Financial News Analysis for Intelligent Portfolio Management

Young-Woo Seo Joseph Giampapa Katia Sycara

CMU-RI-TR-04-04

January 2004

Robotics Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
JAN 2004		**00-00-2004 to 00-00-2004**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Financial News Analysis for Intelligent Portfolio Management	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Carnegie Mellon University,Robotics Institute,Pittsurgh,PA,15213**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

In this paper, we presentWarren, a multi-agent system for intelligent portfoliomanagement which is motivated by the great benefits of working in teams within the domain of Distributed Artificial Intelligence (DAI) and TextMiner which takes advantage of information retrieval techniques to complement quantitative financial information. In the portfolio management domain, software agents that evaluate the risks associated with the individual companies in a portfolio should be able to read news articles that indicate the financial outlook of a company. There is a positive correlation between news reports on a company?s financial outlook and its attractiveness as an investment. Since it is impossible for financial analysts or investors to track and read each one, it would be very helpful to have a technology for automatically analyzing news reports that reflect positively or negatively on a company?s financial outlook. It is also necessary for an agent to learn contextual changes in the news reports autonomously. To accomplish these tasks, we devised a new text classification method and a sampling method. With comprehensive quantitative information gathered by efficient coordinations between agents, and the supplementing of quantitative information by financial news analysis, we showed a successful application of amulti-agent system for portfolio management.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**23**	

Abstract

In this paper, we present Warren, a multi-agent system for intelligent portfolio management, which is motivated by the great benefits of working in teams within the domain of Distributed Artificial Intelligence (DAI) and TextMiner which takes advantage of information retrieval techniques to complement quantitative financial information. In the portfolio management domain, software agents that evaluate the risks associated with the individual companies in a portfolio should be able to read news articles that indicate the financial outlook of a company. There is a positive correlation between news reports on a company's financial outlook and its attractiveness as an investment. Since it is impossible for financial analysts or investors to track and read each one, it would be very helpful to have a technology for automatically analyzing news reports that reflect positively or negatively on a company's financial outlook. It is also necessary for an agent to learn contextual changes in the news reports autonomously. To accomplish these tasks, we devised a new text classification method and a sampling method. With comprehensive quantitative information gathered by efficient coordinations between agents, and the supplementing of quantitative information by financial news analysis, we showed a successful application of a multi-agent system for portfolio management.

Contents

1 Introduction

An important premise in financial investing is that there must be a reasonable amount of validated information before a security is considered from an investment standpoint [6]. Given the requirements of having various expertise and the difficulties in locating and evaluating information sources, financial portfolio management has to date carried out by investment firms that employ teams of specialists for finding, filtering and evaluating relevant information. It has primarily focused on the portfolio selection process (i.e., asset allocation) as opposed to portfolio monitoring – the ongoing, continuous, daily provision of an up-to-date financial picture of an existing portfolio [14].

In portfolio management, it is important for an investor to monitor his or her portfolio regularly in addition to asset allocation, because it must be determined whether or not the return results of the portfolio meet the expectations of the investor, or whether there is a need to change the strategic asset allocation. The monitoring process also provides comprehensive, detailed information on the investment positions of the investor. The result of the controlling monitoring might require changes in the asset allocation in order to realign the long-term asset allocation strategy. It is important to note that portfolio management, as an investment process, is not a static, but a dynamic one, where you should regularly adapt your decisions to changes in the market and in your own circumstances [6].

In the application domain of portfolio management, a large volume of information exists about a company and its financial performance that humans must effectively attend to and manage in order to make decisions. To address this problem, we proposed and implemented a multi-agent system, called *Warren*[1] [5], [13]. Warren is composed of several agents that help the user manage his or her portfolio by providing quantitative information: stock price, performance history, earnings summaries and risk (β value), and to proactively advise the user whenever the portfolio may be too risky for the user's specified tolerance to risk.

In addition to such quantitative information, it is desirable to look into qualitative data such as financial news reports, in order to get multiple perspectives on the financial performance of the company of interest, because there is a positive correlation between news reports on a company's financial outlook and its attractiveness as an investment. However, because of the tremendous volume of such reports, it is impossible for financial analysts or investors to track and read each one. Therefore, it would be very helpful to have a technology for automatically analyzing news reports that reflect positively or negatively on a company's financial outlook. To accomplish this task, we devised and implemented a new agent, called TextMiner, which performs the tasks of information retrieval for news from on-line news providers such as Reuters, CNN Financial Network, Business Wire, Forbes.com and others.

The goal of TextMiner is to provide an accounting of news articles on the company of interest for a period, in terms of good or bad financial performance. As a software agent in Warren, the TextMiner agent, upon a request from the user or other agents, selectively attends to news reports on the company of interest by filtering non-financial

[1]The system is named after Warren Buffet, a famous American investor and author about investment strategies.

news out and then classifying them in terms of the company's current financial status.

We devised a new text classification method that helps TextMiner carry out its classification task. The devised method predicts the class of a financial news article through the voting process among experts, which are frequently co-located phrases. A co-located phrase is a sequence of nearby but not necessarily consecutive words. In addition, it is important for an agent to learn the content-shift autonomously, because the vocabularies of text domains change slightly from time to time, and the intervention by humans in order to label text data is quite expensive. The devised method for providing TextMiner with self-learning capability estimates the class of unlabeled data on the basis of the learner's confidence, which is obtained through the training phase.

In this paper, we present Warren, which is a multi-agent system for intelligent portfolio management, motivated by the great benefits of working in teams within the domain of Distributed Artificial Intelligence (DAI), and TextMiner, which is a text classification agent that takes advantage of information retrieval techniques to complement quantitative financial information.

The paper is organized as follows. Section 2 details characteristics of portfolio management domain and our previous approaches to this domain. Section 3 describes text analysis for augmenting management of portfolio. Section 4 takes an example of intelligent portfolio management. Section 5 discuss the results and future works.

2 Portfolio Management Domain

Traditionally, the purpose of portfolio management, as stated by modern portfolio theory [10], is to provide the best possible rate of return for a specified level of task, or conversely, to achieve a specified rate of return with the lowest possible risk. Risk here means the probability that the actual return on an investment will be less than the expected return. Usually, there is a strong correlation between risk and return, namely the higher the risk, the higher the return [6].

Given requirements of various expertise and difficulties in locating and evaluating information source, financial portfolio management has to date carried out by investment firms that employ teams of specialists for finding, filtering and evaluating relevant information. It has primarily focused on the portfolio selection process (i.e., asset allocation) as opposed to portfolio monitoring – the ongoing, continuous, daily provision of an up-to-date financial picture of an existing portfolio [14].

In portfolio management, it is important for an investor to monitor his portfolio regularly, in addition to asset allocation, because it must be determined whether or not the return results of the portfolio meet the expectations of the investor whether or not there is a need to change the strategic asset allocation. The monitoring process also provides comprehensive, detailed information on the investment positions of the investor. The result of the controlling monitoring might require changes in the asset allocation in order to realign to the long-term asset allocation strategy. It is also important to note that portfolio management, as an investment process, is not a static, but a dynamic process, where one should regularly adapt one's decisions to changes in the market and in one's own circumstances [6].

In contrast to past environment of portfolio management, with the rapid progress

of computer technology in recent years, it is rather easy to access financial markets and information sources over the Internet. In addition, intelligent agent technologies have been exploited to locate a set of relevant information, which can help the users carry out their tasks.

A number of Artificial Intelligence and Information Retrieval technologies have been applied to this domain. FOLIO [2] is an expert system to assist portfolio managers. It determined the client's investment goals and the portfolio that best meets them based on interviews with a number of clients and, on the basis of expert knowledge. Constantino and his colleagues [3] applied information extraction techniques to the analysis of financial news articles, in order to produce a set of relevant templates which represent the most important information in the article.

This task has many interesting features, including:

- The enormous amount of continually changing, and generally unorganized information available
- The variety of kinds of information that can and should be brought to bear on the task (market data, financial report, technical models, analysts' reports, breaking news, etc.)
- The many sources of uncertainty and dynamic change in the environment
- Information timeliness and criticality features that present the agents with hard and soft real-time deadlines for certain tasks
- Resource and cost constraints – not all data are available for free
- Relatively well-structured evaluation criteria and an experimentally verifiable testbed where decisions supported by the system can be evaluated using real world data and feedback

Given these observations, a multi-agent system approach is appropriate for portfolio management (or monitoring), because the multiple threads of control are a good match for the distributed and ever-changing nature of the underlying sources of information and news that affect higher-level decision-making processes. A multi-agent system, as described in [12], can more easily manage the detection and response to important time-critical information that could appear suddenly at any of a large number of different information sources. Last but not least, a multi-agent system provides a spontaneous mapping of multiple types of expertise to be brought to bear during any portfolio management decision-making process. A single-agent system could still take advantage of intelligent agents' properties, such as adaptiveness, proactiveness, and intelligence, but it would be vulnerable to a "single point of failure" and could not manage the large amount of information from various sources. On the contrary, a multi-agent system (MAS) has the following advantages over either a single agent system or centralized system:

- A MAS distributes computational resources and capabilities across a network of interconnected agents. Whereas a centralized system may be plagued by resource limitations, performance bottlenecks, or critical failures, an MAS is de-

centralized and thus does not suffer from the ”single point of failure” problem associated with centralized systems.

- A MAS allows for the interconnection and interoperation of multiple existing legacy systems. By building an agent wrapper around such systems, they can be inporporated into an agent society.
- A MAS models problems in terms of autonomously interacting component-agents, which is proving to be a more natural way of representing task allocation, team planning, user preferences, open environments, and so on.
- A MAS efficiently retrieves, filters, and globally coordinates information from sources that are spatially distributed.
- A MAS provides solutions in situations where expertise is spatially and temporally distributed.
- A MAS enhances overall system performance, specifically along the dimensions of computational efficiency, reliability, extensibility, robustness, maintainability, responsiveness, flexibility, and reuse.

2.1 Warren: A Multi-Agent System for Intelligent Portfolio Management

Taking those considerations described in the previous section into account, we proposed and implemented Warren, a multi-agent system for financial portfolio management [5], [13]. Briefly stated, the goal of this system is to provide an integrated financial picture on the companies of interest for managing an investment portfolio over time, using information from various sources available from the Internet.

A team of software agents in WARREN is derived from the set of reusable software component agents that comprise RETSINA agent framework. RESTINA is our domain-independent agent control, organization, coordination, and architectural scheme. This architecture coordinates four different types of agents: interface, task, middle, and information agents. An interface agent is in charge of interacting with users by receiving users' input and representing the results. A task agent helps users perform tasks by formulating problem-solving plans and carrying out these plans in collaboration with other agents. An information agent provides information from various sources. Middle agents help match agents that request services with agents that provide services. Warren consists of eight different agents that help the user manage their portfolio:

Warren Interface An interface agent for interacting with the user

ComptrollerAgent A task agent for managing the portfolio

RiskCriticAgent A task agent for analyzing the risk (β value) of the portfolio

MatchMaker A middle agent for maintaining an updated mapping between the agents in Warren and the services that they provide

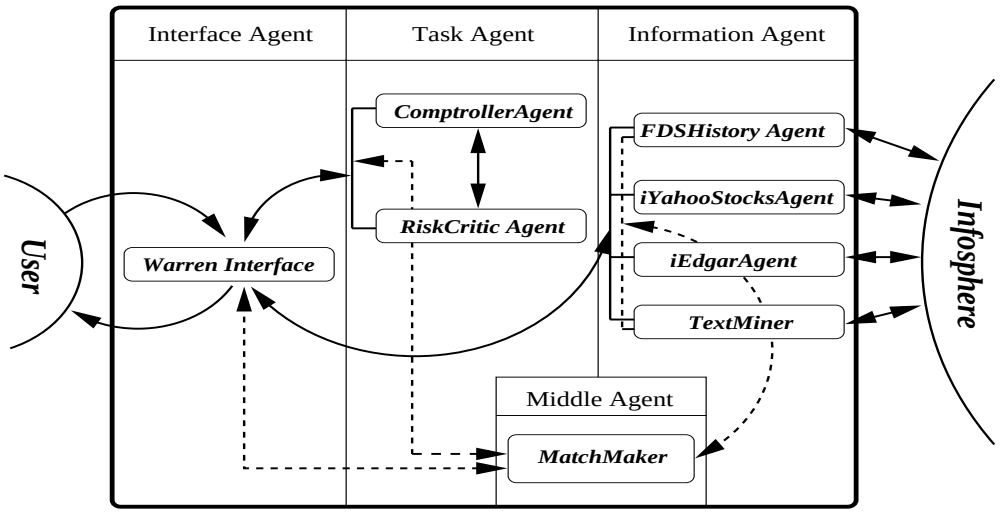

Figure 1: Warren is built on the top of our RETSINA framework. Two different types of line (solid and dashed) represent interactions among software agents. We distinguish interactions between the matchmaker and an agent and that between other agents because of the different semantics of these interactions.

FdsHistoryAgent An information agent for providing a historical view of financial data summary (FDS) on the company of interest

iYahooStocks An information agent for providing stock prices on the company of interest from Yahoo.com

iEdgar An information agent for providing financial data summaries from SEC's Edgar web site

TextMiner An information agent for providing financial news analysis

Figure 1 shows the architecture of Warren and interactions between agents. In this figure, the line from one agent to another represents an interaction between them. We distinguish interactions because they have a different semantics. In other words, an interaction (a dashed line) between the matchmaker and other agents is intended to locate a specific service provided by an agent, whereas another interaction (a solid line) between agents represent a request-response on a specific service. In the later section we will describe these interactions in detail.

3 TextMiner: An Agent for Text Analysis

TextMiner has been implemented to complement the quantitative financial information of Warren by providing an analysis of news articles. In particular, the task of TextMiner is to provide an accounting of the number of news articles on the company of interest, which reflect good or bad financial performance over a period. To accomplish this, the TextMiner agent performs the tasks of information retrieval on the company of interest,

selectively attending to news reports on the company by filtering out non-financial news, and classifying them into predefined categories by analyzing their contents.

Given a sequence of daily stock price movements in response to current "news" about the economy, world politics, industries, and companies, it might be possible to separate events that are directly associated with the price movement, from those that are not. Prior work in this field has been done to predict the future market trends, by analyzing the correlation between an event and the pattern of stock prices. Wutrich and his colleagues [15] focused on forecasting major market indexes, using a keyword based system. In [7], the naive Bayes classifier was used to link news stories to trends in intra-day trading for prediction. Note, however, that we are not trying to predict future financial performance of a company, but rather to provide a summary of trends about current financial performance on the company of interest.

We define a "financial" news article as one reporting facts directly related to a company's current financial status. For example, a news article reporting a company's earnings, activities on capital markets, revenues, and movement of stock price are considered "financial" news, whereas facts about corporate control (e.g., shareholder meetings, and personnel management), legal or regulatory issues (e.g., SEC filing) are filtered out as a non-financial news. Given our definition of "financial", a financial news article is classified into one of the following five classes, based on its content:

GOOD News articles which explicitly show evidence of the company's healthy financial status.
e.g.) ... Shares of *ABC* Company rose $1/2$ or 2 percent on the Nasdaq to $24-15/16. ...

GOOD, UNCERTAIN News articles which refer to predictions of future profitability, and forecasts.
e.g.) ... *ABC* Company predicts fourth-quarter earnings will be high. ...

NEUTRAL News articles which mention financial facts but do not provide good or bad aspects.
e.g.) ... *ABC* contributes $ 700 million in stock to its pension plan ...

BAD, UNCERTAIN News articles which refer to predictions of future losses, or no profitability.
e.g.) ... *ABC* (Nasdaq: ABC) warned on Tuesday that Fourth-quarter results could fall short of expectations. ...

BAD News articles which explicitly show evidence of the company's bad financial status.
e.g.) ... Shares of *ABC* (ABC: down $0.54 to $49.37) fell in early New York trading. ...

Two "uncertain" classes were added to deal with the "inter-indexer inconsistency" problem. This problem occurs when two different humans must make a decision on whether to classify a news article under the given classes, and they may disagree [1]. In other words, one may be allowed to decide the class of a news article, but there is

another classification reasonably possible. For example, the prediction of future earning by a (reliable or unreliable) news provider could be classified into either "good, uncertain" or "bad, uncertain."

It is important that intelligent software agents provide only relevant information, as one of the solutions to reduce information overload of the user. What the users of Warren are most probably interested in are the news reports about financial facts. However, the user may want to see the news articles which are not directly related with financial issues, but could nevertheless affect the company's financial outlook in the future, thus giving a general view of business activity by the company of interest. In short, it is important to provide a set of relevant information to the user while including information that is still valuable for a given task.

Taking these requirements into account, it is necessary for the TextMiner agent to segment a set of news articles on the company of interest into financial and non-financial categories after downloading them from various on-line news providers.

3.1 Information Retrieval Tasks of TextMiner

In this section, we will describe in detail the information retrieval tasks assigned to TextMiner, that is, we describe how it filters out non-financial news and classifies financial news into predefined classes.

The information retrieval tasks of TextMiner – which take place after it finishes downloading a set of news articles and before it presents results to users or other agents – proceed on the basis of the concepts from the information retrieval domain. In order to provide TextMiner with a set of learned classification rules, we downloaded 6,239 news articles and labeled them manually [2]. The collected news articles are first converted into machine-readable form, which is desirable for the given information retrieval tasks: filtering and classification, after removing textual noises such as stopwords, SGML-variant tags, and symbols. We adopt the conventional (real-valued) vector space model [11]. It is one of the most widely used models for text analysis because of its conceptual simplicity and the appeal of the underlying metaphor of using spatial proximity for semantic proximity. To be more specific, a news article is represented in a high-dimensional space, in which each dimension of the space corresponds to a term (word or phrase) in the document set. Next, a given document collection is represented by the term-by-document matrix $M = T \times N$, where there are T word (or phrase) features and N, $W = \{w_1, ..., w_t, ..., w_T\}$ and $D = \{\vec{d_1}, ..., \vec{d_i}, ..., \vec{d_N}\}, \vec{d_i} \in R^T$ respectively. The word feature set (W) is constructed by eliminating infrequent words and high frequency words. The elimination of words indicates that words are only considered as features, if they occur more than *frequent threshold* or at most less than *infrequent threshold*. Each term t_t has its weight w_t, which indicates how important it is for a given text learning task. A variant of TFIDF (Term Frequency $\times$ Inverse Document Frequency) [11] is used for calculating a weight. The idea of this weighting method is to ensure that the weight of a word is scaled from 0.0 to 1.0 while preserving the original idea of TFIDF, which gives a word higher weight if it is frequently appeared in a document and less frequently occurred across the document collection.

[2] The (financial) news article data set is available at http://www.cs.cmu.edu/∼softagents/textminer/data_set.html.

The weight of a word, w_t defined as:

$$w_t = \frac{(1 + \log(tf_{i,t})) \times \log \frac{N}{df_t}}{\sqrt{\sum_{s \neq t} (\log tf_{i,s} + 1)^2}} \quad (1)$$

where $tf_{i,t}$ is the number of times word t occurs in document d_i and df_t is the number of documents in the collection in which the word t occurs. The weight is then normalized by a document length. This model is often called the "bag-of-words model" because the factorial expression reflects conditional independence assumptions about word occurrences in d_i.

3.1.1 Segmentation of News Articles

The filtering process is carried out by comparing the similarity between one of classes, $C = \{financial, nonfinancial\}$, and a given news article $\vec{d_i}$ and then by assigning the news article to the closest class. A class model is a mean vector of the class which is generated by adding all the document vectors in the class, $\vec{c} = \frac{1}{|c|} \sum_{d \in c} \vec{d}$. For calculating the similarity (s), we measured the cosine angle between two vectors:

$$s(\vec{d_i}, \vec{c_j}) = \arg\max_{c_j \in C} \frac{\vec{d_i} \cdot \vec{c_j}}{||\vec{d_i}|| \cdot ||\vec{c_j}||}$$

In order to provide an overview of the whole range of news articles at a glance, TextMiner segments a set of non-financial news articles into more detailed categories. We construct non-financial categories manually, which are comprised of "product", "M&A", "strategy", and "miscell". It is possible to derive a set of manual segmentation rules because these categories are usually described in a limited and unambiguous vocabularies.

3.1.2 Classification of News Articles

We devised a new text classification method, called Domain Experts (DE), which classifies news articles into predefined classes in terms of the current financial status of the company of interest. The proposed method predicts the class of a financial news article through the voting process among experts, which are frequently co-located phrases. A co-located phrase is a sequence of nearby but not necessarily consecutive words. Thus, a set of frequently co-located phrases in a class is available for discriminating the class of financial news articles because it often appears in the class. For example, *Shares* and *rose* can be selected from a sentence in a news article such as "Shares of Company ABC rose 1/2 or 2 percent on the Nasdaq to $24-15/16...," as a frequently co-locating phrase for a "good" class. It is often desirable to consider such contextual information (i.e. word-collocation) rather than frequency statistics with respect to the characteristics of English text, because "word-collocation" has characteristics of a syntatic and semantic unit, whose exact and unambiguous meaning of connotation cannot be derived directly from the meaning or connotation of its components [9]. Not all co-located phrases are selected as a feature, due to the existence of the most informative phrases. The most

informative co-located phrases are those that would reduce classification error and variance over the distribution of examples. In order to select these features, we computed the information gain for each of the frequently co-located phrases in the training data and removed from the feature space those phrases whose value was less than a predefined threshold. Given these features, our method was trained to adjust the weight of each of the experts before they were deployed in Warren.

In addition to this classification task, it is important for an agent to learn the content-shift by itself because the vocabularies of text domain is slightly changed from time to time and the intervention of humans for labeling text data is quite expensive. The devised method for providing TextMiner with self-learning capability shares a property of the uncertainty-sampling [8], in that it predicts the label of an unlabeled data on the basis of the learner's confidence, which is obtained through the training phase. The instances (i.e. news articles) that are labeled with the class label as least uncertain. Unlike uncertainty-sampling, our method relies only on the vote by each of member of domain experts group, which has knowledge induced from the labeled training data. We, however, could not rely on its knowledge completely, due to the existence of noise in the training data. To do this, λ is introduced for regulating the degree of reliance on learner's experience. Empirically, the proposed sampling method shows the best performance at 70% confidence.

The class uncertainty of an unlabeled news article is determined by the value of vote entropy. Vote entropy is the entropy of the class label distribution resulting from having each (experts) group member, which are appeared in a news article, deterministically "vote" for its winning class [4]. Let $V(j)$ be the number of domain experts which are extracted from news article d_i and are involved in 'voting' for the class j.

$$VE(d_i) = -\sum_{j}^{|C|} \frac{V(j)}{|K|} \log \frac{V(j)}{|K|}$$

where $|K|$ is the total number of domain experts which took part in voting of ith news article, d_i which is ith news article from the unlabeled data set.

While the vote entropy is 0 if a number of domain experts participating in the vote belong to the same class, the vote entropy is 1 when the vote committee consists of an equal number of each class. We found empirically that the vote entropy for a class assigned correctly was less than 0.25, whereas the average entropy for incorrectly classified data was greater than 0.7. In other words, the class of an unlabeled news article is attached by the class label voted by majority if the vote entropy was less than 0.25. Otherwise, the class of an unlabeled news article is not determined.

3.1.3 Evaluation of Financial News Classification

In this section, we describe the experimental results of the proposed classification method, as compared with existing methods. Experiments were performed using the text data which we had made by ourselves. The data set amounts to 6,239 news articles: 1,239 labeled manually and 5,000 unlabeled. These news articles were gathered from various electronic news providers: CNN Financial Network, Forbes, Reuters/Reuters

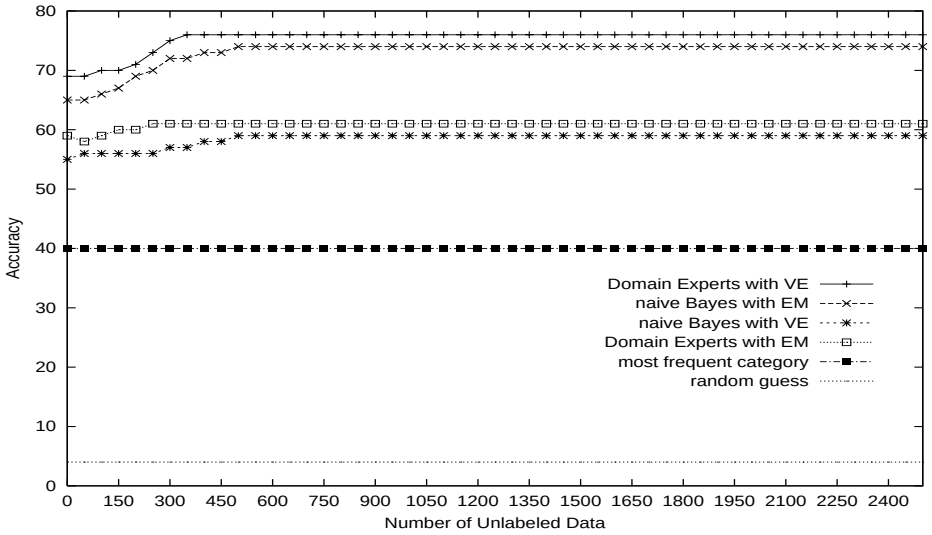

Figure 2: A result of sampling experiment was represented after training each of the methods with 1,239 labeled data. “Most frequent class” and “random guess” are base lines of performance. In the case of the “most frequent class” method, it always labeled the class of an news article “neutral” when asked to predict the class of an unseen news article.

Securities, NewsFactors, Motley Fool, CNet, ZDNet, Morningstar.com, Associate Press (AP), AP Financial, and Business Wire.

Experiments aimed to verify the proposed methods in terms of two performance criteria: how well it makes use of unlabeled data for improving classification accuracy and how accurately it classifies the latest news articles into predefined classes.

Firstly, we evaluated whether the proposed sampling method would improve classification performance better than those trained by existing methods, such as the combination of naive Bayes classification and Expectation-Maximization (EM). Figure 2 shows results of verifying the accuracy performance of each sampling method with a different number of labeled data. In this experiment, all labeled data were used for training. A total of 50 iterations were carried out for each method. At each iteration, 50 unlabeled news articles were given to each of the methods and were used for improving its performance. After the training phase, each method was tested by classification accuracy defined in terms of the proportion of the number of news articles classified correctly to the number of total news articles that were used. From this observation, we assumed that approximately 1,700 news articles (1,239 labeled and 450 unlabeled news articles) would allow us to make a classifier with 75% accuracy, because of the fact that most of news companies that we used for the delivery of financial news have a restricted vocabulary set.

The second experiment was performed to show the accuracy of classification of the latest financial news articles. The latest data is made up of the news articles that are gathered from the same news sources as the labeled data set and that report the latest financial news at the experimental time. For this experiment, we downloaded 1,200 news articles from the same online news providers. This data set was made up

classes	$+$	$+/?$	$+/-$	$-/?$	$-$	total
# of articles	85	1	243	0	220	549
DE	.76	1	.8	–	.78	.79
naive	.61	0	.68	–	.62	.65

Table 1: The result from the experiment on the latest news was shown. DE and naive represent "Domain Experts" and "naive Bayes" text classification methods, respectively. Each column at the third and fourth row represents the accuracy of each category in terms of the proportion of the number of news articles correctly classified to the total number of news articles for the category. These values are derived after a human finished manually labeling all of these news articles.

of 20 downloading trials where each trial was designed to collect 60 news article on a company. We could get 549 financial news articles out of the latest 1,168 news articles – 32 downloaded news articles out of 1,200 are too short to use for training data. As a result, the proposed method has 79% averaged accuracy, which means 433 out of 549 total financial news articles were classified correctly. Table 1 shows the accuracy of tested methods per each class.

The proposed algorithm which observed the co-located phrase of a certain class from news contents and predicted the label with Weighted-Majority voting outperformed the naive Bayes classifier by approximately 14%. In order to acquire improved accuracy and self-learning, we proposed a sampling technique which can determine the class of an unlabeled news article, given its entropy value. With the proposed sampling method of self-confident sampling, a 16% accuracy is improved by using 9% unlabeled data (450/5000). The successful results from the sampling test and online test supports the hypothesis that proposed algorithms effectively help TextMiner carry out its information retrieval task, even though the promising results have been derived partly from the task characteristics whose decision boundaries are relatively objective. With these experimental results, the TextMiner is deployed in Warren.

4 Putting It All Together

In this section, we describe how Warren manages a portfolio intelligently by coordinating a team of software agents.

Since the matchmaker, as a middle agent, is responsible for maintaining an updated mapping between the agents in Warren and the services that they provide, it initializes the virtual work-space for agent-naming and resources for Warren. Next, other agents in Warren, – ComptrollerAgent, RiskCriticAgent, MatchMaker, FdsHistoryAgent, iYahooStocks, iEdgar, and TextMiner – are invoked to register their services' advertisements with the Matchmaker.

If the initial coordination among the agents is successful, the individual user will see the Warren interface agent that describes the current status of his portfolio (Figure 3). The Warren interface agent displays a comprehensive summary of the user's portfolio and also allows the user to buy and sell stocks.

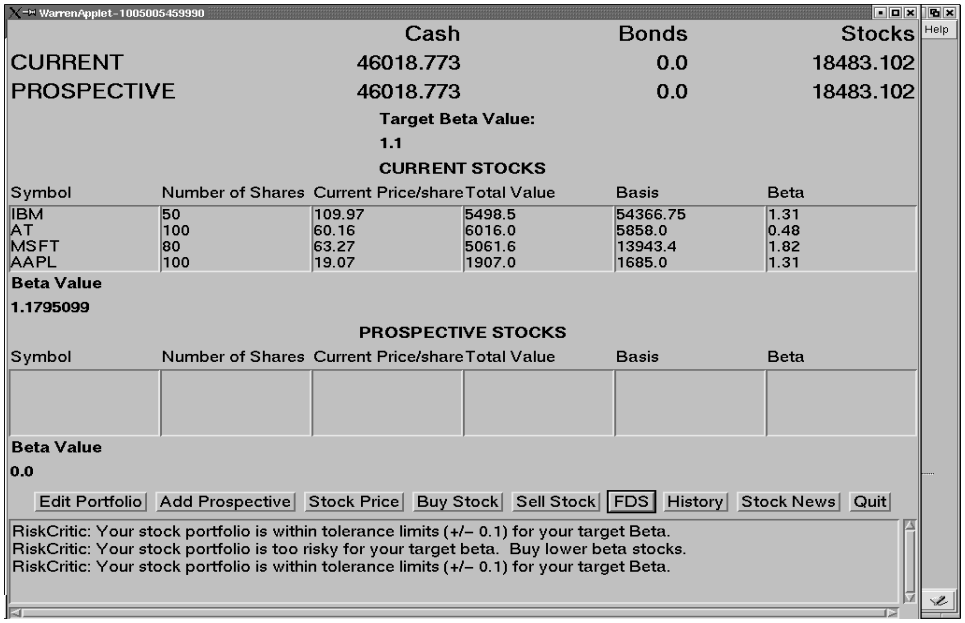

Figure 3: The user interface of Warren is shown. The Warren interface agent is in charge of presenting a comprehensive summary of the user's portfolio. In this figure, the Warren interface agent provides current valance, his or her holdings of each of the four companies, and risk value

Upon requests from the user or other agents, – in this case the user want to see financial information about IBM – the Warren interface agent delegates task components to one or more agents by sending a query to the matchmaker for an agent, which is able to provide an appropriate service. As the Warren interface agent graphically and textually interacts with the user, other agents coordinate tasks, acquire information, and send results, recommendations, and analysis to the user via the interface agent. Figure 4 shows interactions among the agents via a control panel. In order to accomplish a given task, two agents interact with each other after acquiring information about other agents from the matchmaker.

Information agents monitor and start to collect stock and other financial sources from the web in real time after getting requests from the matchmaker; the TextMiner collects and classifies news articles on the company of interest; iYahoo gathers stock prices and other stock related information; iEdgar harvests Financial Data Summaries (FDS) from SEC 10-k filings, from the EDGAR web site; FdsHistoryAgent gathers data from multiple years of financial data summaries and presents a historical view of the data. Figure 5 and 6 show financial data summaries and financial news analysis that present multiple perspectives on IBM, respectively. Data culled from the infosphere and stored locally by information agents are sent to one or more task agents upon request. Next, following a process of data analysis and integration at the task agent level, information is ultimately displayed to the user via the interface agent.

According to the user's activities, the RiskCriticAgent evaluates portfolios for financial risk using a risk measure referred to as "β". The ComptrollerAgent is in charge of maintaining records of a user's portfolio, and buying and selling the user's stocks.

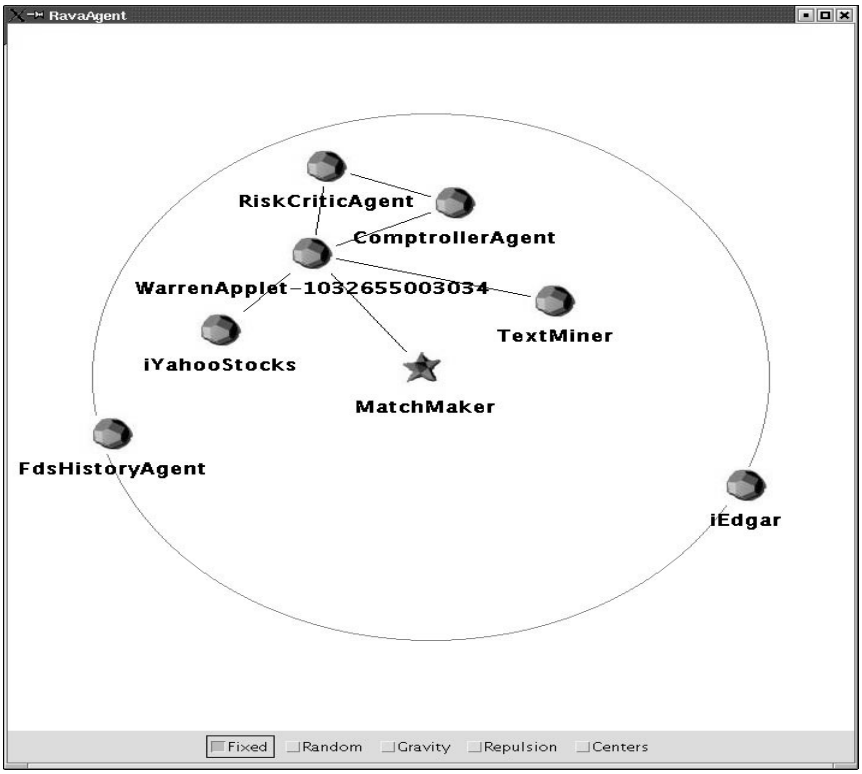

Figure 4: A control panel in Warren shows interactions between TextMiner and other agents. Through this panel, the user can monitor the stream of information between agents. Two agents collaborate with each other after acquiring information about other agents from the matchmaker. Again, the line from an agent to another represents an interaction between them. The circle around agents represents a virtual agent workspace.

5 Conclusion and Future Work

In this paper, we presented Warren, a multi-agent system for intelligent portfolio management, which is motivated by the great benefits of working in teams within the domain of Distributed Artificial Intelligence (DAI), and TextMiner, which takes advantage of information retrieval techniques to complement quantitative financial information. The goal of portfolio management in Warren is to provide an integrated financial picture for managing an investment portfolio over time, using the information from various sources available over the Internet.

With comprehensive quantitative information gathered by efficient coordinations between agents, and quantitative information supplemented by financial news analysis, we showed a successful application of a multi-agent system for portfolio management.

In future work, we will employ information extraction techniques for the summary of financial news articles, in order to help an investor's understanding of market situations. We will also devise a component for decision support that employs the statistical techniques, such as regression and correlation analysis, between the risks and returns

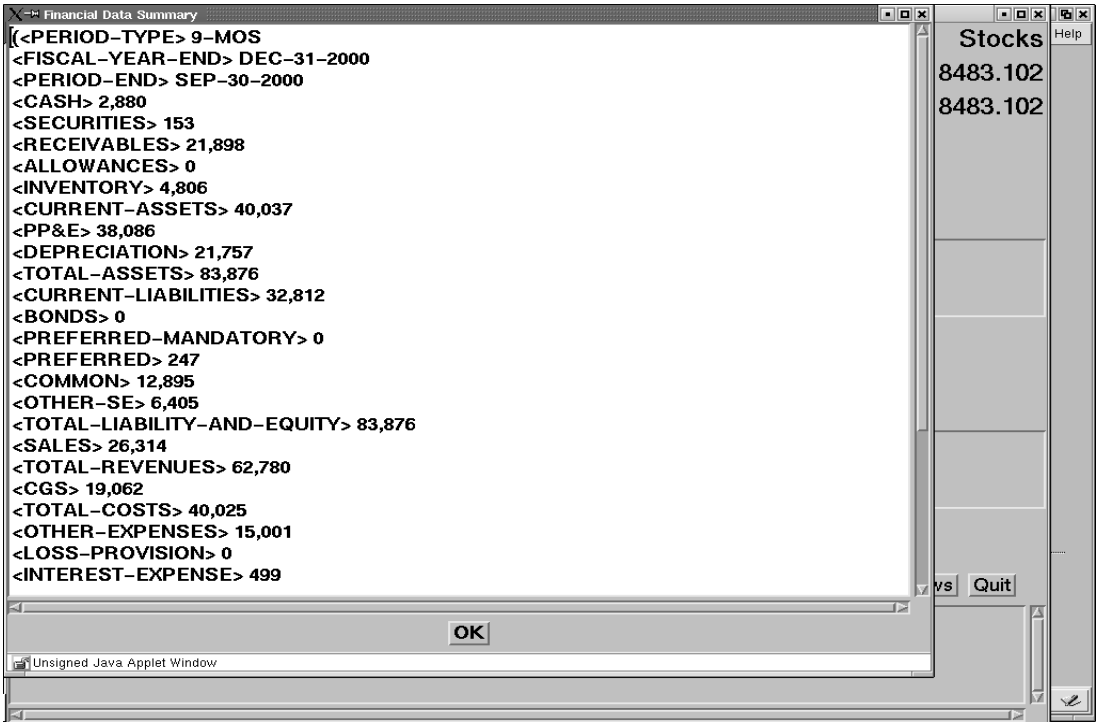

Figure 5: The financial data summary on IBM is shown. The Financial Data Summary is a section of the 10-k report that the US Securities and Exchange Commission (SEC) requires that all corporations file.

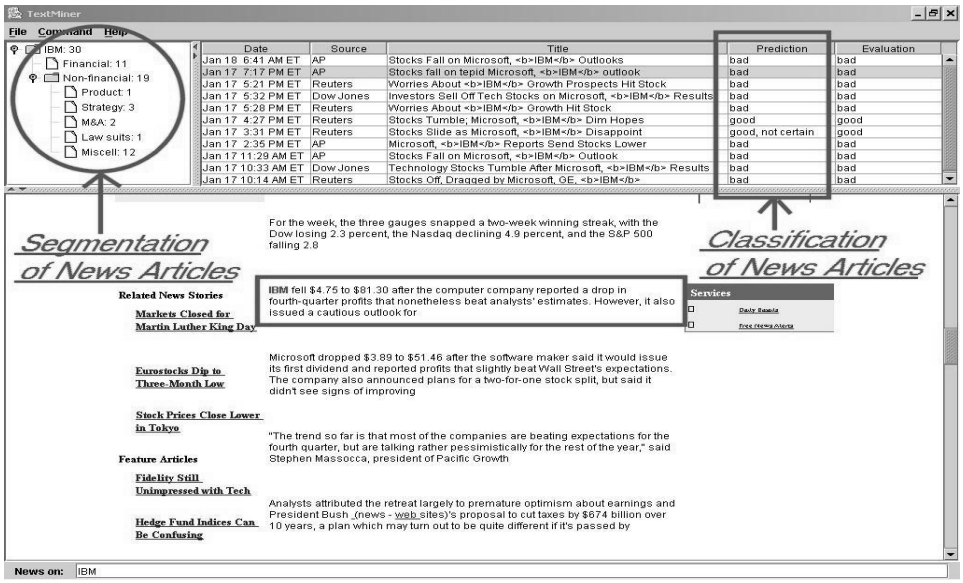

Figure 6: A set of news articles (30 is the default value) on IBM is shown on the user interface of TextMiner agent. The top left shows that there are 11 financial news and 19 non-financial news on the company. At the top right is a summary table that presents collected news articles in real time and results of classification. The content of a news article clicked is displayed in the bottom section. The "Prediction" and "Evaluation" columns represent the result of classification by TextMiner and by Human respectively.

from individual investments by fusing information from various sources over Internet.

6 Acknowledgements

This research has been sponsored in part by DARPA Grant F-30602-98-2-0138 and the Office of Naval Research Grant N-00014-96-16-1-1222.

References

[1] C. Celverdon. Optimizing convenient online access at biliographic database. In *Information Services and User*, pages 37–47, 1984.

[2] P. Cohen and M. Lieberman. A report on folio: An expert assistant for portfolio managers. *Investment Management: Decision Support and Expert Systems*, pages 135–139, 1990.

[3] M. Constantino, R. Morgan, R. Collingham, and Garingliano. Natural language processing and information extraction: Qualitative analysis of financial news articles. In *Proceedings of the Conference on Computational Intelligence for Financial Engineering*, 1997.

[4] I. Dagan and P. Engelson. Committee-based sampling for training probabilistic classifiers. In *Proceedings of International Conference on Machine Learning*, 1995.

[5] K. Decker, K. Sycara, A. Pannu, and M. Williamson. Designing behaviors for information agents. In *Proceedings of International Conference on Autonomous Agents*, pages 404–413, 1997.

[6] D. Hayes and W. Bauman. *Investments Analysis and Management.* MacMillan Pub., 1976.

[7] D. Lawrie, P. Oglivie, D. Jensen, V. Lavrenko, M. Schmill, and J. Allan. Language models for financial news recommendation. In *Proceedings of the International Conference on Information and Knowledge Management*, pages 389–396, 2000.

[8] D. Lewis and W. Gale. Training text classifiers by uncertainty sampling. In *Proceedings of International ACM Conference on Research and Development in Information Retrieval*, pages 3–12, 1994.

[9] C. Manning and H. Schutte. *Foundations of Statistical Natural Language Processing*. MIT Press, 1999.

[10] H. Markowitz. *Portfolio Selection: Efficient Diversification of Investments.* B.Blackwell, 1991.

[11] G. Salton. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*. Addison-Wesley, 1989.

[12] K. Sycara. Multiagent systems. *AI Magazine*, 10(2):79–93, 1998.

[13] K. Sycara, K. Decker, and D. Zeng. Intelligent agents in portfolio management. In *Agent Technology: Foundations, Applications, and Markets*, pages 267–283. Springer, 1998.

[14] R. Trippi and E. e. Turban. *Investment Management: Decision Support and Expert Systems*. Van Nostrand Reinhold, 1990.

[15] B. Wuthrich, V. Cho, and J. Zhang. Text processing for classification. *Journal of Computational Intelligence in Finance*, 7(2):6–22, 1999.