

AMCIS2004 Submission to Text and Data Mining for Decision Support¹

Balaji Rajagopalan, Prabhudev Konana, Chan-Gun Lee, Matt Wimple

Research In Progress

Extracting Relevance from Virtual Investing-Related Community Postings

Keywords: Sentiment extraction, virtual communities, text mining, message boards, online investing, genetic algorithms

ABSTRACT

The rapid growth of online investing and virtual investing-related communities (VICs) has a wide-ranging impact on research, practice and policy. In this context, this research addresses how information is generated, discussed, and diffused within and across VICs, and how such activities impact market efficiency. Regulators are particularly interested given the potential for fraud and spreading of false rumors. However, understanding information processing in VIC is a challenge given enormity of posted messages. Automated analysis of these messages is primarily complicated by three factors: (a) the amount of irrelevant messages or "noise" messages (e.g., spam, insults), (b) the highly unstructured nature of the text (e.g., abbreviations), and finally, and (c) the wide variation in what is considered relevant information for a given company. We have developed a mechanism relying on commonly occurring terms and a set of classifying criteria to identify: (1) "noisy" messages that bear no relevance to the topic at hand, (2) messages that have relevance to the topic at hand, but do not express an opinion as to the quality of the investment, and (3) messages that are both relevant and express a sentiment about the quality of the investment. To test our mechanism we have collected approximately 3 million messages related to 46 stocks over a 2-year period. Preliminary results show sufficient promise to classify messages and how participants react. Preliminary results show the classifier a classification accuracy of 54% .

¹ This research is supported by NSF ITR grant number IIS-0218988

Introduction

Over 20 million investors trade online in the U.S. The rapid growth is attributed to low costs, convenience, easy access to information, and control (Konana and Balasubramanian 2004). The increase in “do-it-yourself” investors is also associated with intense use of virtual investing-related communities (VICs) such as those on Yahoo!, and Morningstar. VICs provide platforms to seek, disseminate, and discuss stock-related information. Such information spreads rapidly to thousands of investors within and across virtual communities, and can influence the attitudes and decisions of these investors. VICs are particularly salient in the context of online investing, since investors who do not use human intermediaries are endowed with great control over their trading decisions.

VICs have both positive and negative attributes. They may reduce information asymmetry given the velocity at which new information can spread. However, VICs may amplify and propagate rumors or false information very quickly through the economic system with undesired effect. For example, Emulex Corporation and NEI Webworld Incorporated were subject to wild price swings that resulted from people disseminating false information. Past research has shown that message activity on VIC boards is correlated with trading volume and volatility (Wysocki 1999).

This research attempts to understand how information is created and diffused within VICs and its impact on the markets. This requires us to first identify information content within the messages. The challenges are numerous: the anonymity and ease of posting information are conducive for significant irrelevant messages or “noise.” Noise may come from insults, unsolicited advertisements (i.e. spams), and digressions. Further, it is critical to explicate the *sentiment* - the thought, view, or attitude, expressed in the message. Das and Chen [2002] were one of the first to attempt to extract the *emotive*² content in the messages. In this paper, we build upon their work and test a methodology to extract the relevance and sentiment of messages within the context of VICs. This technique, when refined and validated in different contexts, can be used within an interactive environment to inform users of the relevancy of the messages. It can also serve as tool for researchers interested in examining the content and impact of virtual community postings

Classifying the emotive content of messages posted on VICs poses several problems. The first problem is the nature of the classification itself. Messages can be “noise”, “perfectly relevant”, or “ambiguous”. The subjective nature of some messages may lead to disagreement among readers as to whether a given message is truthful, important, or reliable. For instance, it is common to find messages with postings “XYZ sucks” (XYZ refers to some stock symbol) without any elaboration. While it appears such messages would be noise the above message also seem to provide some useful information. Such problems have been encountered in previous text classification research. Foltz et al (1999) used classifiers for automated grading of student projects where there was disagreement among three human graders with a correlation of only 0.73. The classification of stock messages has other problems not observed in past text classification research.

Second, messages generally do not observe proper grammatical rules and spelling, and therefore, readability analysis measures on their own may be less effective. Users use abbreviations for many words (e.g., “u” for “you”, “L8er” for “Later”) and generally ignore spelling errors. Third, the

² *emotive* refers to attitude, view or thought based mainly on emotion instead of reason

problem is further compounded by semantic differences based on the context. For example in drug companies much of the conversation centers on potential products that are “in the pipeline” of research and development. However, in the energy sectors the term pipeline takes on a very different connotation. Each industry and company has rather different combinations of words that are often used within relevant conversations. Thus, finding a common set of words for classification is challenging.

Methodology

Previous studies have addressed the issue of classifying messages and extracting sentiment from stock message boards. Das and Chen (2001) classified message sentiment into three categories: positive, negative and neutral. Neutral message included spam messages or messages that are neither bullish nor bearish. Bullish or bearish on a particular stock were classified as positive or negative, respectively. Messages were classified based on five different algorithms and a voting scheme was used to combine them to arrive at a classification. They developed classifiers specifically for each stock to take into account the unique characteristics of the postings.

We adopt the general multi-algorithmic approach of Das and Chen (2001) in our study. However, we differentiate our classification method in several ways. First, we attempt to develop classifiers that are more *generic* – with applicability to a broad range of virtual investing-related postings as opposed to developing classifiers for individual stocks. Second, we propose to classify the messages along a different set of categories – Noise, Relevant, and Signal. We consider a message to be *noise* if the content is spam, or completely unrelated to message board topic. We consider a message to be *relevant* if the content relates to the stock in particular and/or the market in general with implications for the stock. We consider a message to be a *signal* carrier if and only if it is *relevant* and a discernable sentiment (positive or negative) is expressed toward the stock. As a comparison, Das and Chen (2001) focused on *Signal* and *Noise* only. Through the process of manually examining a random sample of several hundred messages we discovered the need for the third category – *relevant (but no signal content)*. Third, we design and develop an algorithm based on readability analysis, with theoretical foundations in reading and writing, to classify the messages. Fourth, we apply evolutionary computing methods – *genetic algorithms* to induce classification rule sets.

The methodology to extract relevance was carried out in four steps: Sample Selection and Preparation, Classifier Development, Testing & Validation, and Application. Sample selection involved random selection of messages (482 messages) across several message boards. Sample preparation was carried out by manual coding of each message into one of the three categories. Two graduate students were briefed on the criteria for classifying messages into the three categories. Inter-rater reliability (> 90%) of their classification indicated a high degree of consensus. The small number of messages that were classified differently by the students was revisited and a consensus reached regarding their categorization.

The second step of classifier development involved designing five classifiers based on different theoretical underpinnings. As mentioned earlier, this multi-algorithmic approach is consistent with earlier attempts [Das and Chen, 2001]. The five relevant extraction models – Lexicon-based Classifier (LBC), Readability-based Classifier (RBC), Weighted Lexicon Classifier (WLC), Vector Distance Classifier (VDC) and Differential Weights Lexicon Classifier (DWLC) are described in the

next section. A sixth classifier combining the outputs of each of the five classifiers was developed along the lines proposed by Das and Chen (2001).

The third step involved testing the classifier on a subset of the sample quarantined and not used for inducing the rule sets. Classification rates were then examined and the classifiers refined to improve relevant extraction accuracy. The final step involved applying the classification method to a larger data set and report the categorization distribution.

Relevance Extraction Models

In this section we detail the five classification mechanisms implemented for this study. We also detail a sixth classifier that effectively combines the output of the five classifiers to categorize the messages.

Lexicon-based Classifier (LBC): LBCs have been effectively used in earlier studies (Das and Chen, 2001). To design a LBC, we first developed a set of frequently occurring keywords (See Table 1) for the three categories – Noise (C_1), Relevant (C_2), and Signal (C_3). LBCs categorize a message m_l , where $l = \{1, 2, \dots, M\}$, by matching the message content against this set of keywords for each category and classifying the message as belonging to a category with the highest degree of matches.

Message m_l will belong to a category C_i if it has the $\max[n(k_i)]$ where $n(k_i)$ represents the number of keyword matches for the i^{th} category. In instances where two or more categories tie for $\max[n(k_i)]$, we choose lowest index i .

Formally, $\text{Category}(m_l) = C_i$ where i is the index for which $\sum \text{Count}(m_l, \text{Key}_{ij}) = \text{Max}_{k=1,2,3} \{ \sum \text{Count}(m_l, \text{Key}_{kj}) \}$, where Key_{ij} is the keyword j in the keyword list for Category i . $\text{Count}(m_l, \text{Key}_{ij})$ returns the number of occurrences of Key_{ij} in the message m_l . For example, $\text{Count}(\text{“I am too tired, too”}, \text{“too”})$ returns 2.

Table 1: Keyword Lists for Categories

Noise	Relevant	Signal
Keyword 1	Keyword 1	Keyword 1
Keyword 2	Keyword 2	Keyword 2
Keyword N_1	Keyword N_2	Keyword N_3

Readability-based Classifier (RBC): This classifier is based upon research on readability analysis. Though not expected to be a high performer as a sole classifier, it can be valuable when used in conjunction with other categorization methods. To develop RBC we randomly selected a subset of messages from the sample and used genetic algorithm (GA) to induce a rule set based on three variables: word count, mean word length, and number of unique words. In essence, the GA attempts to find the range of values for the three variables to define a category. The resulting “if

then...” decision tree is then applied to a test set for validation. This process is carried out iteratively to further refine the derived rule-set and improve performance.

Weighted Lexicon Classifier (WLC): This classifier, as its name suggests, is a variation of LBC. WLC overcomes the drawback of LBC which relies on absolute counts to categorize and hence, inducing a bias for categories with higher number of keywords by appropriately adjusting for it. To eliminate the keyword size bias, WLC bases its classification on $Max[\frac{n(k_i)}{N_i}]$, where N_i represents

the total number of keywords in category i . More formally,

$$\text{Category}(m) = C_i \quad \text{where } i \text{ is the index for which} \\ \sum \text{Count}(m_l, \frac{\text{Key}_{ij}}{N_i}) = \text{Max}_{k=1,2,3} \{ \sum \text{Count}(m_l, \frac{\text{Key}_{kj}}{N_k}) \}.$$

Vector Distance classifier (VDC): We implement VDC just as described in Chen and Das (2001). This method treats each message as a word vector in D-dimensional space where D represents the size of the keyword list. The proximity between a message m_l and grammar rule G_j is computed by the angle of the $Vector(m_l)$ and $Vector(G_j)$ where $Vector(V)$ is the D-dimensional word vector for V . Message m_l belongs to category of G_k for which the computed angle is the minimum, which means that the proximity is the maximum.

Formally, $\text{Category}(m) =$ the category of G_k where G_k is a grammar rule and k is the index for which

$$\text{Cos}(Vector(m_l), Vector(G_k)) = \text{Max}_{j=1 \dots \text{Sizeof(Grammar)}} \{ \text{Cos}(Vector(m_l), Vector(G_j)) \} \text{ where } \text{Cos}(V_1, V_2) =$$

$$\frac{V_1 \bullet V_2}{|V_1| |V_2|}$$

Differential Weights Lexicon Classifier (DWLC): This classifier represents another variation of the LBC. By assigning differential weights to each word in the lexicon (keyword list for each category) this mechanism recognizes the varying importance of each keyword in classification and overcomes the *equal weight bias* in LBC. In DWLC, message m_l will belong to category C_i if it has the $Max[\text{Weight}_{ij} \times n(k_{ij})]$, where Weight_{ij} represents the weight for the j^{th} keyword in the i^{th} category and $n(k_{ij})$ counts the number of keyword matches for the j^{th} keyword in the i^{th} category.

$$\text{More formally, } \text{Category}(m) = C_i \text{ where } i \text{ is the index for which} \\ \sum \text{Weight}_{ij} \times \text{Count}(m_l, \text{Key}_{ij}) = \text{Max}_{k=1,2,3} \{ \sum \text{Weight}_{kj} \times \text{Count}(m_l, \text{Key}_{kj}) \}.$$

A sixth classifier is designed by combining the outputs of the five classifiers using a *simple majority* voting mechanism. If we assume that each classifier categorizes (votes) message m_l as belonging to category C_j , then this combination classifier simply relies on the number of votes each message gets to determine which category the m_l belongs to. So for example, if three of the five classifiers voted for m_l belonging to C_1 , by simple majority principle message m_l is categorized as belonging to C_1 .

Results and Discussion

The results for all classifiers are presented in Table 2. Figure 1 presents a graphical representation of the performance of the classifiers.

Figure 1: Classifier Accuracy

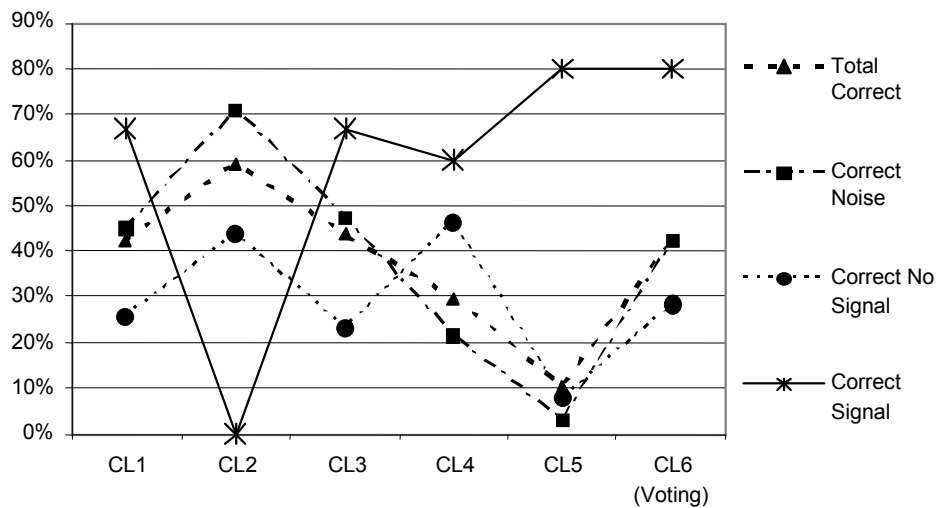


Table 2: Classifier Performance Statistics

Classifier	Total Correct	Correct Noise	Correct No Signal	Correct Signal
LBC	.4254	.4488	.2564	.6667
RBC	.5912	.7086	.4359	0
WLC	.4365	.4724	.2308	.6667
VDC	.2983	.2126	.4615	.6
DWLC	.1050	.0315	.0769	.8
Combined (Simple Majority Voting)	.4254	.4252	.2821	.8

We are in the process of analyzing the preliminary results and refining the classifiers. Some points of discussion we will present at the conference include: (a) Differential performance of a classifier over different categories, (b) Differential performance of classifiers within and across categories, (c) Challenges in developing and implementing the classifiers and (d) Enhancements to classifiers for future research.

Conclusions

The use of VICs is increasing as a primary source of information for do-it-yourself investors. Therefore, tools to automate messages based on noise, signal and sentiments have wide utility in

practice. The approach proposed is generic and has advantages over existing approaches. We are further refining our technique by improving word set, and integrating well-known algorithms for similar words matching, namely, “soundex indexing” and “edit distance.”

Bibliography

1. Antunovich, Peter and Laster, David S. (1998) “Do Investors Mistake a Good Company for a Good Investment?”, Federal Reserve Bank of New York.
2. Antweiler, Werner and Frank, Murry Z. (2004) “Is All That Talk Just Noise?-The Information Content of Internet Stock Message Boards”, *Journal of Finance* 59(3), June 2004, pages 1259-1295
3. Das, Sanjiv R. and Chen, Mike Y.”Yahoo! for Amazon: Sentiment parsing from small talk on the web.” Proceedings of the 8th Asia Pacific Finance Association Annual Conference, 2001.
4. Foltz, P. W., Laham, D. & Landauer, T. K. (1999). Automated Essay Scoring: Applications to Educational Technology. In proceedings of EdMedia '99.
5. Hearst, Marti A. 2000, “The Debate on Automated Essay Grading”, IEEE Intelligent Systems September/October 2000
6. McCallum, A., 1996, “Bow: A Toolkit for Statistical Language Modeling, Text Retrieval, Classification and Clustering,” School of Computer Science, Carnegie-Mellon University
7. Wosocki, P.D., 1999, “Cheap Talk on the Web: The Determinants of Postings on Stock Message Boards” Working Paper No.98025, University of Michigan