ADMIRAL: A Data Mining Based Financial Trading System

Gil Rachlin[1] ,Mark Last[1] , Dima Alberg[1] and Abraham Kandel[2]

Abstract— **This paper presents a novel framework for predicting stock trends and making financial trading decisions based on a combination of Data and Text Mining techniques. The prediction models of the proposed system are based on the textual content of time-stamped web documents in addition to traditional numerical time series data, which is also available from the Web. The financial trading system based on the model predictions (ADMIRAL) is using three different trading strategies. In this paper, the ADMIRAL system is simulated and evaluated on real-world series of news stories and stocks data using the C4.5 Decision Tree Induction Algorithm. The main performance measures are the predictive accuracy of the induced models and, more importantly, the profitability of each trading strategy using these predictions.**

I. INTRODUCTION

The Efficient Market Hypothesis (EMH), as stated by Fama ([7], [10], [11]), assumes that 'Stock prices fully reflect all their relevant information at any given point in time'. As the basis for growth and development of a modern economy this means that no information or analysis can be expected to outperform the market and that stock prices follow 'Random Walks' ([9]), where a change in stock price over time is purely random and statistically independent of the stock price in the past. However, to this day no one can explain the anomalies in the market, which can be utilized to assure some short term predictive power ([6], [9], [12]).

In making their own forecasts most financial specialists try to exploit the time gap of the market's adjustment to new information. They reduce their risk by combining both technical (base future price predictions on past prices) and fundamental (base predictions on real economy factors, such as inflation, trading volume, organizational changes in the company etc.) analysis strategies, which are mentioned by Gidófalvi ([4]) and fully explained by [9]. In order to obtain the data required by both strategies, one can refer to various publicly available resources like the stock market itself, the companies, news articles, etc.

A rather new source for information in the late 20th and the 21st centuries is, of course, the Internet. In order to exploit this relatively new media as an additional tool supporting the forecasting task, we need to combine techniques from both time series data mining and web content mining.

The conventional approach to modeling stock market returns is to model the univariate time-series with autoregressive (AR) and moving average (MA) models. Recently, Engle [13] and Bollereslev [1,2] provided a new very powerful tool for the modeling of financial data in general and stock market returns in particular. The new process suggested by Engle and Bollereslev [14] is different from earlier conventional time series models [7] in that, instead of making the assumption that the variances are constant they allow the conditional variances to change over time as functions of past errors. These models are deterministic in the sense that they attempt to use mathematical equations to describe the process that generates the time-series. A disadvantage of these models lays in the assumption that trader or financial analyst needs to determine the appropriate number of lags and sometimes the successful analysis is based on the experience of analyzing the enormous variety of time series econometrical models. The advantage of these models lays in their ultimate interpretability.

Most studies ([8], [13], [23]) agree that the process of Knowledge Discovery in Databases (KDD), involves iterating over four general steps each using independent tools: 1) data cleaning and preprocessing (create a common data representation from different sources and different data types ranging from relational, transactional and spatial databases to large repositories of unstructured data such as the World Wide Web), 2) discover relationships in the data using data and text mining algorithms, 3) post processing of discovered patterns, 4) Use the model to perform actions in the real world.

When adding the aspect of time to the Data Mining process, it is understood ([8]) that database records are time stamped and meaningful only as part of a time segment or time series. In [8], Last et al. use a signal processing technique to pre-process the raw time series data.

Most of the studies done in order to combine inference from time-stamped news stories and time series stock data are different in their concepts and methods. Each study uses a different time series, text classifier, features, target attributes, time window length, weighting method etc. However, they do go through the following common stages:

– Define stock trends from the raw Time Series stock data using similar methods to those used by [6].

– Define a Window of Influence, which is a time frame taken before and after the publication time, *t*, of a web

Manuscript received November 25, 2006. This work was partially supported by the National Institute for Systems Test and Productivity at University of South Florida under the USA Space and Naval Warfare Systems Command Grant No. N00039-01-1-2248.

[1] - Ben-Gurion University of the Negev, Department of Information Systems Engineering, Beer-Sheva 84105, Israel, {gilrach, mlast, alberg} @bgu.ac.il

[2]- Dept. of Computer Science and Engineering, University of South-Florida, Tampa, Florida 33620, USA, kandel@csee.usf.edu

article ([Window Start Point], [Window End Point]): In [2], Gidófalvi defined it as [-20, 20] minutes from the publication time *t*. Lavrenko et al. ([3]) define the Window as 5 hours before t ([-300, 0] minutes) and [15] use a one-hour Window before t ([-60, 0] minutes).

– Align the time stamped news articles to stock trends according to the Window of Influence and score them: in [2] the news articles are scored relative to the change in stock price and the change in the price of the index it belongs to (Δstock-price, Δindex-price) and labels them in reference to a threshold value. Both [19] and [16] compare a predefined list of key words, which were given by experts, to the occurrence of words in the text.

– According to the common KDD practice, induce a model, which learns how to classify an article with a predefined trend and use the prediction model to detect future trend occurrences: [4] uses the Rainbow Naïve Bayesian text classifier package and [5] also uses the Bayes theorem to find trend relevance probability, but they compare a new arriving document about a certain stock against a representative 'language model' of five possible trends. Both [19] and [20] follow a rule base approach to create probabilistic classification data-log rules, which are applied to one or more backward time periods (one hour to one day).

– Evaluate the model prediction ability: [4] showed low predictive power explained by the existence of duplicate stories in the dataset. After running a 40-day simulation on the real market, [5] showed better results than random actions. Predictions made by [19] outperformed conventional time series analysis, two different neural nets and random guessing. [20] tried to predict the indices of five global markets and the importance of their results is in showing that the best accuracy (sometimes over 60%) was achieved in the US market indices.

In this paper, we present a new system (Admiral) for detecting stock trends based on the combination of Data Mining and Web Content Mining techniques. Admiral - is a new Financial Trading System which: 1) creates a "melting pot" of numeric and textual data before running an induction algorithm, 2) extracts automatically key words and phrases instead of using a prior expert list of phrases, 3) eliminates the need for word independence assumption by using Decision Trees rather than Naïve Bayes, 4) extends the Window of Influence of news articles in the prediction task from minutes to days.

The rest of this paper is organized as follows: In section II we describe the stages needed for ADMIRAL to be operationally useful. Section III describes the evaluation of the system in a simulated environment. Section IV presents the evaluation results and finally, Section V provides the conclusions.

II. ADMIRAL – Performing the Prediction Steps

The ADMIRAL system is designed as a full cycle prediction system for stock trends according to past numeric values of the stocks as well as their related textual web articles. ADMIRAL goes through six steps, as shown in Fig. 1, which are:

– Step 1: Data Collection from the Web.

– Step 2: Feature Extraction.

– Step 3: Textual Weighting.

– Step 4: Combined Data-Set Construction.

– Step 5: Classification Model (Decision Tree) Induction.

– Step 6: Market Action Recommendation.

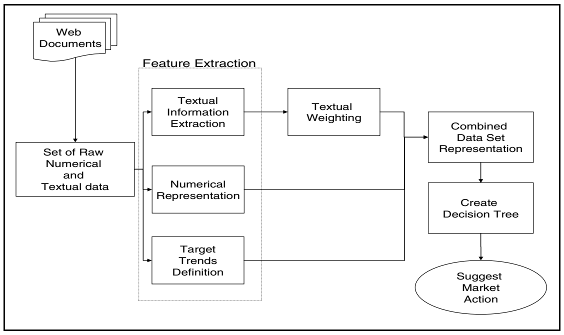

Fig. 1. Prediction Scheme with Mixed Numerical and Textual Data

Most steps are supported by various tools to perform the task at hand. The preferred configuration of tools for ADMIRAL's task can either be set manually or automatically. The automatic setting aims at finding the best configuration, which will eventually bring more accurate predictions of the stock market. In this paper, the system recommendations are compared against actual market results in order to evaluate the predictive power of several system configurations. Following is a brief description of each step in the system.

A. *Step 1: Data Collection*

In this research, our system needs to collect data from financial web sites, which are considered to have accurate real-time data (numerical and textual). Good choices of financial sites, which are also being used by financial professionals and can be used in our system are: http://www.forbes.com and http://today.reuters.com. In order to extract the relevant data from each one of the selected sites we created a configuration XML file, which includes its structural characteristics. Thus our system becomes generic and whenever we are interested to collect data from a new financial site the only addition will be the definition of an appropriate configuration XML file.

The frequency of data collection for the textual training data is once a day after the end of trade. For each textual article we keep its publication time stamp, its header and content. Numerical data is collected three times a day: after the trade opens, in the middle of the day and after the trade is closed.

B. Step 2: Feature Extraction

The feature extraction is done for both textual and numerical data. Feature Extraction for Textual Data includes two activities: first is the automatic extraction of key words and key phrases from a predefined Window of Influence. Second is the creation of a word vocabulary, which includes the most influential words in all our past Web articles. Feature Extraction for the Numerical data also includes two activities: first is the calculation of additional commonly used financial values. Second is the long term stock price trend discovery.

The Automatic Key Word Extraction is done by the Extractor software package ([24]). The Extractor is a text summarization engine, which uses a patented genetic extraction algorithm, GenEx. GenEx analyzes the recurrence of words and phrases, their proximity to each other, and the uniqueness of the words in a particular document. It removes all stop words from the document, applies a stemming procedure and selects a limited number of the most influential words in the document. In ADMIRAL, we construct each document, *d*, to be analyzed by the Extractor, as a set of the Web articles related to a specific stock within a backward Window of Influence.

The Term Dictionary has a predefined number of words, which we define as N. It is rebuilt each time a new training set is evaluated for creating a new classification model. Each word in the word dictionary receives a score *S*, which determines its degree of membership inside the dictionary and only the *K* highest ranking words will eventually be used from the final dictionary ($K \leq N$). As seen in Equation 1, the score, *S*, is calculated based on several parameters.

$$S = \frac{1}{2} * \frac{TF}{N} + \frac{1}{2}(\frac{P}{L} * \frac{B}{L}) \qquad (1)$$

Where:

L: the time frame, in days, for the word dictionary.

B: the time window between the first and last occurrence of a word.

P: the number of days to the last occurrence of a word.

TF: the number of occurrences of a word during *L* (known as Term Frequency).

N: the number of words in the dictionary.

The Target Trends Definition is obtained by following the method introduced by Last et al. in [8]. We are interested in the value (stock rate) for each stock at each point of measurement, t. Every such point is part of a trend of values (mostly increasing, mostly decreasing, mostly remain the same) which has a starting point and an end point (the length of the trend is determined in day units), a slope degree and a fluctuation of the values, which constitute the trend.

We gave an equal importance to the Term Frequency vs. the multiplication of the last occurrence of a word and the block of days where it occurred. As an example, let us define the final number of words to be entered into the word dictionary as 20,000 (N = 20,000) and the total number of days for creating the word dictionary as 30 (L = 30). Let us assume that the phrase "High Volume" appeared 8 times in all the previously collected documents (TF = 8). The last occurrence of the phrase was 3 days ago (P = 3). Its first occurrence was 10 days ago (B = 7). Thus after substituting the values into Equation 1, the Grade of Membership S, for the key phrase "High Volume" is set to 0.0118.

C. Step 3: Term Weighting

In order to later use the extracted textual features within each Window of Influence, [t-i, t] we need to assign a normalized value between zero and one to each key phrase. The normalization is done by dividing the weights, which were either provided by the Extractor or calculated using TF or Boolean methods ([16, 19]), by the overall grade of membership in the word dictionary, which was also calculated in Step 2 above.

D. Step 4: Data Set Construction

After having prepared both numerical and textual data and assigned a trend to each one of the stock's prices, we need to combine the data, which we want to include in our prediction task.

Our goal is to predict the forthcoming trend, which will last more than a predefined time length. Hence our target attribute is the trend, which can take the following five values: Up, Slight-Up, Expected, Slight-Down and Down. We assume that the target trend is influenced by the two latest periods of textual data prior to its occurrence: [t-2i, t-i] and [t-i, t] and the extracted numerical trend information. We create a training dataset by concatenating all that extracted data. The final dataset structure is shown in Table 1.

E. Step 5: Decision Tree Induction

We use a Decision Tree Induction algorithm, which doesn't assume attribute independence, The algorithm is C4.5 developed by Quinlan in [22]. This algorithm will yield a set of trend and length predicting rules on which we can rely in order to perform our next step of recommendation. In order to show the effect of the combination between Numerical and Textual data, we have evaluated the algorithm on each separate type of data.

An example of an induced decision tree is shown in Fig .2. Each path, which goes from the root node through the different layers of nodes to a target node, can be viewed as a decision rule.

TABLE 1. DATA SET STRUCTURE

Candidate Articles Windows of Influence (24 hours). number of key words/Phrases predefined (30 * 2). Data Set is for two consecutive backward windows.													TARGET
Date/ Time Stamp (t)	Stock	General Numerical Data			Time window (t-2i)				Time window (t-i)				
		ß	…	Profit Ratio	Term1	…	Term 29	Term 30	Term1	.	.Term 29	Term 30	
01/01/2006 09:45	MSFT	1.7		19	0.02		0.45	0.3	0.02		0.45	0.3	Slight Up
: .													
01/10/2006 17:45	GOOG	1.9		14	0.03	…	0.27	0.35	0.03	. .	0.27	0.35	Down

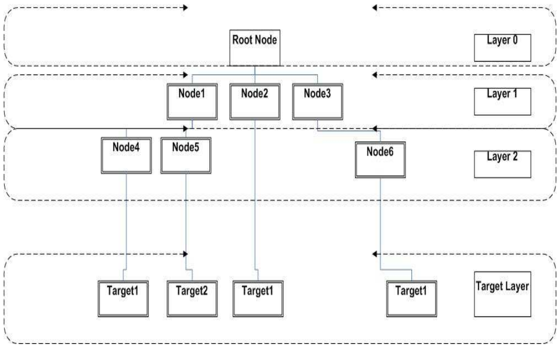

Fig. 2. Example of a Decision Tree with two Target Trends

F. Step 6: Trading Recommendation

After running the induction algorithms on the training set, the system can run on real-time data in order to recommend trading actions. New data collected from the web sites are put through the above mentioned steps 1-4. After predicting a trend and its length from the induced model, a good and objectively measurable trading strategy should be followed. We currently formulated three different, non random, trading strategies to be compared in this research:

- Buy and Hold –at the beginning of a predefined time period, we buy the recommended stocks and hold them until the end of the predefined time period.
- Automatic – We follow the exact recommendations of the system's rules defined in Table 2 below.
- Semi Automatic – We combine actions based on the recommendations of the system with our own assessment, which is based on our prior knowledge and guesses.

As shown in Table 2, when an automatic strategy is used the recommended action is set according to the trend. If the trend is Up we will buy the stock expecting to earn from the increase. If the trend is Down we sell short, hoping to make money on the ability it gives us to sell at a higher price than the expected future price. If the trend is Expected (stable prices) we do nothing.

TABLE 2. REAL TIME MARKET ACTIONS BASED ON THE PREDICTED TRENDS

Trend in the Model	Action
Up	Buy
Down	Sell Short
Expected	No Action

III. SYSTEM EVALUATION

We decided to evaluate ADMIRAL on stocks from the US NASDAQ index on a three months period time frame of financial stock data from February 7th through May 7th 2006. We collected data on 40 stocks and later chose the five stocks which had the largest amount of associated textual data and which are listed in Table 3.

TABLE 3: THE 5 STOCKS, WHICH WERE CHOSEN FOR OUR EXPERIMENTS

Stock Number	Stock Symbol	Stock Company	Stock Sector
1	CSCO	CISCO SYSTEMS INC.	Technology
2	EBAY	EBAY INC.	Services
3	MSFT	MICROSOFT CORP.	Technology
4	TEVA	TEVA PHARMACEUTICAL	Healthcare
5	YHOO	YAHOO! INC.	Technology

The data was collected from the two financial web sites mentioned in Sub-section II.A above: http://www.forbes.com and http://today.reuters.com. In addition to their high reputation in the financial world, two more reasons made us to choose them: first is that, at the time of our experiments, the articles on each web site were mutually exclusive, which means that a comparison between the information gathered from them would not overlap. Second, they allow data filtering according to requested stocks, which reduced our information extraction efforts.

IV. RESULTS AND COMPARISONS

Table 4 lists the prediction accuracy results based on textual and numeric data collected from two financial internet sites: Forbes and Reuters. From this table, we note that both sources gave us the same highest accuracy result (83.3%) for numeric data. In our opinion, this result is quite expected since both web sites are supposed to provide the same numeric information on stock prices.

On the whole, the most efficient source for textual and numeric analysis is Reuter's site (reported accuracies are: 82.4% for join textual and numeric analysis and 80.6% for textual analysis) and on other hand we also note that the best accuracy result for pure textual analysis (80.6%) is not significantly lower than the best numeric and joint textual numeric accuracy results. However, contrary to our initial expectations, these results do not indicate that the textual information can improve the predictive accuracy of the numeric analysis.

TABLE 4: PREDICTION RESULTS ACCURACY (C4.5, EXTRACTOR WEIGHTS)

Forbes	Reuters	Textual	Numeric	Accuracy
Yes	No	Yes	Yes	82.4%
Yes	**No**	**No**	**Yes**	**83.3%**
Yes	No	Yes	No	77.5%
Yes	Yes	Yes	Yes	81.5%
Yes	Yes	No	Yes	83.3%
Yes	**Yes**	**Yes**	**No**	**77.0%**
No	Yes	Yes	Yes	82.4%
No	Yes	No	Yes	83.3%
No	Yes	Yes	No	80.6%

Table 5 shows the configurations, which had the highest Return on Investment. It should be noted that in order to compute the expected profit we needed to avoid overnight risk during hours when the market was closed. Thus we employed a Single Day Trading Strategy (SDTS), which was also used in other studies and was mentioned in the literature ([15, 24]). This means that at the end of each day all our short term stocks holdings were sold. We need to compare our results against a random activity scheme, where an arbitrary daily action is taken on each stock. The overall ROI which was obtained on our data, was $2,091 in Random Strategy, $2,000 in System Recommendation Strategy and $23,341 in Buy and Hold Day Trading Strategy (SDTS). The main drawback of random strategy stems from its random nature, since a stockholder cannot change his stockholding position, for instance he is unable to turn his sell position into a short sell position (TEVA -760.15), and vise versa.

At the same time the other strategies allow the stockholder to execute switching in his stockholding position (sign of asterisks *- means that system has performed switching in stockholding position recently). From this aspect, the System Recommendation Strategy looks more preferable than Buy and Hold Day Trading Strategy (SDTS) because it required less switches of stockholding position for the experimental period.

TABLE 5: ROI RESULTS OF RANDOM, SYSTEM, AND BUY AND HOLD [SDTS] STRATEGIES

Stock	Random Strategy	System Recommendation Strategy	Buy and Hold Strategy [SDTS]
MSFT	+911.8$	+1280.5$	+1969$
YHOO	+2091$	+2000.4$	+20729.2$*
EBSY	+763.8$	+11264.2$*	+17626.8$*
CSCO	+160.9$	+16845.7$*	+16845.7$*
TEVA	-760.1$	+6850$*	+23341.4$*

V. CONCLUSION

In this research a new model for discovering future stock trends by using data mining and web content mining techniques was developed. The research study aimed at showing an improvement in the stock trend profitability by finding the best configuration of prediction and trading strategies. The methods to improve the profitability include:

1. Combination of both numeric and textual data.
2. The use of an automatic text extraction mechanism instead of a predefined expert list.
3. The use of decision tree prediction model instead of Naïve Bayes classification.
4. The use of smart trading strategies.
5. The implementation of a full cycle prediction system (ADMIRAL).

The components of the method were also evaluated and compared against existing techniques on data from two mutually exclusive financial web sites. The proposed ingredients of the method showed improved prediction ability as well as improved profitability with a relatively low number of attributes necessary to achieve them.

Future research may enhance the capabilities of ADMIRAL to take into account factors like different stock markets and different time zones, which can have critical affect on the prediction ability and should eliminate the assumption that the longer the time frame the better the possible prediction results by using different sets of time frames both consecutive and overlapping.

Eventually, we believe that this research study makes a significant contribution to the interaction of data mining and web content mining fields with real time financial problems faced by financial analysts and it demonstrates a first attempt to enhance the current methods.

VI. ACKNOWLEDGMENT

We wish to thank the research participants Liat Suissa, Orli Morad, Itai Oksenfeld and Eyal Solomon for their contribution to the Admiral's software coding and evaluation. This work was partially supported by the National Institute for Systems Test and Productivity at University of South Florida under the USA Space and Naval Warfare Systems Command Grant No. N00039-01-1-2248.

REFERENCES

[1] T. Bollerslev, Generalized Autoregressive Conditional Heteroscedasticity, Journal of Econometrics, 31, 307-327, 1986.

[2] T. Bollerslev, A Conditionally Heteroscedastic Time Series Model For Speculative Prices and Rates of Return, Review of Economics and Statistics, 69(3), 542-546, 1987.

[3] E.F. Fama, Random Walks in Stock Market Prices, Financial Analysts Journal, September/ October 1965 (reprinted in January-February 1995).

[4] G. Gidófalvi, 2001. Using News Articles to Predict Stock Price Movement, Online at: [http://citeseer.nj.nec.com/517027.html] .

[5] V. Lavrenko, M. Schmill, D. Lawrie, P. Ogilvie, D. Jensen and J. Allan, Language Models for Financial News Recommendation, CIKM 2000, McLean, VA USA, ACM 2000.

[6] M.A. Kaboudan, Genetic Programming Prediction of Stock Prices, Computational Economics 16: 207-236, 2000.

[7] E.F. Fama, Long Term Returns and Behavioral Finance, Social Science Research Network.

[8] M. Last, Y. Klein and A. Kandel, Knowledge Discovery in Time Series Databases, IEEE Transactions on Systems, Man and Cybernetics – Part B: Cybernetics, Vol. 31 No. 1, February 2001.

[9] Z. Bodie, A. Kane, A.J. Marcus, Investments, 4th Edition, McGraw Hill, 2001.

[10] E.F. Fama, Efficient Capital Markets: A Review of Theory and Empirical Work, Journal of Finance, 25 (May 1970): 383-417.

[11] E.F. Fama, Efficient Capital Markets: II, Journal of Finance, 46 (December 1991): 1575-1617.

[12] R.A. Huagen, The New Finance: The Case Against Efficient Markets. Prentice-Hall, 1995.

[13] R.Engle and T. Bollerslev, Modelling the Persistence in Conditional Variances, Econometric Reviews, 5, 81-87, 1986.

[14] R. Engle, Autoregressive Conditional Hetroscedasticity with Estimates of the Variance of United Kingdom Infation, Econometrica, 50(4), 987-1007, 1982.

[15] A.K. Jain, M.N. Murty, P.J. Flynn, Data Clustering: A Review, ACM Computing Surveys, Volume 31, No. 3, September 1999.

[16] R. Cooley, B. Mobasher and J. Srivastava, Web Mining: Information and Pattern Discovery on the World Wide Web, In Proceedings of the IEEE International Conference on Tools with Artificial Intelligence (ICTAI'97), Newport Beach, CA, November 1997.

[17] R. Kosala and H. Blockeel, Web Mining Research: A Survey, SIGKDD Explorations, Volume 2, Issue 1.

[18] O. Maimon, A. Kandel and M. Last, Knowledge Discovery and Data Mining, The Info-Fuzzy Network (IFN) Methodology, Norwell, MA: Kluwer, 2000.

[19] D. Peramunetilleke, R.K. Wong, Currency Exchange Rate Forecasting from News Headlines, Thirteenth Australasian Database Conference (ADC2002), Melbourne, Australia, Conferences in Information Technology, Vol. 5.

[20] B. Wuthrich, V. Cho, S. Leung, D. Permunetilleke, K. Sankaran, J. Zhang, W. Lam, Daily Stock Market Forecast from Textual Web Data, In IEEE International Conference on Systems, Man. and Cybernetics, Volume: 3, Page(s): 2720 -2725, 1998.

[21] L.Torgo, the TNT Financial Trading System: a midterm report, ECML-PKDD Workshop on Data Mining for Business, 2005.

[22] J.R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufman Publishers Inc., San Francisco, CA, 1993.

[23] R Landry Jr., R. Debreceny, G.L. Grey, Grab Your Picks and Shovels! There's Gold in Your Data, Strategic Finance, January 2004, (85, 7).

[24] Extractor DBI technologies (2003) [http://www.dbi-tech.com]