Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 2004 Proceedings

Americas Conference on Information Systems (AMCIS)

December 2004

Classification of Virtual Investing-Related Community Postings

Balaji Rajagopalan
Oakland University

Chan-Gun Lee
University of Texas at Austin

Matthew Wimble
Oakland University

Prabhudev Konana
University of Texas at Austin

Recommended Citation

Rajagopalan, Balaji; Lee, Chan-Gun; Wimble, Matthew; and Konana, Prabhudev, "Classification of Virtual Investing-Related Community Postings" (2004). *AMCIS 2004 Proceedings*. 237.
http://aisel.aisnet.org/amcis2004/237

Classification of Virtual Investing-Related Community Postings

Balaji Rajagopalan*
Oakland University
rajagopa@oakland.edu

Prabhudev Konana*
University of Texas at Austin
pkonana@utexas.edu

Matthew Wimble
Oakland University
mwwimble@oakland.edu

Chan-Gun Lee
University of Texas at Austin
cglee@utexas.edu

ABSTRACT

The rapid growth of online investing and virtual investing-related communities (VICs) has a wide-raging impact on research, practice and policy. Given the enormous volume of postings on VICs, automated classification of messages to extract relevance is critical. Classification is 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 relevancy for a given firm. We develop and validate an approach based on a variety of classifiers to identify: (1)"noisy" messages that bear no relevance to the topic, (2) messages containing no sentiment about the investment, but are relevant to the topic, and (3) messages containing sentiment and are relevant. Preliminary results show sufficient promise to classify messages.

Keywords

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

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.

As a first step in the research to understand how information is created and diffused within VICs, this study seeks to develop and validate a mechanism to automatically classify the VIC message postings. 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 (2001) were one of the first to attempt to extract the emotive 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.

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. 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 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

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 set of 3 categories – Noise, Relevant (also referred to as No Signal), and Signal. As a comparison, Das and Chen (2001) focused on Signal and Noise only. We consider a message to be *noise* if the content is spam, or completed unrelated to message board topic. We consider a message to be *relevant* (No Signal) 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. 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 a classifier 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. Fifth, Das and Chen (2001) did not classify every message, a significant portion of the messages were grouped into an unclassified category.

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 (> 80%) 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 reporting the categorization distribution.

Relevance Extraction Models

In this section we detail the five classification mechanisms implemented for this study. For three of the classifiers we tried two different approaches to analyze the same inputs. 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 for the three categories – Noise (C_1), Relevant (C_2), and Signal (C_3). LBCs categorize a message m_l , where l = {1, 2, ….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. We implemented two versions of the LBC. The first implementation used simple counts of keywords such that 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, Category(m_l) = C_i where *i* is the index for which $\sum Count(m_l, Key_{ij}) = Max_{k=1,2,3}\{\sum Count(m_l, Key_{kj})\}$, where Key_{ij} is the keyword *j* in the keyword list for Category *i*. $Count(m_l, Key_{ij})$ returns the number of occurrences of Key_{ij} in the message m_l. For example, *Count*("I am too tired, too", "too") returns 2.

In the second implementation, we induced a decision set to fit an "if…then" decision tree (Figure 1) using a genetic algorithm to optimize the total # correctly classified by encoding the solution set as a 10 element array, with the 10 elements being (a) the 3 variables to be used encoded as an ordinal number {1,2,3}, (b) the decision point represented by an integer bounded by the maximum and minimum values of the set, and (c) the results of the dependant "if…then" encoded as an integer {0,1,2} representing the message category.

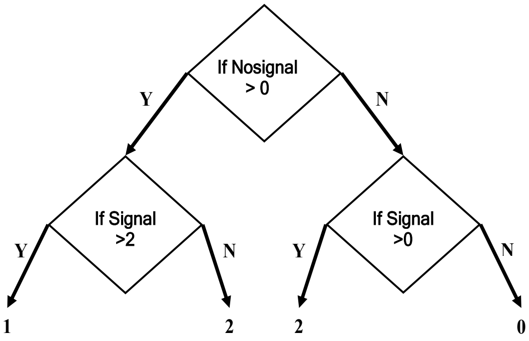

Figure 1: Induced decision set for LBC

Readability-based Classifier (RBC)

This classifier is based upon research on readability analysis. The initial rationale was that the RBC could detect noise if noisy messages were written more hastily than relevant messages. The idea was that somehow the degree of thought or haste would somehow be expressed in readability or "grade level" measurements. We then induced a decision set in the same manner described in the LBC section using word count, percentage of unique words, and number of unique words. The result was that two decision sets performed equally well on the training set and we elected to try both sets independently (Figure 2 & 3).

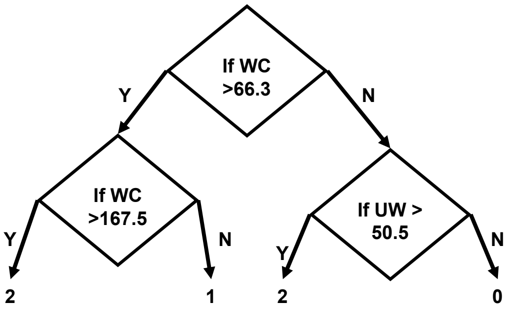

Figure 2. Induced decision set #1 for RBC

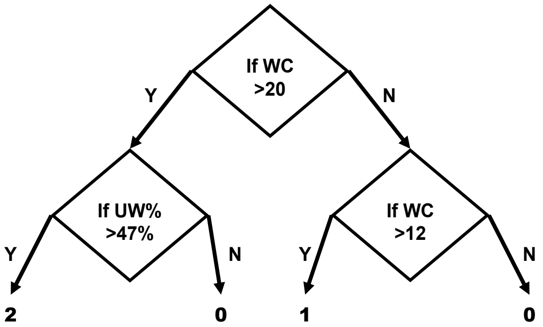

Figure 3. Induced decision set #2 for RBC

Weighted Lexicon Classifier (WLC)

This classifier, as its name suggests, is a variation of LBC. WLC overcomes a limitation of LBC that relies on absolute counts to categorize. In the case of WLC, this bias is adjusted for by taking the ratio of number of keywords in the message to message size. To eliminate the keyword size bias, WLC bases its classification on $Max[\frac{n(k_i)}{N_l}]$, where N_l represents the word count for message *l*. More formally, Category(m_l) = C_i where *i* is the index for which $\sum \frac{Count(m_l, Key_{ij})}{N_l} = Max_{k=1,2,3}\{\sum \frac{Count(m_l, Key_{kj})}{N_k}\}$. We then induced a decision set in the same manner described in the LBC section using the percentage measures (Figure 4).

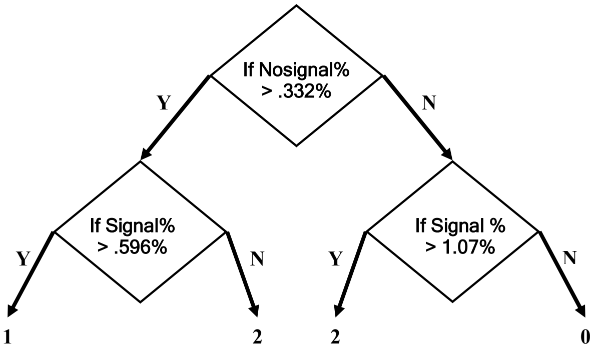

Figure 4: Induced decision set for PLC

Vector Distance classifier (VDC)

We implement VDC 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, Category(m_l) = the category of G_k where G_k is a grammar rule and *k* is the index for which

$Cos(Vector(m_l), Vector(G_k)) = \mathrm{Max}_{j=1\ldots\mathrm{Sizeof(Grammar)}}\{Cos(Vector(m_l), Vector(G_j))\}$ where $Cos(V_1, V_2) = \frac{V_1 \bullet V_2}{|V_1||V_2|}$

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

$Max[Weight_{ij} \times n(k_{ij})]$, 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.
More formally, Category(m_l) = C_i where i is the index for which $\sum Weight_{ij} \times Count(m_l, Key_{ij}) = Max_{k=1,2,3}\{\sum Weight_{kj} \times Count(m_l, Key_{kj})\}$. We then induced a decision tree (Figure 5) in the same manner described in the LBC section using the weighted measures.

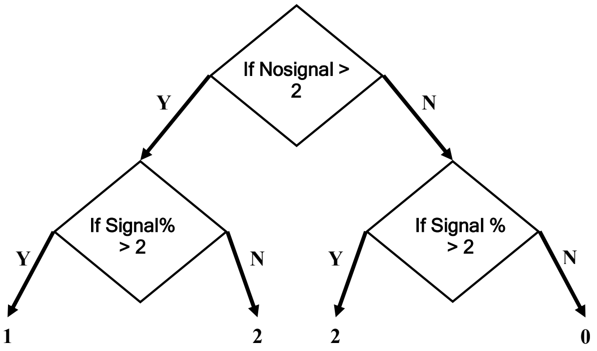

Figure 5: Decision set for DWLC

Combined Majority Voting Classifier (CVC)

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_i, 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

Classification results for LBC, WLC, and DWLC classifiers are presented based on two approaches - induced decision trees (Figure 6) and word-count based (Figure 7). We also present the results for the 2.7 million messages (Figure 8).

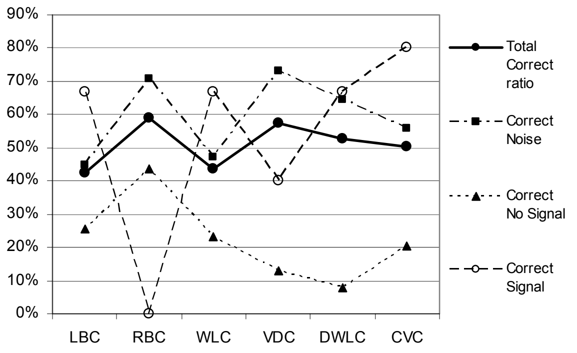

Figure 6: Classifier Performance (decision trees approach)

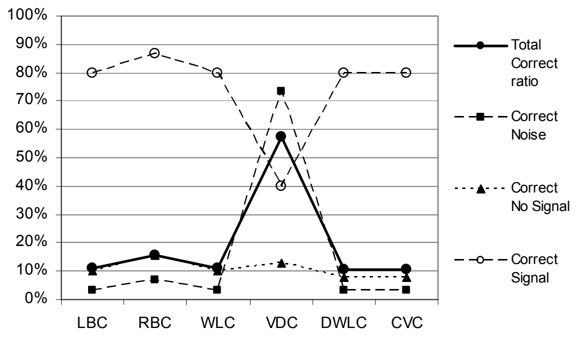

Figure 7: Classifier Performance (Word count approach)

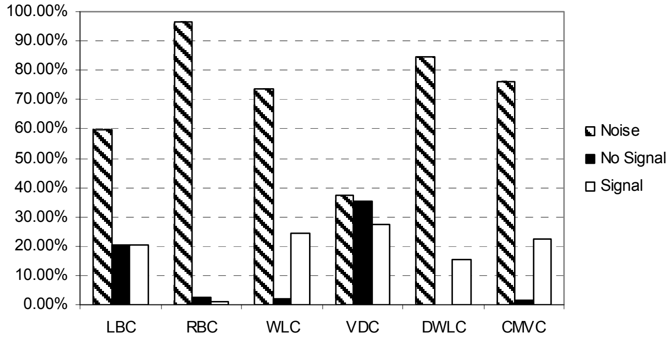

Figure 8: Extracting Relevance: Classification results for 2.7M messages

Performance Analysis

Overall, initial results show that a decision tree approach offers better performance than the word count when we have a limited keyword set. For the noise classification RBC, DWLC and VDC performed well. RBC was designed as a noise detection mechanism and results indicate utility for that purpose. The DWLC performed significantly better than the LBC or WLC and this suggests that some words, such as profanity, are stronger indicators of noise than others. Signal classification rates were highest for the RBC, but the results are misleading when only the correct percentage is considered since the RBC failed to classify any relevant but *no signal* messages. Classification accuracy for no signal messages was less than 30% for all classifier besides RBC. Poor performance for no signal messages is probably due to the difficulty of differentiating no signal from signal messages. Combined classifier performed well in identifying *signals,* but did relatively poorly on *no signal*.

Several design factors provide significant challenges to out approach they include a limited keyword list, inherent message ambiguity, increased complexity of 3-category problem and the forced classification of all messages. Based on this exploratory study potential improvements to the approach could include word list expansion, altering the voting mechanism, and more sophisticated word matching techniques

Conclusions

This study aimed at developing a classifier to automate extracting relevance from free text messages on stock bulletin boards. Preliminary results indicate sufficient promise for the proposed approach. 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."

ACKNOWLEDGMENTS

* This material is based upon work supported by the National Science Foundation (NSF) under Grant No. 0219107 to Balaji Rajagopalan and NSF grant 0218988 to Prabhudev Konana. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

REFERENCES

1. Das, Sanjiv R. and Chen, Mike Y. (2001)"Yahoo! for Amazon: Sentiment parsing from small talk on the web." Proceedings of the 8th Asia Pacific Finance Association Annual Conference, 2001.
2. Foltz, P. W., Laham, D. & Landauer, T. K. (1999). Automated Essay Scoring: Applications to Educational Technology. In proceedings of EdMedia '99.
3. Holland, J. (2000) "Building Blocks, Cohort Genetic Algorithms, and Hyperplane-Defined Functions", *Evolutionary Computation* 8(4): 373-391
4. Konana and Balasubramanian, "Social-Economic-Psychological Model of Technology Adoption and Usage: An Application in Online Investing.", Forthcoming, *Decision Support Systems* (2004).
5. Wysocki, P.D., (1999) "Cheap Talk on the Web: The Determinants of Postings on Stock Message Boards" Working Paper No.98025, University of Michigan