Real-Time Investors' Sentiment Analysis from Newspaper Articles

Konstantinos Arvanitis and Nick Bassiliades

Department of Informatics
Aristotle University of Thessaloniki
Thessaloniki, Greece
{arvanitk, nbassili}@csd.auth.gr

Abstract. Recently, investor sentiment measures have become one of the more widely examined areas in behavioral finance. They are capable of both explaining and forecasting stock returns. The purpose of this paper is to present a method, based on a combination of a naïve Bayes classifier and the n-gram probabilistic language model, which can create a sentiment index for specific stocks and indices of the New York Stock Exchange. An economic useful proxy for investor sentiment is constructed from U.S. news articles mainly provided by The New York Times. Initially, a large amount of articles for ten big companies and indices is collected and processed, in order to be able to extract a sentiment score from each one of them. Then, the classifier is trained from the positive, negative and neutral articles, so that it is possible afterwards to examine the sentiment of any unseen newspaper article, for any company or index. Subsequently, the classification task is tested and validated for its accuracy and efficiency. The widely used Baker and Wurgler sentiment index [2] is used as a comparison measure for predicting stock returns. In a sample of S&P 500 index from 2004 to 2010 on monthly basis, it is shown that the new sentiment index created has, on average, twice the predictive ability of Baker and Wurgler's index, for the existing time frame.

***Keywords*:** Sentiment Analysis, Data Mining, Sentiment Index, Investor Sentiment, Stock Returns, Naïve Bayes Classifier, n-gram Language Model

1 Introduction

News media is a very competitive industry whose main goal is to capture attention. Shiller [21] notes that news play a crucial role in buying or selling decisions among traders, who constantly react to new incoming information. He further argues that the news media are important players in creating market sentiment and similar thinking as it spreads ideas and, thus, can significantly contribute to herding behavior and influence price movement on financial markets. Behavioral finance supplements standard finance by introducing the revolutionary belief that behavior is not 'rational' but 'normal' [24]. If financial markets are not always rational then perhaps investors should take into account the psychology of the market. How this should be achieved has received great

attention in the academic literature during the last decade. Most research tries to construct an index of investor sentiment with the help of various indicators. Baker and Wurgler [2] construct an index of investor sentiment that is based on the common variation in six proxies for sentiment: the closed-end fund discount, share turnover, the number and average first-day returns on IPO's, the equity share in new issues and the dividend premium.

As for ways to measure investor sentiment: there are direct and indirect measures. Direct measures are based on surveys taken from certain groups of people, for instance: global fund managers. Indirect measures are based on market data such as price and volatility. Both have their own merits and drawbacks. Investor sentiment is a much debated topic but it is not yet clear how it should be measured. Current literature attempts to capture investor sentiment by combining multiple imperfect proxies. Such a proxy is for example the market volatility index ('VIX'), which measures the implied volatility of options on the Standard and Poor's 100 stock index and is known as the 'investor fear gauge'. Popular directed measures of sentiment are different sort of confidence indices. In numerous countries and markets there are multiple indices available that try to track consumer or (retail) investor confidence by means of surveys.

Recently, research is focusing more on methods that capture sentiment with the help of media and computational linguistics. A freely available tool called Google Search Volumes is used by [12] to predict stock returns. Changes in volumes of words like "market crash" and "bear market" can predict stock returns while changes in positive search word volumes such as of "bull market" do not. One of the most easy and effective but less sophisticated ways for analyzing text is by means of a Bayesian approach. All that is required are two text files that represent negative and positive words or sentences. Then, a specific text can be classified as negative or positive depending on the similarity with the two basis files. This method was tested by [14] and successfully determined if a movie was regarded good or bad. Major drawback of this method is the fact it only classifies text into positive/negative, it does not tell us the degree of positivity/negativity. Another drawback is that the quality of the basis files determines the quality of the analysis.

In this paper a useful proxy for investor sentiment is constructed with the help of financial news from U.S. newspapers from 2004 to 2014. The construction of the sentiment index follows a variation of a Bayesian approach, combining a naïve Bayes classifier with the n-gram probabilistic language model that is based on Markov chains. The classifier is trained from three highly targeted text lists containing positive, negative and, also, neutral text acquired from the newspaper articles. The main objective is that a sentiment index could be constructed for any ticker of the U.S. Stock Exchange in real-time, in order to help investors classify stocks or measure the overall market sentiment.

In contrast to existing literature, our analysis is much broader given that a sentiment index can be created for any company or index of the stock exchange. The results in many of the previous studies where Twitter is used as data source suffer from noise, since many Tweets are insignificant but affect the overall result. Our sentiment index is created from official news feed and articles of the New York Times and the result is much more factual and clear. In other studies, there is the limit of the research area

which only deals with some indices, while others have the limit of the time frame, which has to be many days or months in contrast with our approach where you can create the sentiment index on a daily, weekly, monthly or annual time frame.

The paper is structured as follows: section 2 provides the appropriate background knowledge to our work, section 3 reports some related work, section 4 presents the data sources and the methodology, section 5 continues with the empirical findings and finally, section 6 concludes.

2 Background

2.1 Framing Effects

The framing effect is an example of cognitive bias, in which people react to a particular choice in different ways depending on how it is presented; e.g. as a loss or as a gain [15]. People tend to avoid risk when a positive frame is presented but seek risks when a negative frame is presented [26].

Framing effects within the news media have been an important research topic among journalism, political science and mass communication scholars. Price et al. argue [17] that the news framing effect has to do with the way events and issues are packaged and presented by journalists to the public. They believe that news frames can fundamentally affect the way readers understand events and issues. Authors suggest that news frames can activate certain ideas, feelings, and values, encourage particular trains of thoughts and lead audience members to arrive at predictable conclusions.

Price and Tewksbury [16] explain the news media framing effect by using the applicability effect in their knowledge activation process model. A framing effect of a news story renders particular thoughts applicable through salient attributes of a message such as its organization, selection of content or thematic structure. The knowledge activation model assumes that at any particular point in time, a mix of particular items of knowledge that are subject to processing (activation) depends on characteristics of a person's established knowledge store. When evaluating situations, people tend to use (activate) ideas and feelings that are most accessible and applicable.

Iyengar [11] examines the impact of news framing on the way people ascribe responsibility for social, political, and economic conditions. He finds that media more often take an episodic rather than a thematic perspective towards the events they cover.

Vliegenthart et al. [27] investigate the effect of two identified news frames, risk and opportunity, on public support regarding the enlargement of the European Union. They find that participants in the opportunity frame condition show significantly higher support compared to participants in the risk condition.

These studies show that framing influences the perception of new information and may be a powerful tool in influencing public opinion and, as a consequence, the public's future actions. Casual observation suggests that the content of news about the stock market could be linked to investor psychology and sociology. However, it is unclear whether the financial news media induces, amplifies, or simply reflects investors' interpretations of stock market performance.

2.2 Investor Sentiment Proxy Construction

Investor sentiment is a much debated topic but it is not yet clear how it should be measured. Current literature attempts to capture investor sentiment by combining multiple imperfect proxies. Such a proxy is for example the market volatility index ('VIX'), which measures the implied volatility of options on the Standard and Poor's 100 stock index and is known as the 'investor fear gauge'. The VIX index is often used as a contrarian indicator in that extreme levels indicate market turning points and is supported by the theory of market over- & under reaction. Popular directed measures of sentiment are different sort of confidence indices. In numerous countries and markets there are multiple indices available that try to track consumer or (retail) investor confidence by means of surveys.

Two widely known indices for U.S. consumer confidence are the Conference Board's Consumer Confidence Index (CCI) and the University of Michigan's Index of Consumer Sentiment. Bram and Ludvigson [5] found the former is better at explaining most categories of consumer spending. Qiu and Welch [18] find that consumer confidence can be a good proxy for investor sentiment and plays a robust role in financial market pricing.

Tumarkin and Whitelaw [25] use the opinions and views of message board users for examining the relationship between sentiment and abnormal stock returns and trading volume. Although investor opinion correlates with abnormal industry-adjusted returns they find no evidence contrary to market efficiency.

Bollen et al. [4] analyze large amounts of tweets (short bursts of inconsequential information) for mood swings. They use two tools that determine mood with the help of computational linguistics: OpinionFinder and Google-Profile of Mood States (GPMOS). OpinionFinder determines positive vs. negative moods and GPMOS measures mood in six dimensions (Calm, Kind, Happy, Vital, Alert and Sure). They claim that the daily closing price of the DJIA can be predicted four days ahead with 87.6% accuracy. However, this cannot be verified because the GPMOS is not made public. OpinionFinder is an open source project of several American Universities and identifies positive/negative words, actions and subjective/objective statements. Its developers claim to accurately classify polarity about 74% of the time.

While OpinionFinder is a clear improvement over a Bayesian method it still lacks the ability to determine the degree of negativity/positivity. A company called OpenAmplify3 claims to have successfully resolved this problem. Although their method is black box we can analyze the input and output of their service. OpenAmplify requires English text files as input and returns output with the help of an application programming interface (API). Their analysis is quite extensive and can be divided into five main categories: topics, actions, styles, demographics and topic intentions analysis. Topic analysis is done on a co-reference basis, meaning that different words can be identified as belonging to the same topic. For instance: 'Jack' and 'Jill' and 'He' and 'She" are connected but also 'Coca Cola Company' and 'CCC' are linked together. Every topic scores a degree of polarity (negativity/positivity) on a scale of -1 to 1 where the former indicates extreme negativity and the latter indicates extreme positivity. Overall text polarity is the weighted average polarity of all separate topics. Weighting is done based

on a relevance score. Topics that are weakly related to all other topics are given a low relevance score and have low impact on overall text polarity. This is possible because OpenAmplify identifies relationships between topics and organizes them into a broad range of domains. Other interesting features are action analysis, contrast (degree of certainty) and temporality (timeframe). Given that OpenAmplify works, having an extensive analysis of a large amount of news articles gives you the possibility to construct a wide variety of (investor) sentiment proxies. For example: irrelevant text can be filtered out by focusing on the domain 'business' with subdomain 'stock market'. The average polarity of the remaining text can be a proxy for investor sentiment.

3 Related Work

Previous research investigates the immediate impact news media might have on the performance of financial markets. For instance, Antweiler and Frank [1] investigate the effect of Internet stock message boards posted on the websites of Yahoo! Finance and Raging Bull on the short-term market performance of 45 U.S. listed companies. They find weak evidence that the number of content messages posted helps to predict stock's intraday volatility but do not find evidence of news media content in-between the content of the Wall Street Journal column Abreast of the Market and the stock market on a daily basis. They, also, find that unusually low or high values of media pessimism predict high trading volume, while low market returns lead to high media pessimism, and conclude that news media content can serve as a proxy for investor sentiment. In a more recent study, Garcia [10] constructs a daily proxy for investor sentiment by taking a fraction of negative and positive words in two columns of financial news, Financial Markets and Topics in Wall Street from the New York Times. He finds evidence of an asymmetric predictive activity of news content on stock returns, especially during recessions. The effect is particularly strong on Mondays and on trading days after holidays, which persists into the afternoon of the trading day.

While some trading in the market brings noise traders with different models who cancel each other out, a substantial percentage of trading strategies are correlated, leading to aggregate demand shifts. As Shleifer and Summers elaborate [22], the reason for this is that the judgmental biases affecting investors in information processing tend to be the same. For example, subjects in psychological experiments tend to make the same mistake; they do not make random mistakes. Indeed, Barber et al. [3] utilize brokerage data and find that individual investors predominantly buy the same stocks as each other contemporaneously, and that this buying pressure drives prices upwards. Similarly, Schmeling [19] employs survey data and finds that individual investor sentiment forecasts stock market returns. In effect, these studies reveal that arbitrageurs are not always successful in bringing prices back in line with fundamentals. Thus, shifts in the demand for stocks that are independent of fundamentals may persist, and thus be observable.

Dickinson and Hu [8] seek to predict a sentiment value for stock related tweets on Twitter, and demonstrate a correlation between this sentiment and the movement of a company's stock price in a real time streaming environment. They use both *n*-gram and

"word2vec" [1] textual representation techniques alongside a random forest classification algorithm to predict the sentiment of tweets. These values are then evaluated for correlation between stock prices and Twitter sentiment for that each company. The results show significant correlations between price and sentiment for several individual companies. Some companies such as Microsoft and Walmart show strong positive correlation, while others such as Goldman Sachs and Cisco Systems show strong negative correlation. This suggests that consumer facing companies are affected differently than other companies.

Das and Chen [6] developed a methodology for extracting small investor sentiment from stock message boards. Their findings showed that five distinct classifier algorithms coupled by a voting scheme are found to perform well against human and statistical benchmarks. Also, they state that time series and cross-sectional aggregation of message information improves the quality of the sentiment index. Their empirical applications evidence a relationship with stock returns, on a visual level, by phase-lag analysis, using pattern recognition and regression methods. Last but not least, they state that sentiment has an idiosyncratic component, and aggregation of sentiment across stocks tracks index returns more strongly than with individual stocks.

Sehgal and Song [20] introduce a novel method to predict sentiment about stock using financial message boards. They state that web financial information is not always reliable and for this reason they propose a new measurement known as TrustValue which takes into account the trustworthiness of an author. In their work, it is shown that TrustValue improves prediction accuracy by filtering irrelevant or noisy sentiments. Sentiment and TrustValue are used together to make the model for stock prediction. They used the intuition that sentiments effect stock performance over short time period and they captured this with Markov model. Their stock prediction results showed that sentiment and stock value are closely related and web sentiment can be used to predict stock behavior with seasonable accuracy.

The linear causality framework is widely adopted in the behavioral finance literature when evaluating the predictive content that sentiment may have upon stock returns. Dergiades [7] finds out that there is reasonable statistical evidence to support that sentiment embodies significant predictive power with respect to stock returns. His study contributes to the understanding of the non-linear causal linkage between investors' sentiment dynamics and stock returns for the US economy, by employing the sentiment index developed by Baker and Wurgler and within a non-linear causality framework.

4 News Articles Classification Methodology and Sources

In this section, we present the sources that were used for this work; the methodology we followed and the processing the data went through. The key concept in this work is to train a classifier which is the most appropriate to classify articles with financial content about companies (as positive, negative or neutral).

[1] https://code.google.com/p/word2vec

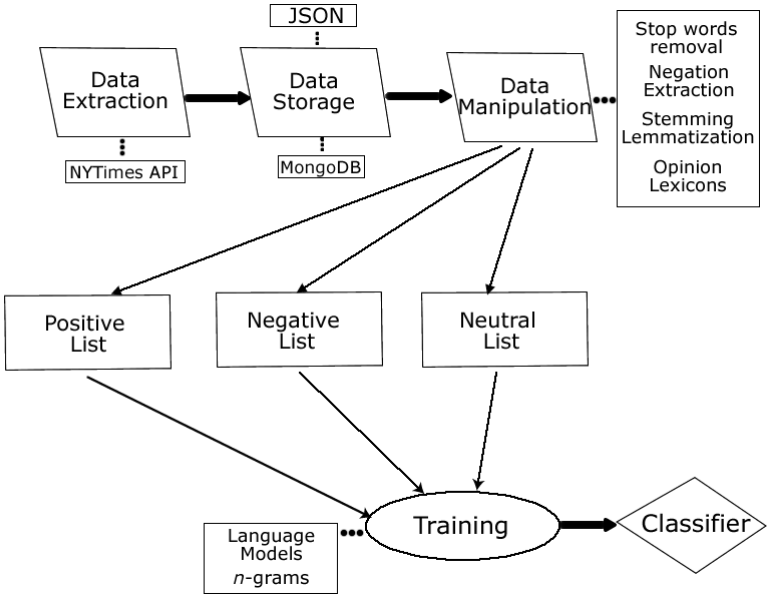


Fig. 1. Architecture and workflow of our methodology

Fig. 1 shows diagrammatically the processes and the methods used for the data extraction, storage and the preprocessing in order to construct the three lists from which the classifier is trained using *n*-gram language models.As previously mentioned, except for positive and negative categories, a text can also be classified as neutral so that the result would be more accurate with reduced noise.

4.1 News Sources and Preprocessing

Newspaper articles used for the analysis were obtained from the New York Times with NYT Article Search API v2[2], which can search articles from 1851 to today, retrieving headlines, abstracts, lead paragraphs and other article metadata.

Search requests follow a standard URI structure whose main parameter is the search query term which is being searched on the article body, headline and byline. The structure of a search request is the following:

```
http://api.nytimes.com/svc/search/v2/articlesearch.re-
sponse-format?[q=search term&fq=filter-field:(filter-
term)&additional-params=values]&api-key=####
```

Besides the search query term, a filtered search query feature is applied, which uses standard Lucene[3] syntax and can specify the fields and the values that a query will be

[2] http://developer.nytimes.com/docs/read/article_search_api_v2
[3] https://lucene.apache.org/

filtered on. Lucene syntax separates the filter field name and value with a colon, and surrounds multiple values with parentheses, like the following:

```
&fq=news_desk:("Sports" "Foreign") AND glocations:("NEW
YORK CITY")
```

In this way, the scope of the search can be narrowed and the articles returned would be more accurate, which in this work involve exclusively business or financial topics.

All articles are returned in JSON[4] format and stored in MongoDB[5] database. News data for each company is stored in a different collection so that all bulk of data is clustered and easy to manipulate. All collections are sorted by ascending order according to the publish date of the articles. New articles for a company are stored in the corresponding collection in chronological order. The initial look of an article is shown in **Fig. 2** which was obtained from New York Times website. The JSON format of the article in **Fig. 2** has the structure shown in **Fig. 3**.

For the opinion lexicons, two positive and negative dictionary files are used [13], which are useful for textual analysis in financial applications. Phrases like 'not good' are converted to '!good' and added at the corresponding dictionary to distinguish negation.

Another widely used feature of natural language processing is used, which has to do with removing the stop words from the examined text, so that the text classification can be applied to only the words that really count and have positive or negative effect to the overall sentiment. The Stop Word Lists used in the analysis [13] are divided in five categories: Generic, Names, Geographic, Currencies, Dates and Numbers. Besides the removal of stop words, the training procedure consists of another feature which is to collapse all the different inflectional forms of a lemma to its base dictionary form, which can be found in the lexicons. Classification was tested in many ways with the stop word lists, like excluding some of them or applying the classification to the text without removing any stop words. The tests showed that the best performance was achieved by removing all the stop words from the text and leaving only words that have sentiment impact.

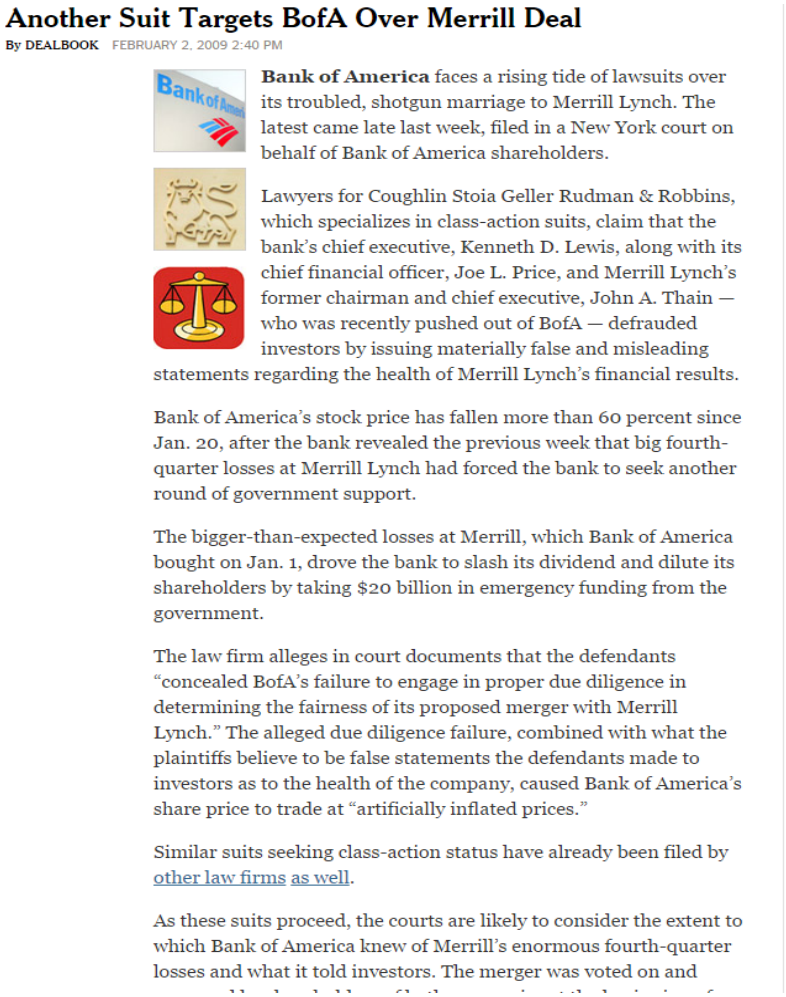
Since the text gets a form which is optimal and easy to extract a safe score, it is passed to the lexicons to count the occurrence of each word in the text that exist in any of the lexicons. If a word in the text belongs to the positive lexicon, the counter of the sentiment score is increased by one and if it belongs to the negative lexicon, the counter is decreased by one. Finally, a sentiment score about the examined text is obtained, which must be used to categorize it as positive, negative or neutral. For accuracy reasons, a threshold is set for sentiment score of higher than 2, for positive, and lower than -2, for negative, and all between them are categorized as neutral.

Three lists are created, one for positive, one for negative and one for neutral articles. For our study, we have 5000 positive articles, 5000 negative articles and 10000 neutral articles. These lists are used to train the classifier, which is a Dynamic Language Model classifier that uses n-gram language models, as explained in the next section. Training

[4] http://json.org/

[5] https://www.mongodb.org/

is based on a multivariate estimator for the category distribution and dynamic language models for the per-category character sequence estimators. It calculates conditional and joint probabilities of each category for the classified object and the classifier returns one best category as result of classification process. Experimental results show that using language models in classification, we are able to obtain better performance than traditional Naïve Bayes classifier.



Another Suit Targets BofA Over Merrill Deal

By DEALBOOK FEBRUARY 2, 2009 2:40 PM

Bank of America faces a rising tide of lawsuits over its troubled, shotgun marriage to Merrill Lynch. The latest came late last week, filed in a New York court on behalf of Bank of America shareholders.

Lawyers for Coughlin Stoia Geller Rudman & Robbins, which specializes in class-action suits, claim that the bank's chief executive, Kenneth D. Lewis, along with its chief financial officer, Joe L. Price, and Merrill Lynch's former chairman and chief executive, John A. Thain — who was recently pushed out of BofA — defrauded investors by issuing materially false and misleading statements regarding the health of Merrill Lynch's financial results.

Bank of America's stock price has fallen more than 60 percent since Jan. 20, after the bank revealed the previous week that big fourth-quarter losses at Merrill Lynch had forced the bank to seek another round of government support.

The bigger-than-expected losses at Merrill, which Bank of America bought on Jan. 1, drove the bank to slash its dividend and dilute its shareholders by taking $20 billion in emergency funding from the government.

The law firm alleges in court documents that the defendants "concealed BofA's failure to engage in proper due diligence in determining the fairness of its proposed merger with Merrill Lynch." The alleged due diligence failure, combined with what the plaintiffs believe to be false statements the defendants made to investors as to the health of the company, caused Bank of America's share price to trade at "artificially inflated prices."

Similar suits seeking class-action status have already been filed by other law firms as well.

As these suits proceed, the courts are likely to consider the extent to which Bank of America knew of Merrill's enormous fourth-quarter losses and what it told investors. The merger was voted on and

Fig. 2. Example of a New York Times article

```
{
 "response": {
   "meta": {
     "hits": 20,
     "time": 385,
     "offset": 0   },
   "docs": [
     {
       "web_url": "http:\/\/dealbook.nytimes.com...",
       "snippet": "Bank of America faces ...",
       "lead_paragraph": "Bank of America faces ...",
       "abstract": "Bank of America faces [...]",
       "print_page": null,
       "blog": [       ],
       "source": "The New York Times",
       "multimedia": [       ],
       "headline": {
         "main": "Another Suit Targets ...",
         "kicker": "DealBook"       },
       "keywords": [
         {
           "rank": "1",
           "name": "type_of_material",
           "value": "News"         }
       ],
       "pub_date": "2009-02-02T14:40:29Z",
       "document_type": "blogpost",
       "news_desk": null,
       "section_name": "Business Day",
       "subsection_name": null,
       "byline": {
         "person": [
           {
             "organization": "",
             "role": "reported",
             "rank": 1           }
         ],
         "original": "By DEALBOOK"
       },
       "type_of_material": "Blog",
       "_id": "4fd394388eb7c8105d8c8fdd",
       "word_count": 512     }
   ]
 },
 "status": "OK",
 "copyright": "Copyright (c) 2013..."
}
```


Fig. 3. JSON format of a New York Times article.

4.2 Classification Methodology

At this point, we describe the methods adopted for the classification of the news articles and the validation check of the methodology. The classification procedure uses *n*-gram language models and it is considered as an extension of the traditional Naive Bayes classifier, with the difference that the Laplace smoothing is replaced by some more sophisticated smoothing methods. A naïve Bayes classifier assumes that the value of a particular feature is unrelated to the presence or absence of any other feature, given the class variable. For instance, a vehicle may be considered to be a bike if it has two wheels, no doors, and has about 150cm wheelbase. A naïve Bayes classifier considers each of these features to contribute independently to the probability that this vehicle is a bike, regardless of the presence or absence of the other features. Experimental results show that using a language model, we are able to obtain better performance than traditional Naïve Bayes classifier. Language models have been successfully applied in many application areas such as speech recognition and statistical natural language processing.

In recent years, it is confirmed that they are also an attractive approach for Information Retrieval (IR) such as the query likelihood model, because they can provide effectiveness comparable to the best state of the art systems. As a result of this fact, language models are used to other IR-related tasks, such as tracking, topic detection and classification. In this work, an attempt is being made to use language models in text classification, specifically from newspaper articles with financial content about companies, evaluate the accuracy of the method and compare the new sentiment index created with another widely used index.

Language modeling aims to predict the probability of natural word sequences. More simply, the goal is to put high probability on word sequences that actually occur and low probability on the ones that never occur. Given a word sequence $w_1 w_2 \ldots w_T$ to be used as a test corpus, the quality of a language model can be measured by the empirical perplexity (or entropy) on this corpus:

$$Perplexity = \sqrt[T]{\frac{1}{P(w_1 \ldots w_T)}} \tag{1}$$

$$Entropy = \log_2(Perplexity) \tag{2}$$

The main objective is to obtain a small perplexity. The simplest and most successful basis for language modeling is the *n*-gram model: Note that by the chain rule of probability we can write the probability of any sequence as

$$P(w_1 w_2 \ldots w_T) = \prod_{i=1}^{T} P(w_i \mid w_1 \ldots w_{i-1})$$

An *n*-gram model approximates this probability by assuming that the only words relevant to predicting $P(w_i \mid w_1 \ldots w_{i-1})$ are the previous $n-1$ words; that is, it assumes the Markov *n*-gram independence assumption

$$P(w_i \mid w_1 \ldots w_{i-1}) \quad = \quad P(w_{i-n+1} \mid w_1 \ldots w_{i-1})$$

A straightforward maximum likelihood estimate of *n*-gram probabilities from a corpus is given by the observed frequency

$$P(w_{i-n+1} \mid w_1 \ldots w_{i-1}) \quad = \quad \frac{\#(w_{i-n+1} \ldots w_i)}{\#(w_{i-n+1} \ldots w_{i-1})}$$

where #(.) is the number of occurrences of a specified gram in the training corpus. Unfortunately, using grams of length up to *n* entails estimating the probability of W^n events, where W is the size of the word vocabulary. This fact makes it necessary to choose a relatively smaller *n* (beyond 2 to 7). In addition, it is likely to encounter novel *n*-grams that were never witnessed during training, because of the heavy tailed nature of language (i.e. Zipf's law). Therefore, a mechanism for assigning non-zero probability to novel *n*-grams is needed. One standard approach to cope with potentially missing *n*-grams is to use some sort of back-off estimator, which is relatively simple and has the following form:

$$P(w_i \mid w_{i-n+1} \ldots w_{i-1}) = \begin{cases} \hat{P}(w_i \mid w_{i-n+1} \ldots w_{i-1}), & if \#(w_{i-n+1} \ldots w_i) > 0 \\ \beta(w_{i-n+1} \ldots w_{i-1}) \quad X \quad P(w_i \mid w_{i-n+2} \ldots w_{i-1}), & otherwise \end{cases}$$

where

$$\hat{P}(w_i \mid w_{i-n+1} \ldots w_{i-1}) = \frac{discount\#(w_{i-n+1} \ldots w_i)}{\#(w_{i-n+1} \ldots w_{i-1})} \tag{3}$$

is the discounted probability, and $\beta(w_{i-n+1} \ldots w_{i-1})$ is a normalization constant calculated to be

$$\beta(w_{i-n+1} \ldots w_{i-1}) = \frac{1 - \sum_{x:\#(w_{i-n+1} \ldots w_{i-1} x)>0} \hat{P}(x \mid w_{i-n+1} \ldots w_{i-1})}{1 - \sum_{x:\#(w_{i-n+1} \ldots w_{i-1} x)>0} \hat{P}(x \mid w_{i-n+2} \ldots w_{i-1})}$$

An *n*-gram is first matched against the language model to see if it has been observed in the training corpus. If that fails, the *n*-gram is then reduced to an $n-1$-gram by shortening the context by one word. The discounted probability (Equ. 3) can then be

computed using different smoothing approaches. Smoothing techniques are analyzed further below.

Text classifiers, like dynamic language model classifiers, attempt to identify attributes which distinguish documents in different categories. Vocabulary terms, local *n*-grams, word average length, or global syntactic and semantic properties may be such attributes. Also, Language models provide another natural avenue to constructing text classifiers as they attempt to capture such regularities. An *n*-gram language model can be applied to text classification in a similar manner to a naive Bayes model. That is, we categorize a document according to

$$c^* = argmax\{P(c \mid d)\}$$

Using Bayes rule, this can be rewritten as

$$\begin{aligned} c^* &= argmax\{P(c)P(d \mid c)\} \\ &= argmax\{P(c)\prod_{i=1}^{T} P(w_i \mid w_{i-n+1}...w_{i-1}, c)\} \\ &= argmax\{P(c)\prod_{i=1}^{T} P_c(w_i \mid w_{i-n+1}...w_{i-1})\} \end{aligned} \tag{4}$$

Here, $P(d|c)$ is the likelihood of d under category c, which can be computed by an *n*-gram language model. Likelihood is related to perplexity and entropy by Equ. (1) and Equ. (2). $P_c(w_i|w_{i-n+1}...w_{i-1})$ is computed using back-off language models which are learned separately for each category by training on a data set from that category. Then, to categorize a new document d, the document is supplied to each language model, the likelihood (or entropy) of d under the model is evaluated, and the winning category is picked according to Equ. (4).

The *n*-gram, which is a subsequence of length *n* of the items given, has a certain size that needs to be set for the Language Model classifier algorithm. The Language Model rule is to classify a newly given document based on prediction occurring *n*-grams. The algorithm uses a word based *n*-gram to classify articles so an appropriate size should be the average length of a sentence.

If we take into account that the traditional naïve Bayes classifier is a unigram classifier with Laplace smoothing, then it is obvious that *n*-gram classifiers are in fact a straightforward generalization of naive Bayes. However, *n*-gram language models possess many advantages over naive Bayes classifiers, for larger *n*, including modelling longer context and exploiting better smoothing techniques in the presence of sparse data. Another notable advantage of the language modelling based approach is that it does not incorporate an explicit feature selection procedure. For naïve Bayes text classifiers, features are the words, which are considered independent of each other given the category. Instead, Language Model classifiers consider all possible *n*-grams as features. Their importance is implicitly considered by their contribution to the quality of language modelling. The over-fitting problems associated with the subsequent feature

explosion are nicely handled by applying smoothing techniques like Laplace smoothing.

Two general formulations are used in smoothing: back-off and interpolation. Both smoothing methods can be expressed in the following general form:

$$P(w \mid c_i) = \begin{matrix} P_s(w \mid c_i) & w \text{ is seen in } c_i \\ a_{c_i} P_u(w \mid C), & w \text{ is unseen in } c_i \end{matrix}$$

This form shows that for a class c_i, one estimate is made for the words seen in the class, and another estimate is made for the unseen words. In the second case, the estimate for unseen words is based on the entire collection, i.e., the collection model. The zero-probability problem is solved by incorporating the collection model, which also generates the same effect as the IDF factor [23] that is commonly used in IR [9].

The accuracy of the classification is estimated by applying a popular method in machine learning, called *k*-fold cross-validation. Estimating the accuracy of a classifier induced by supervised learning algorithms is important not only to predict its future prediction accuracy, but also for choosing a classifier from a given set (model selection), or combining classifiers [28]. An estimation method with low bias and low variance is the best fit to estimate the final accuracy of a classifier.

A classifier is a function that maps an unlabeled instance to a label using internal data structures. An inducer, or an induction algorithm, builds a classifier from a given dataset. Let *V* be the space of unlabeled instances and *Y* the set of possible labels. Let *X = V x Y* be the space of labeled instances and $D = \{x_1, x_2, ..., x_n\}$ be a dataset (possibly a multiset) consisting of n labeled instances, where $x_i = \langle u_i \in V, y_i \in Y \rangle$. A classifier *C* maps an unlabeled instance $\upsilon \in$ V to a label y $\in$ Y and an inducer *I* maps a given dataset *D* into a classifier *C*. The notation $I(D,\upsilon)$ will denote the label assigned to an unlabeled instance υ by the classifier built by inducer *I* on dataset *D*, *i.e.*, $I(D,\upsilon) = (I(D))(\upsilon)$.

The **accuracy** of a classifier *C* is the probability of correctly classifying a randomly selected instance, *i.e.*, $acc = Pr(C(\upsilon) = y)$ for a randomly selected instance $\langle u, y \rangle \in X$, where the probability distribution over the instance space is the same as the distribution that was used to select instances for the inducer's training set. Given a nite dataset, the future performance of a classifier induced must be estimated by the given inducer and dataset. A single accuracy estimate is usually meaningless without a confidence interval, so such an interval should be approximated when possible. Also, in order to identify weaknesses the cases where the estimates fail should be identified.

In *k*-fold cross-validation, sometimes called rotation estimation, the dataset *D* is randomly split into *k* mutually exclusive subsets (the folds) $D_1, D_2, ..., D_k$ of approximately equal size. The inducer is trained and tested *k* times; each time t $\in$ {1, 2, ..., k}, it is trained on D/D_t and tested on D_t. The cross-validation estimate of accuracy is

the overall number of correct classifications, divided by the number of instances in the dataset. Formally, let $D_{(i)}$ be the test set that includes instance $x_i = \langle v_i, y_i \rangle$, then the cross-validation estimate of accuracy

$$acc_{cv} = \frac{1}{n} \sum_{\{u_i, y_i\} \in D} \delta(I(D / D_{(i)}, u_i), y_i)$$

The cross-validation estimate is a random number that depends on the division into folds. In cross-validation, it is useful to obtain an estimate for many performance indicators such as accuracy, precision, recall, or F-score. In most cases, the accuracy of a classifier is estimated in a supervised-learning environment. In such a setting, there is a certain amount of labeled data and the goal is to predict how well a certain classifier would perform if this data is used to train the classifier and subsequently ask it to label unseen data. In 10-fold cross-validation, the 90% of the data is repeatedly used to build a model and the remaining 10% to test its accuracy. The average accuracy of the repeats is an underestimate for the true accuracy. Generally, this estimate is reliable, especially if the amount of labeled data is large enough and if the unseen data follows the same distribution as the labeled examples.

5 Results and Discussion

For classification tasks, the terms **true positives, true negatives, false positives** and **false negatives** (also Type I and Type II errors) compare the results of the classifier under test with trusted external judgments. The terms *positive* and *negative* refer to the classifier's prediction (sometimes known as the *expectation*), and the terms *true* and *false* refer to whether that prediction corresponds to the external judgment (sometimes known as the *observation*).

Accuracy is the overall correctness of the model and is calculated as the sum of correct classifications divided by the total number of classifications.

$$Accuracy = \frac{tp + tn}{tp + tn + fp + fn}$$

Precision is a measure of the accuracy provided that a specific class has been predicted. It is defined by:

$$Precision = \frac{tp}{tp + fp}$$

where *tp* and *fp* are the numbers of true positive and false positive predictions for the considered class.

Recall is a measure of the ability of a prediction model to select instances of a certain class from a data set. It is commonly also called sensitivity, and corresponds to the true positive rate. It is defined by the formula:

$$Recall = Sensitivity = \frac{tp}{tp + fn}$$

where *tp* and *fn* are the numbers of true positive and false negative predictions for the considered class. $tp + fn$ is the total number of test examples of the considered class.

F-measure or **balanced F-score** is the harmonic mean of precision and recall.

$$F = 2 * \frac{precision * recall}{precision + recall}$$

While error rate or accuracy dominates much of the classification literature, F-measure is the most popular metric in the text classification and information retrieval communities. The reason is that typical text mining corpora have many classes and suffer from high class imbalance. Accuracy tends to undervalue how well classifiers are doing on smaller classes, whereas F-measure balances *precision* and *recall.*

After obtaining the 10 classifiers created by the 10-fold cross-validation on the training newspaper data, each one of them is evaluated at the corresponding test data set and the performance indicators are recorded which are shown in **Table 1**.

Table 1. Performance Indicators

	TP rate (recall)	**FP rate**	**Accuracy**	**Precision**	**F-Score**
S&P 500	0.79	0.15	0.82	0.84	0.81
Dow Jones	0.88	0.06	0.92	0.90	0.89
Google	0.97	0.42	0.87	0.88	0.92
Bank of America	0.97	0.06	0.90	0.90	0.88
Apple	0.97	0.35	0.89	0.89	0.93
Ebay	0.96	0.31	0.90	0.91	0.93
Nike	0.97	0.49	0.90	0.92	0.94
Citigroup	0.84	0.03	0.91	0.94	0.89
Amazon	0.98	0.45	0.91	0.91	0.95
Microsoft	0.93	0.30	0.86	0.87	0.90
Average	0.93	0.26	0.89	0.90	0.91

As **Table 1** shows, the average scores that the classifier can achieve are 0.93 for recall, 0.89 for accuracy, 0.90 for precision and 0.91 for F-Score. Another noticeable thing is that the variance for those four measures is 0.004, 0.00082, 0.00072, 0.00148

respectively and the standard deviation is as low as 0.06 for recall, 0.029 for accuracy, 0.027 for precision and 0.039 for F-Score, which means that the classification procedure performs quite well in all cases, regardless of the index or the company. In addition, most of the scores are close except for those that correspond to the news articles for the main index S&P 500, which has approximately 4-5 times bigger data size than most of the other company tickers. For the Dow Jones Industrial Index, the classification performs quite well despite its big size.

In order to have a benchmark measure, investor sentiment data is used which was provided by Baker and Wurgler [2]. Baker and Wurgler created a sentiment index, which was updated in May 16, 2011, based on first principal component of six (standardized) sentiment proxies over 1962-2005 data, where each of the proxies has first been orthogonalized with respect to a set of macroeconomic conditions.

At first, chronological line-charts of sentiment analysis are created, for both annual and monthly time frames, for all the companies and indices that news data was collected. Afterwards, two charts of Baker and Wurgler are exported for annual and monthly time frames for the S&P 500 index from 2004 to 2010, and one more monthly chart with the closing prices of the index and all the charts are compared respectively. Furthermore, the new monthly sentiment line-chart of the S&P 500 index is compared with the one that contains the closing prices (Fig. 4).

Fig. 4. Comparison chart of closing prices and new sentiment index

Fig. 4 reveals the possible co-movement of the new sentiment index and the closing prices in monthly basis, as it is obvious that the blue line containing the closing prices follows the curve of the red line which is the sentiment index, sometime later than it.

The next step is to examine the new sentiment index on its ability to explain returns. This is done by applying a regression model. We examine the results provided by regressing the sentiment index on monthly data for the S&P 500 index for the period of

January 2004 to December 20106. In addition, we run the same specification using the sentiment index created by Baker and Wurgler for the same period. The first step of the regression model is to create an equation which involves three variables: Returns, which is the dependent variable, and the Sentiment index and the P/E Ratio which are the regressors. P/E Ratio is used here as a variable related to the fundamentals of the index. The results in **Table 2** show that the new sentiment index is very significant as its p-value is lower than 0.05 and very close to zero. R-squared is 0.152, which means that 15.2% of variation in the dependent variable, which is the returns, can be explained by the new sentiment index and P/E ratio jointly.

Table 2. Results of regression model for the new Sentiment Index

Variable	**Coefficient**	**Probability**
New Sentiment Index	0.000896	0.0008
P_E Ratio	0.000322	0.0683
C	-0.001405	0.8398
R-squared	0.152172	

On the other hand, the results in **Table 3** show that Baker and Wurgler's sentiment index is marginally significant as it is close to 0.05. R-squared is 0.074, so only 7.4% of variation in the returns can be explained by the sentiment index and P/E ratio jointly.

Table 3. Results of regression model for Baker and Wurgler's Sentiment Index

Variable	**Coefficient**	**Probability**
Baker and Wurgler Index	-0.052797	0.0428
P_E Ratio	9.09E-05	0.6540
C	-0.001421	0.8482
R-squared	0.074445	

The final step is to apply simple rolling regression. The window size is set to 60, which is the months, in order to have a rolling 5-year time frame of the sentiment indices and the step size to 1 and we store the P-values and the R-squareds, so that we can then make the comparison graph with the ones from Baker and Wurgler's index.

Fig. 5 shows the rolling p-values for the new sentiment index and Baker and Wurgler's sentiment index. As we can see the red p-values of the new index are almost every time close to zero, while the blue ones of Baker and Wurgler's index are a lot higher. This means that the new sentiment index is most of the time very significant for the equation and in any case, more significant than Baker and Wurgler's index.

In the following graph (**Fig. 6**) the rolling R-squareds for the new sentiment index and Baker and Wurgler's sentiment index are presented. In all cases the red one is above the blue which means it can predict better the future returns. The red has a peak at ~33% while the blue at ~28% and approximately the mean R-squared of the red is 20% while

[6] https://research.stlouisfed.org/fred2/series/SP500/downloaddata

the blue has 10%, the half of the red. This fact shows in simple terms, that on average the new sentiment index has twice the predictive ability of Baker and Wurgler's index.

Summarizing, it is obvious from the results that our new sentiment index created from the classification procedure, outperforms the sentiment index created by Baker and Wurgler, for the timeframe examined.

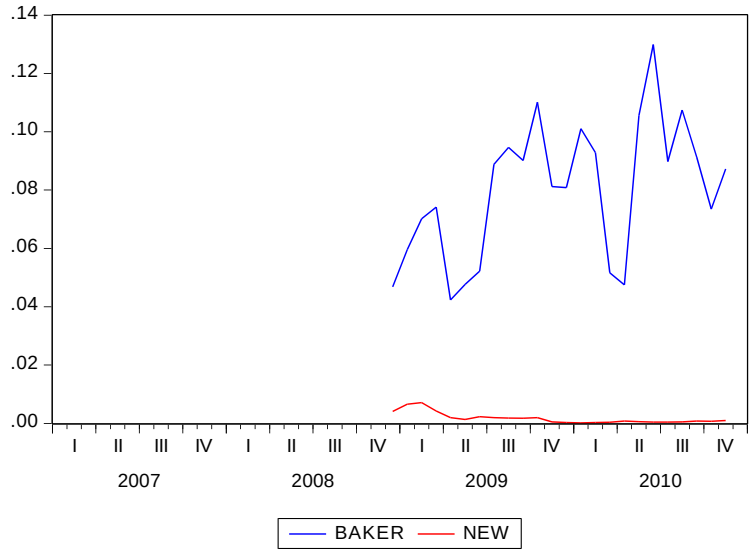


Fig. 5. Comparison chart of rolling p-values of the two indices

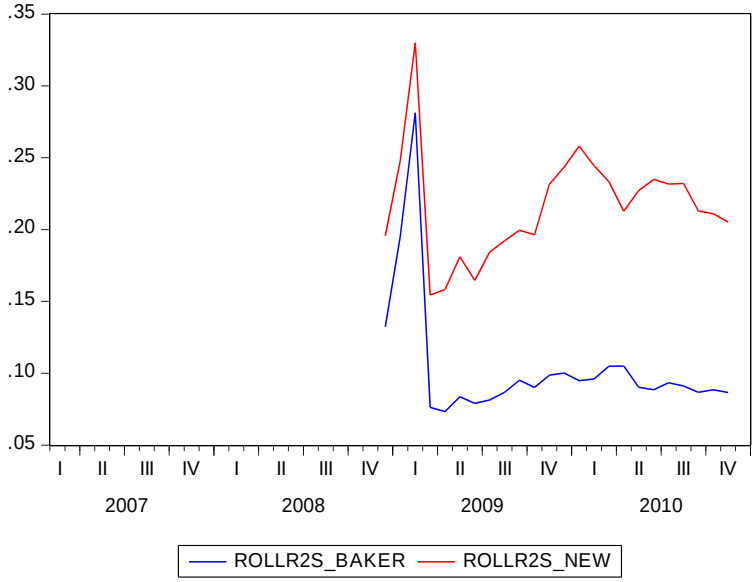


Fig. 6. Comparison chart of rolling R-squared of the two indices

6 Conclusions and Future Work

In the recent years, investor sentiment has become less of an abstract idea and more of a precise measure helpful in both explaining and forecasting stock returns. This paper proposed a new and direct measure of investor's sentiment using newspaper articles, mainly provided by The New York Times. The sentiment index is created using a hybrid method that combines a naïve Bayes classifier and the n-gram probabilistic language model.

First, a large amount of data for ten big companies and indices, which are being traded in the stock exchange, is collected from The New York Times web interface and stored in the NoSQL database MongoDB. Secondly, all articles downloaded are processed and manipulated in order to transform into a word sequence, which has reduced noise and is capable of being passed to the dictionaries and match any positive and negative word occurrences, so that a sentiment score can be extracted. After all articles get a score, three lists are created that contain neutral, most positive and most negative articles, and which will be used to train the classifier.

Once the classifier is created, we can pass unseen newspaper articles to it, in order to classify them as positive, negative or neutral and create a sentiment index for the company or index examined. The tool developed in this work is able to extract a sentiment score for daily, monthly and annual time frames, so it can match most of investors' trading strategies. It can also be extended to any company or index one might be interested in and for any time period.

The experiments performed in this work were based on 10-fold cross-validation, which is a resampling validation technique for assessing how the results of a statistical analysis will generalize to an independent new data set. With the cross-validation procedure, the average performance of the classification task is recorded and the misclassification error is measured. Besides performance estimation, other experiments deal with detecting the forecasting property of the new sentiment index created by the sentiment analysis for a company or index.

In a sample of S & P 500 index from 2004 to 2014 on monthly basis, it is shown that newspaper articles are correlated with the closing prices and the returns. In addition, the new sentiment index created is compared with the sentiment index created by Baker and Wurgler, and it is proven that for the existing time frame, the new index outperforms Baker and Wurgler's index, in terms of predicting returns.

Future research could extend the new index and review its accuracy for future returns. Also, it could be interesting for future work to expand the index backwards to earlier dates and review its forecasting ability and, also, compare it with Baker and Wurgler's index. Furthermore, a good and useful idea would be to store newspaper articles for other companies and indices one may be interested in and apply the tool created in this work to test its effectiveness for both older data and, also, for real-time data to examine the performance on news that pop up instantaneously.

7 Acknowledgments

We would like to give special thanks to Dr. Theologos Dergiades, Academic Associate of the School of Science & Technology at the International Hellenic University, for his advice and assistance whenever it was needed.

8 References

1. Antweiler, W., and M. Frank. (2004). Is all that talk just noise? The information content of Internet stock message boards. Journal of Finance 59(3): 1259-1294.
2. Baker, M., and J. Wurgler. (2006). Investor sentiment and the cross-section of stock returns. Journal of Finance 61(4): 1645-1680.
3. Barber, B. M., Odean, T., Zhu, N., (2009). Do retail trades move markets? Review of Financial Studies 22, 151–186.
4. Bollen, J., H. Mao and X. Zeng, (2010), "Twitter mood predicts the stock market." Journal of Computational Science, 2(1), 1-8.
5. Bram, J. and S.C. Ludvigson, (1998), "Does Consumer Confidence Forecast Household Expenditure? A Sentiment Index Horse Race", Economic Policy Review, Vol.4, No.2.
6. Das S, Chen M (2007) Yahoo! for Amazon: sentiment extraction from small talk on the Web. Manag Sci 53(9):1375–1388.
7. Dergiades, T. (2012). Do investors' sentiment dynamics affect stock returns? Evidence from the US economy, Economics Letters, Volume 116, Issue 3, Pages 404-407, ISSN 0165-1765, http://dx.doi.org/10.1016/j.econlet.2012.04.018.
8. Dickinson, B. and Hu, W. (2015) Sentiment Analysis of Investor Opinions on Twitter. Social Networking, 4, 62-71.
9. Fang H., Tao T., and Zhai Ch. X. (2004) A formal study of information retrieval heuristics. In Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '04). ACM, New York, NY, USA, 49-56. DOI=http://dx.doi.org/10.1145/1008992.1009004
10. Garcia, D. (2013). Sentiment during recessions. Journal of Finance 68(3): 1267-1299.
11. Iyengar, S. 1991. Is Anyone Responsible? How Television Frames Political Issues. Chicago: University of Chicago Press.
12. Klemola, A., J. Nikkinen and J. Peltomäki, (2010), "Investor Sentiment in the Stock Market Inferred From Google Search Volumes."
13. McDonald B. (2013) "Bill McDonald's Word Lists Page", University of Notre Dame. Available at: http://www3.nd.edu/~mcdonald/Word_Lists.html
14. Pang, B. and L Lee (2008), "Opinion Mining and Sentiment Analysis"
15. Plous, Scott (1993). The psychology of judgment and decision making. McGraw-Hill. ISBN 978-0-07-050477-6.
16. Price, V., & Tewksbury, D. (1997). News values and public opinion: A theoretical account of media priming and framing. In G. A. Barett & F. J. Boster (Eds.), Progress in communication sciences: Advances in persuasion (Vol. 13, pp. 173–212). Greenwich, CT: Ablex.
17. Price, V., D. Tewksbury, and E. Powers. (1997). Switching trains of thought: The impact of news frames on readers' cognitive responses. Communication Research 24(5): 481-506.
18. Qiu, L. and I. Welch, (2006), "Investor Sentiment Measures", Brown University and NBER.
19. Schmeling, M., (2007). Institutional and individual sentiment: Smart money and noise trader risk? 23, 127–145.

20. Sehgal, V. and C. Song, “SOPS: Stock Prediction using Web Sentiment”, Seventh IEEE International Conference on Data Mining – Workshops, 2009, pp.21-26.
21. Shiller, R. J. (2005). Irrational Exuberance. Second Edition. Princeton, New Jersey: Princeton University Press.
22. Shleifer, A., Summers, L. H., (1990). The noise trader approach to finance. Journal of Economic Perspectives 4, 19–33.
23. Spärck Jones, K. (1972). "A Statistical Interpretation of Term Specificity and Its Application in Retrieval". Journal of Documentation 28: 11–21. doi:10.1108/eb026526.
24. Statman, M., (2005), "Normal Investors, Then and Now", CFA Institue, Financial Analysts Journal, Vol. 61 No. 2, p. 31-37.
25. Tumarkin, R., R.F. Whitelaw, (2000), “News or Noise? Internet Message Board Activity and Stock Prices”.
26. Tversky, A.; Kahneman, D. (1981). "The Framing of decisions and the psychology of choice". Science 211 (4481): 453–458. doi:10.1126/science.7455683.
27. Vliegenthart, R., A. R. T. Schuck, H. G. Boomgaarden, and C. H. De Vreese. 2008. News Coverage and Support for European Integration, 1990-2006. International Journal of Pulblic Opinion Research 20(4): 415-436.
28. Wolpert, D. H., (1992). “Stacked Generalization”, Neural Networks, 5, 241.