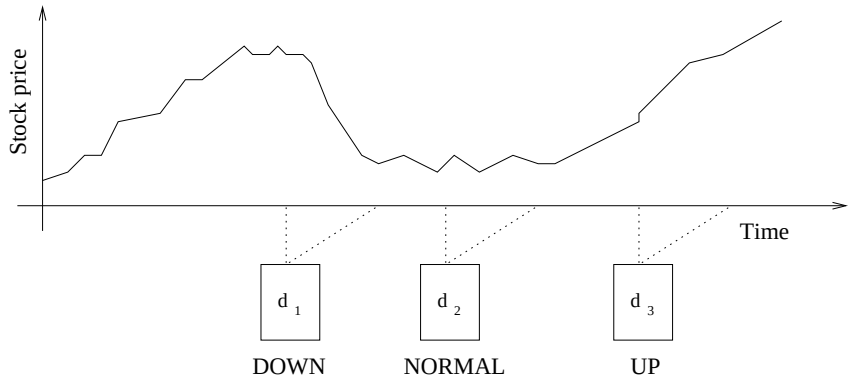

Figure 4: Illustration of how the reappearance of news articles can create possible confusion in the distinct languages for the classes. In this example we assume that the news article d_1 clearly describes some negative news, and d_2 and d_3 are news articles that were published with some correction or by a different news source, but have very similar language usages.

To eliminate this source for confusion we measure the similarity of news articles based on the edit distance of the first 256 characters of the news articles. For each stock specific news article we calculate this measure for every stock specific news article that has appeared in the last 24 hours, and assign the most similar score (lower means more similar) to the news article. Thus news articles with low similarity scores are most likely republished or updated news articles. We eliminate news articles from out dataset whose similarity score is less then the threshold value s.

3.3 Learning of naïve Bayesian text classifier and extracting indicators

After we have defined movement classes and assigned news articles to these classes our learning task can be phrased as follows. Given a news article d, we would like to predict the probability that a movement class c will be observed in the window of influence of news article d. Using Bayes rule we can calculate this conditional probability as:

$$P(M = c|d) = \frac{P(d|M = c)P(M = c)}{P(d)}$$

Since $P(d)$ is independent of the class c we can rewrite the above equation as follows:

$$P(M = c|d) \sim P(d|M = c)P(M = c)$$

After assuming the conditional independence of words within a news article, given a movement class, this can be rewritten as:

$$P(M = c|d) \sim \prod_{w \in d} P(w|M = c)P(M = c)$$

This is the probability modeled by a naïve Bayesian classifier. We have chosen to use the Rainbow naïve Bayesian classifier package [4] for our classification task. We state the exact options used for the classification in the experiment section.

After having trained a classifier for the movement classes, for each novel news article d we calculate the posterior probability $P(M = c|d)$ for every movement class $c \in \{\text{UP,DOWN,NORMAL}\}$ and use them as possible indicators. Hence for each news article we obtain three numerical indicators: $P(M = \text{UP}|d)$, $P(M = \text{DOWN}|d)$, and $P(M = \text{NORMAL}|d)$.

The naïve Bayesian classifier that we trained not only provides a discrete classification, but rather a soft assignment of news articles to the 3 classes. It makes sense to filter out predictions where these probabilities are equal or close to equal (i.e.: the classifier is not so confident about the prediction). We do this by leaving out news articles from our test set during the evaluation phase for which none of the class probabilities is greater than some certainty threshold σ.

4 Experimental design

To evaluate our model for predicting movements in stock prices using indicators derived from news articles, in the following sections first we describe the experimental setup, then derive some evaluation metrics, and discuss the results of several test.

4.1 Experimental setup

As mentioned earlier, we have gathered price information and news articles on a minute-by-minute basis for the DJI index its top 30 components from 7/26/2001 to 3/16/2002. We divide our data such that news articles time-stamped before 11/01/2001 9:30AM belong to the training set and news articles after that date belong to the test set. After leaving out news articles for which we do not have sufficient price information due to gaps or which were retrieved during non-business hours, we obtain a training set of 3,200-3400 articles and a test set of 2,400-2,600 articles depending on the particular alignment. The reported experimental results are all results of classifications performed on the test set.

After having tried several settings and options for the Rainbow text classifier we decide to use the Wittenbell smoothing method assume uniform prior distribution of news articles across classes, remove stop words, use stemming, and use only the top 1,000 words with highest mutual information for the classification. Unless otherwise stated we apply these settings to all the test performed.

4.2 Evaluation metrics

The most common evaluation metrics for classifiers are precision, recall, average prediction accuracy over all the classes. In our case however if we associate actions with particular predictions these actions yield some real-valued profit. Assuming a high class prior on the NORMAL class, a classifier that always correctly predicts the NORMAL class and always confuses the UP and DOWN class for each-other may have a better average prediction accuracy than a classifier that always correctly predicts the UP and DOWN classes and always confuses the NORMAL class. Clearly the predictions made by the later classifier carry more opportunities for profit within them.

4.2.1 Normalized economic value estimates for performance evaluation

True economic value for a set of predictions is hard to calculate since different trading policies will result in different asset allocations at different times. One reasonable assumption to make is that upon predicting classes NORMAL, UP, and DOWN a reasonable action to take would be NO ACTION, BUY, and SELL respectively. Under these assertions for each news article (i.e. prediction) we must assign an economic

value estimate. Recall that we define our 3 classes based on a movement score that measures the change in the stock price during the window of influence of a particular news article with respect to the change in the index price for the same time period. Furthermore note that we adjust this score using the volatility or β-value of the particular stock to take into account the expected behavior of the stock. Here we recall the definition of movement:

$$m(u,v) = \frac{\Delta p_s(u,v)}{\beta} - \Delta p_i(u,v),$$

where $\Delta p_s(u,v)$ and $\Delta p_i(u,v)$ are the changes in price for the stock and the index in the time interval $[u,v]$.

As a first approximation we use this same measure to associate an economic value estimate with each of the 3 actions. The adjustment using the volatility of the stock is necessary to assure consistency between predictions and economic value estimates. Consider the case in which for some time interval both the change in stock price and in index price is 1. If $\beta = 1$ then $m = 0$; if $\beta = 0.5$ then $m = 1$; finally, if $\beta = 2$ then $m = -0.5$. Clearly depending on the β-value the "true" class of a news article during this time interval would be considered NORMAL, UP, and DOWN. Simply taking $\Delta p_s(u,v) - \Delta p_i(u,v) = 0$ as the economic value estimate would clearly create inconsistency in the association of economic value estimates and the actions. Using the the movement m of a stock as the economic value estimate of a prediction guarantees that when the true class is UP the economic value estimate is positive, and when the true class is DOWN then the economic value estimate is negative. Based on the above discussion we associate the following actions with predictions and economic value estimates with predictions.

	Predicted		
	NORMAL	UP	DOWN
NORMAL	0	m	$-m$
UP	0	m	$-m$
DOWN	0	m	$-m$

Table 1: Mapping from predictions to economic value estimates. Rows represent actual while columns represent predicted classes.

In the table above m is the score that the news article received. Thus the economic value estimate of the predictions for a set of news articles D is:

$$E = \sum_{d_i \in D | p_i = \text{UP}} m_i - \sum_{d_i \in D | p_i = \text{DOWN}} m_i,$$

where d_i is a news article in the set of news articles D, p_i is the predicted class of the news article d_i, and m_i is the score of news article d_i.

In order for the economic value estimate to be a true performance measure across predictions for different sets of news articles we must normalize it. We do this by dividing the economic value estimate of the predictions for a set of news articles by the following term

$$\sum_{d_i \in D | t_i = \text{UP}} m_i - \sum_{d_i \in D | t_i = \text{DOWN}} m_i,$$

where t_i represents the true class of the news article d_i. This term represents the maximum possible economic value achievable based on out simple trading policy.

The so obtained normalized economic vale estimate, NEVE, has two major drawbacks. First, in the case $p_i \in \{\text{UP, DOWN}\}$ and $t_i = \text{NORMAL}$ m_i can take on relatively small negative and positive values. This adds some noise to E, which cannot be accounted for in the normalization step since the appropriate action in these cases would be NO ACTION. Thus the NEVE is not restricted to the range of $[-1, 1]$, where a value of -1 represents a completely incorrect set of predictions in terms of economic value estimates, and a value of 1 represents a completely correct set of predictions in terms of economic value estimates. Second, the meaning of NEVE not easy to understand, not even its unit is clear.

4.2.2 Average profit per trade for performance evaluation

To overcome the two difficulties of NEVE metric mentioned above we use the average profit per trade as an evaluation metric. In order to associate profit with actions we use a minimal risk trading policy. To achieve minimal risk we conduct every trade as follows. In previous steps for each stock we have estimated how volatile that stock is. This volatility is reflected by the β-value of the stock. It is clear that a sensible action upon predicting the UP and DOWN class would be to BUY and SELL the stock. At the same time, to minimize risk we also SELL and BUY proportional amount of the index. To see why this set of actions achieves minimal risk based on our expectations consider the following scenario. For a particular stock we estimate the β-value to be 2. Upon predicting the UP class we buy \$1000 worth of shares from the stock. Assume that for a particular alignment $[0, 20]$ the stock goes up by 1% in 20 minutes. Selling our stock shares at this time would result in a profit of \$10. For the same period of time we expect the index to go up by 0.5%. If upon prediction the UP class we sell \$1000 * β worth of shares from the index and 20 minutes later buy back the same amount of index shares we expect the profit from this transaction to be -\$10. Thus according to our expectations the two transactions together would result in a zero profit or loss, thereby minimizing the risk. Clearly this procedure only achieves zero profit or loss if our expectations are correct. To formalize this idea let us define two quantities g_s – gain from buying and later selling $\$M$, and g_i – gain from buying and later selling $\$M * \beta$ from the index:

$$g_s = \left(\frac{\$M}{sp_s} ep_s\right) - \$M$$

$$g_i = \left(\frac{\$M\beta}{sp_i} ep_i\right) - \$M\beta,$$

where sp and ep are starting and ending prices and subscripts represent index and and stock information. Then the profit associated with buying the $\$M$ from the stock and selling an proportionate amount of the index is $g_s - g_i$, and the profit associated with the reverse transaction is $g_i - g_s$. It is true for this measure that news articles in the UP class have a positive, while news articles in the DOWN class have a negative profit associated with them. Thus using this measure as our evaluation metric the mapping from predictions to profits is as follows:

	Predicted		
	NORMAL	UP	DOWN
NORMAL	0	$g_s - g_i$	$g_i - g_s$
UP	0	$g_s - g_i$	$g_i - g_s$
DOWN	0	$g_s - g_i$	$g_i - g_s$

Table 2: Mapping from predictions to profits. Rows represent actual while columns represent predicted classes.

Using this mapping we obtain the total profit for a set of predictions by summing the individual profits for each of the predictions. To use this metric to evaluate and compare different set of predictions we normalize the total profit by the number of trades performed.

Average profit per trade is an proper measure to compare performance for different sets of predictions. To establish a baseline performance let us define the following quantities for a trader. Let p_t represent the trading probability, and p_s and p_b be the probability of performing action SELL and BUY respectively given we perform a trade. Given that for a particular aliment these quantities are known for 3-class naïve Bayesian trader, we can establish a baseline performance for a random trader. This random trader upon receiving a news article performs a BUY action with probability $p_t * p_b$, a SELL action with probability $p_t * p_s$ and a NO ACTION action with probability $1 - p_t$. To guarantee that both the 3-class naïve Bayesian trader and the random trader perform thew same number of transaction we "wrap around" in case the random trader did not perform enough transactions. Since a single random trader can easily produce results that are due to random chance, we average the results of K number of random traders.

5 Experimental results

The alignment of news articles with prices is based on the concept of window of influence. The window of influence of a news article is the time period throughout which that news article might have had an

effect on the price of the stock.

We have performed several tests for different alignments. The result of these test are shown in figure 5 - 6. In these figures the x-axis shows the window boundary offsets in minutes for the the different alignments and should be interpreted as follows. For negative window boundaries we implicitly assume that the upper window boundary offset is zero. While for positive values for window boundaries we implicitly assume zeros values for the lower boundary offsets. We test alignments for window boundaries in increments of 5 with the exception around the zero value, where this measurement is slightly refined ([-3,0]. [0,3] alignments). In order to obtain well balanced classes the labeling threshold $\rho_{negative}$ and $\rho_{positive}$ are always recalculated for each alignment such that the frequencies in classes NORMAL, UP, and DOWN are approximately 50, 25, and 25 percent respectively. It is important to note that due to the relative sizes of the classes a classifier that always predicts the NORMAL class–from here on referred to as the "highest prior" classifier–will always have an average prediction accuracy of approximately 0.5.

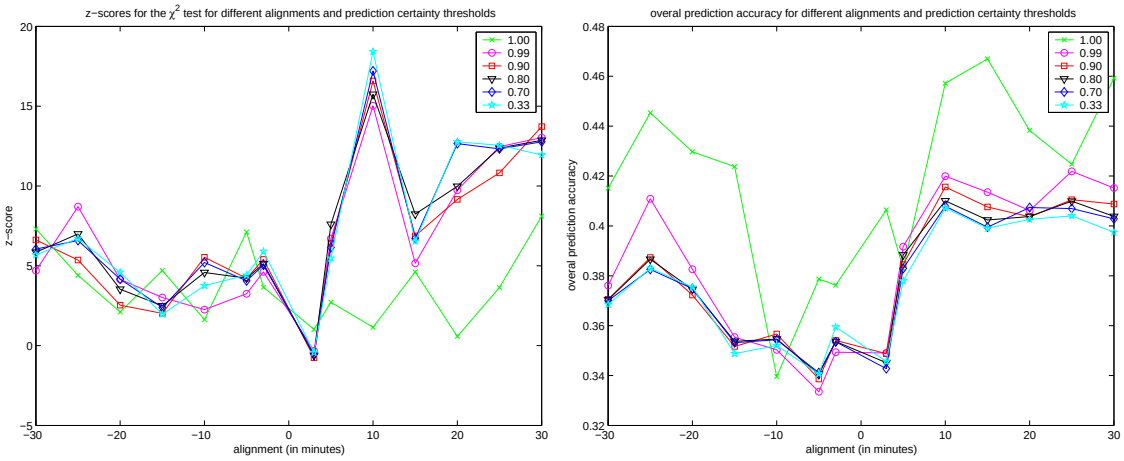

Figure 5: Graph on the left shows the significance of predictions for different alignments based on z-scores obtained from the χ^2 test for various prediction certainty thresholds. Generally, z-scores of more than 6 are considered significant. Graph on the right shows the overall prediction accuracy of the 3-class naïve Bayesian classifier and the "highest prior" classifier for the corresponding alignments and prediction certainty thresholds.

As one can see in left graph of figure 5, the z-scores derived from the χ^2 test suggest that alignments [0,5], [0,10], [0,15], [0,20], and [0,25] are highly significant. Even thought, as the right graph in figure 5 shows, there is a slight increase in overall performance accuracy of the learned 3-class naïve Bayesian classifier with respect to other alignments, the classification accuracy of the classifier that always predicts the class with the highest prior is superior (approximately 0.5). It is important to note that this later classifier always predicts the NORMAL class and hence has no economic value at all.

We observe that setting prediction certainty threshold σ to higher values–only slight, but in all cases–increases the overall prediction of the classifier.

Figure 6 shows the prediction accuracy of the 3-class naïve Bayesian classifier for each individual class. Even though, currently we cannot suggest any reason for, it is interesting to note the change in prediction accuracies between the relevant classes UP and DOWN around the zero value.

5.1 Similarity tests

We perform a set of test by leaving out news articles that have a similarity score s less then a particular threshold value. Results of these tests are shown in figure 7. For all of these tests we use the $[0, 10]$ alignment, which we find to be most optimal in terms of z-scores and average prediction accuracy based on the previous tests. It is important to note, that measurement at the zero value represents the test result where no news articles were discarded due to similarity. The 2 graphs suggest that discarding similar news articles reduces the significance of predictions as the similarity threshold is increased. This is most likely due to the limited number of news articles, since as the similarity threshold is increasing the number of news articles that remain in our dataset is decreasing.

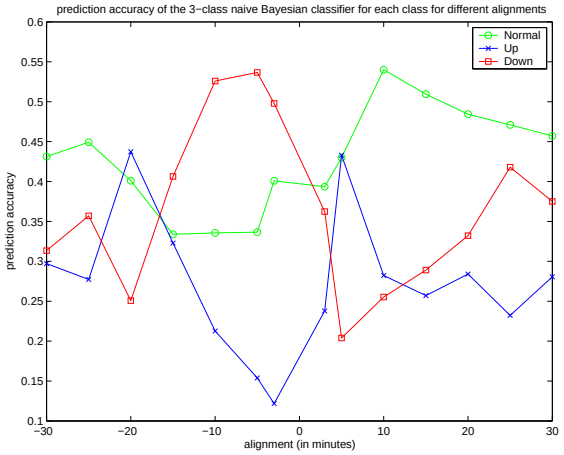

Figure 6: Per class prediction accuracy of 3-class naïve Bayesian classifier for different alignments. Prediction certainty threshold is set 0.33, thereby considering all predictions made by the classifier.

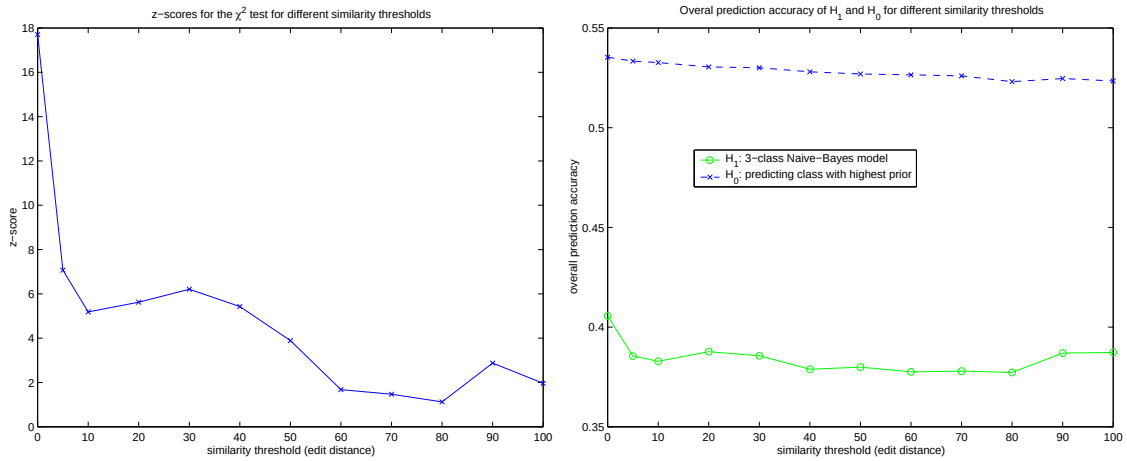

Figure 7: Graph on the left shows the significance of predictions for different similarity thresholds based on z-scores obtained from the χ^2 test. Generally, z-scores of more than 6 are considered as being significant. Graph on the right shows the overall prediction accuracy of 3-class naïve Bayesian classifier and "highest prior" classifier for different similarity thresholds. Note that, in both cases the measurement at the zero value represents the result of the test where no news articles were discarded due to similarity.

5.2 Economic evaluation of predictions

Figures 8 and 9 show the results of the alignment tests when evaluated using the two economic metrics devised in the previous section. Figure 8 shows the normalized economic value estimates, while figure 9 shows the average profit per trade for various alignments and prediction certainty thresholds. In the later figure error bars represent standard error for the particular sets of predictions.

To find sets of predictions with performance that are statistically significants from the performance of the baseline produced by the average of K random traders for that particular alignment and prediction certainty threshold we perform a 2 sample t-test for the equality of the means or average profits per trades produced by the 3-class naïve Bayesian trader and the K random traders. Table 3 shows the results of these tests, where the null hypotheses is that the two means are equal. 0 entries in the table show cases where the null hypothesis is accepted. In all the other cases the null hypothesis is rejected, and -1 (if present) represent cases where the the performance of the 3-class naïve Bayes trader was worse, while 1 represents cases where it was better than the average of the K random traders. All the t-tests are based on significance level $\alpha = 0.1$.

We find that for the $[-20, 0]$ and the $[0, +20]$ alignments for prediction certainty threshold $\sigma = 0.33$

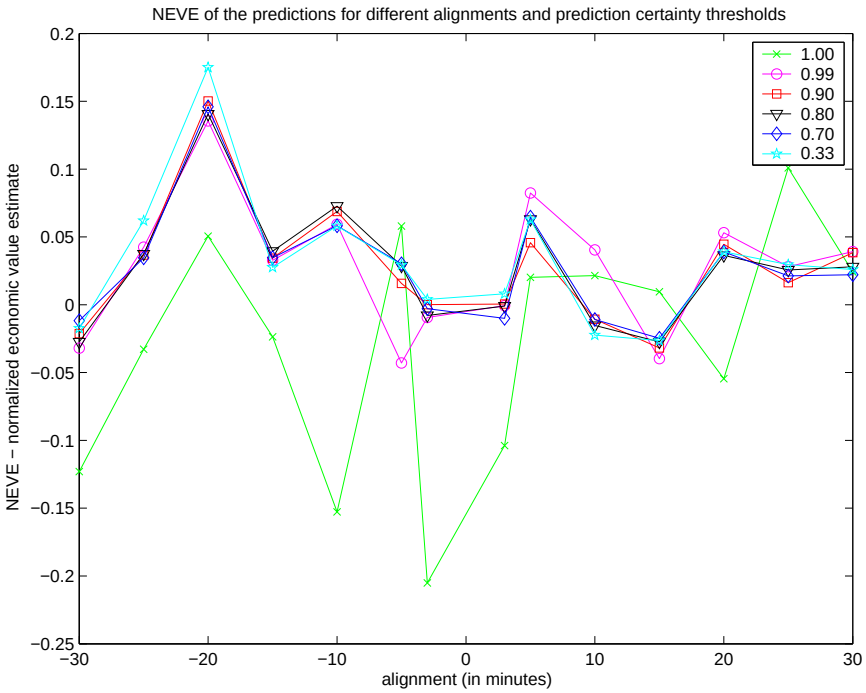

Figure 8: Normalized economic value estimates for different alignments for various prediction certainty thresholds. Positive values represent predictions that have a potential for generating profit under our simple trading policy.

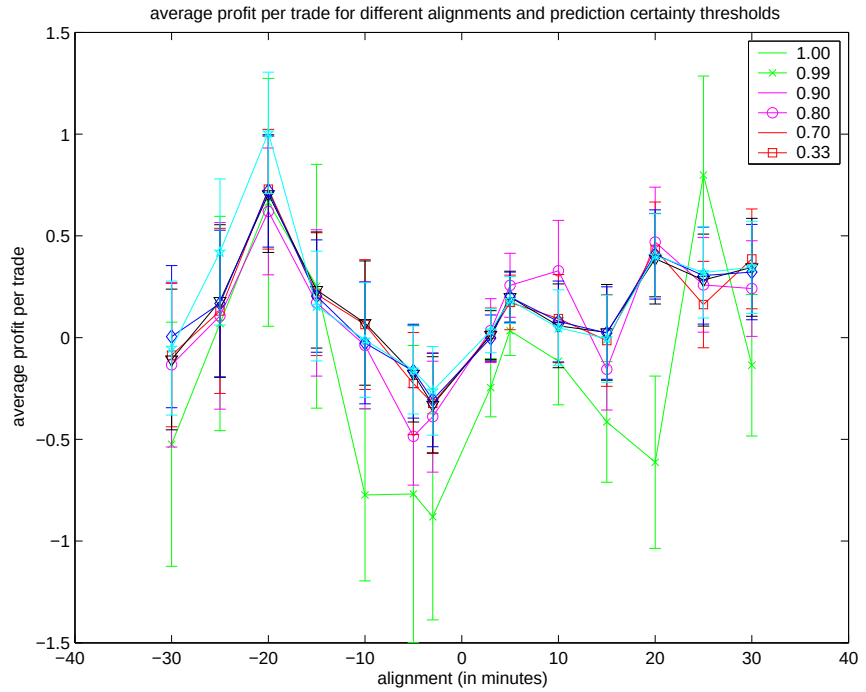

Figure 9: Average profit per trade for different alignments for various prediction certainty thresholds. Units of profit is dollars and is based on trading \$1000 using a minimal risk trading policy.

the performance of the 3-class naïve Bayesian trader is significantly better from the performance of 1000 random traders. For the $[-20, 0]$ alignment both the normalized economic value estimate and the average

Certainty	Alignments													
	-30	-25	-20	-15	-10	-5	-3	3	5	10	15	20	25	30
1.00	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0.99	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0.90	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0.80	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0.70	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0.33	0	0	1	0	0	0	0	0	0	0	0	1	0	0

Table 3: Comparing prediction performance based on average trade per profit between the 3-class naïve Bayesian trader and $K = 1000$ random traders. Os mean that the difference in performance was not statistically significant, 1s mean that the 3-class naïve Bayesian trader is superior, while -1s (if present) mean that the random traders are superior. Significance tests are based on a t-test with confidence level $\alpha = 0.1$.

profit per trade have high values. For this particular setting the average profit per trade is \$1.0063 with a standard error of $\pm$\$0.2979.

The t-test employed to test for the equality of two means assumes that the two samples come from a normal or an approximately normal distribution. While this is true for the 3-class naïve Bayesian trader it does not hold for for the random traders. This fact is depicted on figure 10. In this figure we show the distribution of average profits per trade of $K = 1000$ random traders. Graphs to the left of the center show negative while graphs to the right of the center show positive alignments. Graphs from top to bottom show alignments in order with increasing window of influence.

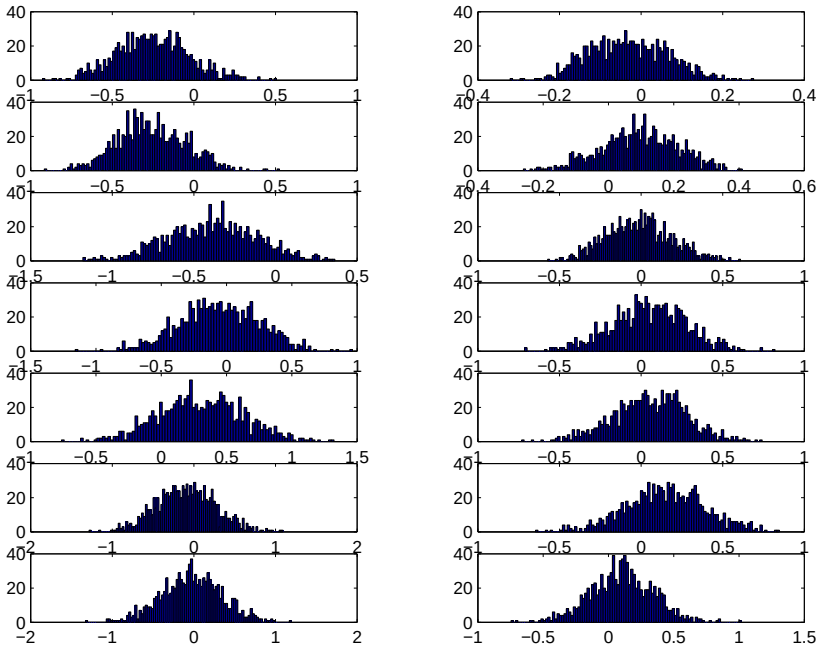

Figure 10: Distribution of average profit per trade generated by $K = 1000$ random traders for various alignments. Graphs on the left show negative while graphs on the right show positive alignments. The window of influence increases going from graphs on the top to graphs on the bottom. In all the graphs the x-axis shows average profit per trade while the y-axis shows frequency counts.

Since the applicability of the t-test is questionable in our case in table 4 we report percentiles for the same $K = 1000$ random traders and our 3-class naïve Bayesian trader. In this table a value of 1 means that our 3-class naïve Bayesian trader generated better average profit per trade than any of the random traders, while a value of 0 means that the 3-class naïve Bayesian trader was outperformed by every random trader.

Certainty	Alignments													
	-30	-25	-20	-15	-10	-5	-3	3	5	10	15	20	25	30
1.00	0.281	0.587	0.652	0.741	0.507	0.207	0.065	0.183	0.513	0.37	0.195	0.091	0.727	0.256
0.99	0.428	0.736	0.826	0.803	0.892	0.252	0.305	0.739	0.786	0.924	0.211	0.913	0.551	0.548
0.90	0.491	0.766	0.886	0.814	0.926	0.739	0.409	0.692	0.596	0.682	0.42	0.916	0.454	0.843
0.80	0.491	0.808	0.88	0.85	0.943	0.746	0.435	0.668	0.747	0.64	0.481	0.903	0.705	0.767
0.70	0.549	0.795	0.914	0.825	0.886	0.737	0.445	0.626	0.765	0.661	0.465	0.93	0.741	0.782
0.33	0.468	0.916	0.987	0.739	0.914	0.685	0.49	0.733	0.773	0.627	0.403	0.942	0.757	0.835

Table 4: Percentiles for the 3-class naïve Bayesian trader against the $K = 1000$ random traders for various alignments and prediction certainty values. Higher values are better.

This test confirms our previous observation and shows that for the $[-20, 0]$ alignment for prediction certainty threshold $\sigma = 0.33$ the 3-class naïve Bayesian trader outperforms over 98.7% of the random traders.

5.3 Internals of the predictive model

To gain more confidence in our learned predictive model we take a look at the internals and examine which words are used by the model to distinguish between the 3 classes. Since we obtain the best economic performance for the $[-20, 0]$ alignment we examine the the predictive model that was trained for this particular alignment. Table 5 shows the top 100 worlds ordered decreasingly by mutual information for the 3 classes. Table entries in columns labeled NORMAL, UP and DOWN are counts for the particular words in each of the classes.

Within these top 100 words with high mutual information we find worlds like *incumbent*, *exploited*, *notify*, *unsophisticated*, *sophisticated*, *detergents*, *damn*, *iplanet*, *netto*, *settings*, *elinor*, and *abreusan* are highly predictive for the UP class. While, words like *counted*, *motc*, *indefeasible*, *upgradeable*, *terrabits*, *quadruples*, *chain*, *fang*, *mti*, *microwave*, *hilliard*, and *chow* are highly predictive for the DOWN class. Further down the list ordered by mutual information (not shown in table 5) we find *rose*, *bugs*, *fixes*, *confidently*, *perspectives*, *retaliating*, *minimize*, *funds*, *improve*, and *compensation* to be predictive of the UP class, and *comprises*, *experimental*, *auditors*, and *outstandings* to be predictive of the DOWN class.

To gain better understanding as to what phenomena the presence of these words are capturing, and to test whether the presence of some of these words capture the same phenomena we calculate the pairwise correlation between the top 100 words that have the highest mutual information with the classes.

Figures 11-13 show pairwise correlation matrices for these words. It is important to note that the word order shown in the correlation matrices is different from the order determined by mutual information. The word order for these correlation matrices are shown in the last column of table 5. In these figures a solid green line through a particular row or column means the absence of that word in the given class. Positive correlations are shown in red, while negative correlations (if present) are shown in blue. Darker tones represent stronger correlations.

Not surprisingly, we find that words *dow*, *jones*, and *index* frequently co-occure in documents as a reference to the Dow Jones Industrial index. The co-occurrence of words *zykan*, *tgda*, and *tangibledata* is due to the fact that Tony Zykan is the president of Tangibledata, a company represented by the symbol tgda. We also find that words *elinor* and *abreusan* are also strongly correlated in all the classes, which is due to the fact that Elinor Abreu is a reporter in the San Francisco area, who appears to write mainly about news that correspond to an upward movement of the price for the particular stock.

WORD	MI	NORMAL	UP	DOWN	MI rank	Correlation index
sbc	0.00468	134	314	416	1	44
msft	0.00400	755	249	244	2	28
websphere	0.00371	383	9	39	3	68
db	0.00350	84	6	2	4	52
index	0.00349	2075	1510	1246	5	8
microsoft	0.00335	2321	1052	780	6	27
safeonline	0.00319	0	14	0	7	99
content	0.00302	323	101	75	8	13
defaultsignaling	0.00296	0	13	0	9	100
merck	0.00293	236	275	249	10	19
cingular	0.00289	13	20	60	11	93
russell	0.00275	134	143	102	12	18
incumbent	0.00270	1	17	3	13	54
software	0.00266	2054	876	678	14	14
domino	0.00265	38	0	2	15	70
industrials	0.00259	44	61	47	16	17
ibms	0.00254	404	69	82	17	65
historic	0.00252	35	3	2	18	24
exploited	0.00250	1	18	2	19	91
notify	0.00248	3	15	1	20	74
unsophisticated	0.00246	1	13	0	21	40
lax	0.00246	1	13	0	22	48
broadcasters	0.00244	25	0	0	23	39
mrk	0.00242	80	86	93	24	45
johnson	0.00239	296	300	293	25	32
fared	0.00239	16	35	8	26	49
landing	0.00237	33	0	23	27	26
tobacco	0.00236	65	134	153	28	36
poors	0.00236	768	513	447	29	7
fulfillment	0.00232	40	2	0	30	64
weeks	0.00231	620	498	437	31	1
space	0.00228	360	109	86	32	16
counted	0.00228	5	3	19	33	55
calhoun	0.00228	0	10	0	34	96
motc	0.00227	3	0	13	35	77
indefeasible	0.00227	3	0	13	36	78
upgradeable	0.00227	3	0	13	37	79
terabits	0.00227	3	0	13	38	80
quadruples	0.00227	3	0	13	39	81
chian	0.00227	3	0	13	40	82
feng	0.00227	3	0	13	41	83
mti	0.00227	6	0	26	42	84
microwave	0.00227	9	0	39	43	85
key	0.00226	635	348	188	44	2
aboutus	0.00224	1	24	0	45	71
illustrated	0.00221	4	15	0	46	51
played	0.00220	56	8	6	47	25
fusion	0.00219	4	79	0	48	60
backdoors	0.00219	2	13	0	49	87
philippines	0.00217	9	3	20	50	76
standard	0.00215	943	604	524	51	5
proprietary	0.00212	78	14	16	52	43
sophisticated	0.00211	46	57	10	53	33
leverages	0.00210	31	0	2	54	42
thornton	0.00207	17	0	0	55	53
wurzler	0.00207	1	13	1	56	92
href	0.00207	5601	2392	2037	57	9
selecting	0.00206	23	2	0	58	57
detergents	0.00205	0	11	0	59	98
damn	0.00204	0	11	2	60	95
jones	0.00201	634	451	372	61	4
crossings	0.00200	17	7	44	62	75
tower	0.00199	26	0	1	63	34
feeds	0.00199	22	0	1	64	37
iplanet	0.00199	3	20	5	65	90
seattle	0.00199	214	55	35	66	21
neto	0.00198	5	36	0	67	59
settings	0.00198	11	21	1	68	72
dow	0.00198	1630	1108	920	69	3
greenspan	0.00197	112	109	67	70	35
bundled	0.00197	24	42	4	71	47
material	0.00195	112	26	27	72	15
zoellick	0.00195	24	0	0	73	23
tangibledata	0.00195	112	0	0	74	61
tgda	0.00195	48	0	0	75	62
zykan	0.00195	48	0	0	76	63
mq	0.00195	16	0	0	77	69
elinor	0.00195	3	15	1	78	88
abreusan	0.00195	3	15	1	79	89
wound	0.00195	0	10	1	80	97
distributed	0.00193	59	67	11	81	41
intangibles	0.00191	12	21	4	82	73
html	0.00191	2496	1073	877	83	10
default	0.00191	10	28	7	84	46
hilliard	0.00191	0	1	9	85	94
permit	0.00191	7	2	16	86	22
resellers	0.00188	12	19	1	87	67
christmas	0.00187	83	47	10	88	38
pharmaceutical	0.00186	111	130	77	89	50
deliver	0.00186	234	150	70	90	6
launch	0.00186	387	143	115	91	20
decentralized	0.00186	22	1	0	92	56
align	0.00185	578	262	199	93	11
insecure	0.00184	4	13	0	94	58
chow	0.00184	9	2	39	95	86
graphics	0.00184	125	29	58	96	31
granted	0.00184	20	10	32	97	29
lotus	0.00183	132	36	8	98	66
gapfiller	0.00183	15	0	0	99	30
transition	0.00183	95	24	17	100	12
TOTAL		895707	503727	431755		

Table 5: Top 100 words in decreasing oder by mutual information for the classes.

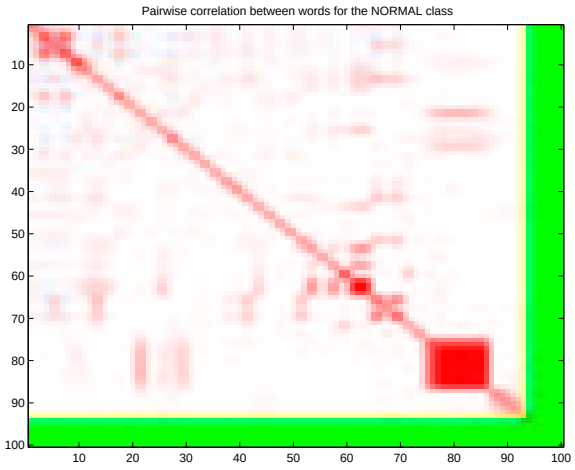

Figure 11: Pairwise correlation for the top 100 words for the NORMAL class.

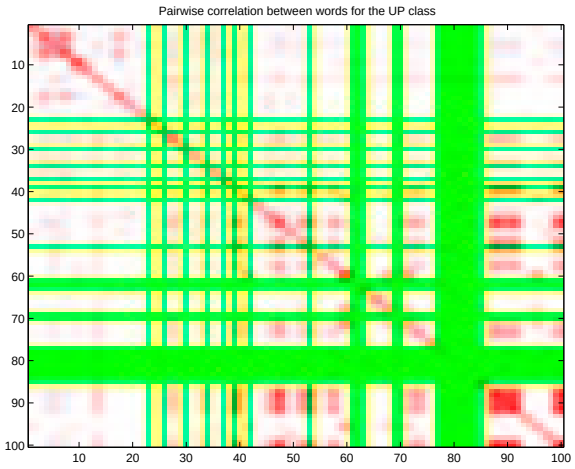

Figure 12: Pairwise correlation for the top 100 words for the UP class.

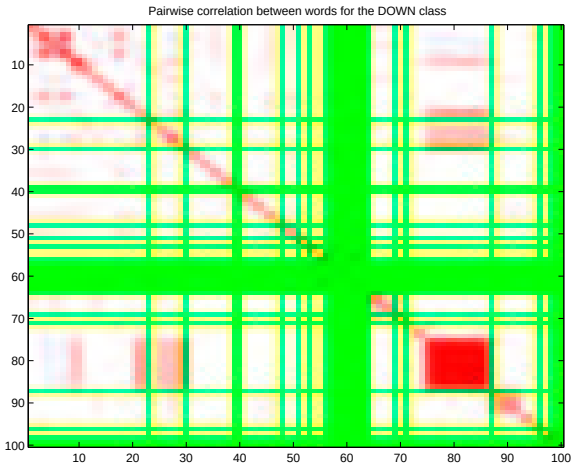

Figure 13: Pairwise correlation for the top 100 words for the DOWN class.

References

[1] Elkan, C., (1999). Notes on discovering trading strategies.

[2] Gidófalvi, G., (2001). Using News Articles to Predict Price Movements.

[3] Lavrenko, V., Schmill, M., Lawrie, D., Ogilvie, P., Jensen, D., & Allan, J. (2000) Mining of Concurrent Text and Time Series. KDD-2000: Workshop on Text Mining.

[4] McCallum, A.K. (1996) "Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering." http://www.cs.cmu.edu/ mccallum/bow.

[5] Thomas, J.D. & Sycara, K. (2000) Integrating genetic algorithms and text learning for financial prediction. In A. A. Freitas, W. Hart, N. Krasnogor and J. Smith (eds.), *Data Mining with Evolutionary Algorithms*, pp. 72-75. Technical Report WS-99-06.

[6] White, H., (1988) Economic prediction using neural networks: the case of ibm daily stock returns. In Proceedings of the 2nd Annual IEEE Conference on Neural NetworksII, pp. 451–458