

KNOWLEDGE ACQUISITION, DELIVERY AND PREDICTION
THROUGH TEXT MINING

by

Robert P. Schumaker

A Dissertation Submitted to the Faculty of the
COMMITTEE ON BUSINESS ADMINISTRATION

In Partial Fulfillment of the Requirements
For the Degree of

DOCTOR OF PHILOSOPHY
WITH A MAJOR IN MANAGEMENT

In the Graduate College

THE UNIVERSITY OF ARIZONA

2007

THE UNIVERSITY OF ARIZONA
GRADUATE COLLEGE

As members of the Dissertation Committee, we certify that we have read the dissertation prepared by Robert P. Schumaker entitled Knowledge Acquisition, Delivery and Prediction through Text Mining and recommend that it be accepted as fulfilling the dissertation requirement for the Degree of Doctor of Philosophy

_____ Date: (April 19, 2007)
Hsinchun Chen

_____ Date: (April 19, 2007)
Zhu Zhang

_____ Date: (April 19, 2007)
Leon Zhao

_____ Date: (April 19, 2007)
Jay Nunamaker

Final approval and acceptance of this dissertation is contingent upon the candidate's submission of the final copies of the dissertation to the Graduate College.

I hereby certify that I have read this dissertation prepared under my direction and recommend that it be accepted as fulfilling the dissertation requirement.

_____ Date: (April 19, 2007)
Dissertation Director: Hsinchun Chen

STATEMENT BY AUTHOR

This dissertation has been submitted in partial fulfillment of requirements for an advanced degree at the University of Arizona and is deposited in the University Library to be made available to borrowers under rules of the Library.

Brief quotations from this dissertation are allowable without special permission, provided that accurate acknowledgment of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the head of the major department or the Dean of the Graduate College when in his or her judgment the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

SIGNED: Robert P. Schumaker

ACKNOWLEDGMENTS

A portion of this work was supported in part by the NSF, ITR: “COPLINK Center for Intelligence and Security Informatics Research – A Crime Data Mining Approach to Developing Border Safe Research,” Sept. 1, 2003 – Aug. 31, 2005.

For the Chatterbot studies, I would like to thank Dr. Mark Ginsburg and Ying Liu for their assistance in developing and evaluating this stream of research in a meaningful way. I would also like to thank Edna Reid and Larry Valida for their patience and insightful comments in taking the time to explain how technical research is conducted. I would also like to thank the MIS 111 instructors, Surya Pandruvada and Dennis Viehland for their assistance in using their students as test subjects. Finally, I would like to thank the following pilot testers: William Paul Burger, Jacquelyn Calvo, Alejandrina Gonzalez, Eric Impraim, Jennifer M. Lee, Kim Sierra and Kristen Smyser.

For the Stock Prediction studies, I would like to thank Lu Tseng for helping me to acquire those ever elusive intraday stock prices.

I would also like to acknowledge the patience and mentoring efforts of my advisor, Dr. Hsinchun Chen, as he labored to shape me into the researcher I have become.

DEDICATION

This dissertation is dedicated to my future wife, Amy, who has had the stamina and willingness to endure reading my papers and letting me know if they are comprehensible. I also dedicate a portion to my parents who have been ever supportive and helped guide me to where I am today as well as my sister who had the misfortune of correcting my grammar and teaching me the differences between its, it's and its'.

TABLE OF CONTENTS

LIST OF TABLES	10
LIST OF FIGURES	12
ABSTRACT.....	13
CHAPTER 1: INTRODUCTION	15
1.1 From Data to Wisdom	15
1.2 The DIKW Framework	16
1.3 Textual Web Mining	19
1.4 Relating DIKW to Textual Web Mining.....	20
1.5 Research Focus.....	21
CHAPTER 2: ACQUIRING KNOWLEDGE FROM WEB USERS	26
2.1 Background	26
2.2 Related Works	27
2.2.1 Approaches to Knowledge Acquisition	28
2.2.2 Dialog Systems	29
2.2.3 ALICE.....	31
2.3 Proposed Approaches	34
2.4 Experimental Design	38
2.4.1 Participants	39
2.4.2 Performance Metrics.....	41
2.4.3 Experiment setup	41
2.5 Experimental Results and Discussion	42
2.5.1 Users made corrections to nonsense responses	42
2.5.1.1 Nonsense responses.....	46
2.5.1.2 Spurious User Corrections	46
2.5.1.3 Wordy and awkward responses.....	47
2.5.1.4 Application or opinion type questions	47
2.5.1.5 Incomplete lexicon of terms.....	48
2.5.1.6 Nonsense responses.....	48
2.5.2 Knowledge Acquisition improved general conversation responses	48
2.5.3 Knowledge Acquisition improved telecommunication responses.....	49
2.5.4 Domain responses are more effective than Dialog	50
2.6 Conclusions	52
CHAPTER 3: ACQUISITION AND DELIVERY ON A TERRORISM DOMAIN.....	53
3.1 Background	53
3.2 Related Works	56
3.2.1 ALICE.....	58
3.2.2 Emergency Response Systems and the C3 Model.....	60
3.2.3 ALICEbots and C3	63
3.2.4 Literature Review Summary.....	64
3.3 Proposed Approaches	65
3.4 Experimental Design	71

TABLE OF CONTENTS--Continued

3.4.1 AIML Knowledge.....	71
3.4.2 Study Participants.....	73
3.4.3 Metrics.....	74
3.5 Experimental Results and Discussion.....	75
3.5.1 Users preferred Conversational to Definitional Responses.....	75
3.5.2 Dialog and Domain knowledge perform better together.....	79
3.5.3 Wh* Interrogatives were popular in user queries.....	80
3.6 Conclusions.....	82
CHAPTER 4: DELIVERING SMALL-SCALE KNOWLEDGE TO USERS.....	84
4.1 Background.....	84
4.2 Related Works.....	87
4.2.1 High-level Dialog Systems.....	87
4.2.2 Mid-level Dialog Systems.....	88
4.2.3 Low-level Dialog Systems.....	89
4.2.4 Challenges of Low-level Dialog System Analysis and Validation.....	91
4.3 Proposed Approaches.....	92
4.4 Experimental Design.....	95
4.4.1 Performance Metrics.....	96
4.4.2 Participants.....	97
4.5 Experimental Results and Discussion.....	98
4.5.1 Knowledge Delivery Accuracy was approximately 80%.....	98
4.5.2 Conversational Dialog Categories.....	100
4.5.2.1 Nonsense responses.....	100
4.5.2.2 Spurious User Corrections.....	100
4.5.2.3 Wordy and awkward responses.....	101
4.5.3 Telecommunications categories.....	101
4.5.3.1 Application or opinion type questions.....	102
4.5.3.2 Incomplete lexicon of terms.....	102
4.5.3.3 Nonsense responses.....	102
4.5.4 Users preferred Domain responses to Dialog.....	103
4.6 Conclusions.....	106
CHAPTER 5: DELIVERING LARGE-SCALE KNOWLEDGE TO USERS.....	108
5.1 Background.....	108
5.2 Related Works.....	109
5.2.1 Question Answer Systems.....	110
5.2.2 Domain Dependent and Independent Systems.....	111
5.2.3 Narrow and Open Domain Systems.....	112
5.2.4 Information Retrieval and Information Extraction.....	112
5.2.5 Document and Sentence-based IR Systems.....	113
5.2.5.1 ALICE.....	113
5.2.5.2 MegaHAL.....	114

TABLE OF CONTENTS--Continued

5.2.5.3 Other Chatterbot Studies	115
5.3 Proposed Approaches	117
5.4 Experimental Design	123
5.4.1 AIML Knowledge	123
5.4.2 Study Participants	124
5.4.3 System Evaluation Metrics	126
5.5 Experimental Results and Discussion	127
5.5.1 Participants sought definitional facts across knowledge domains.....	127
5.5.2 <i>Are</i> interrogatives had the highest Response Satisfaction rating.....	130
5.5.3 Response satisfaction scores show a weak relation to query length...	133
5.6 Conclusions	135
CHAPTER 6: MAKING PREDICTIONS FROM TEXTUAL KNOWLEDGE	137
6.1 Background	137
6.2 Related Works	138
6.2.1 Textual Representation	141
6.2.2 Machine Learning Algorithms.....	143
6.2.3 Financial News Article Sources.....	146
6.3 Proposed Approaches	148
6.4 Experimental Design	154
6.5 Experimental Findings and Discussion	156
6.5.1 Model M2 performed the best	157
6.5.2 A superset of Named Entities was the best textual representation.	159
6.6 Conclusions	164
CHAPTER 7: REFINING PREDICTIONS FROM TEXTUAL KNOWLEDGE	166
7.1 Background	166
7.2 Related Works	168
7.2.1 Fundamentalists and Technicians	169
7.2.2 Financial News Articles.....	170
7.2.3 Textual Representation	171
7.2.4 Industry Classification	174
7.2.5 Prediction Experts, Markets and Techniques	175
7.2.5.1 Prediction Experts	175
7.2.5.2 Prediction Markets and Techniques	176
7.2.5.2.1 Event-based Markets	177
7.2.5.2.2 Internal Markets.....	178
7.2.5.2.3 Commodities.....	179
7.2.5.2.4 Securities	180
7.2.5.2.4.1 Artificial Intelligence.....	181
7.2.5.2.4.2 Statistical Approach.....	183
7.2.5.2.4.3 Mathematical Analysis	183
7.3 Proposed Approaches	185

TABLE OF CONTENTS--Continued

7.4 Experimental Design	187
7.5 Experimental Results and Discussion	192
7.5.1 Sector-based training has the best performance:	192
7.5.2 Sector-based training outperforms Professional Traders & Quants: ..	197
7.6 Conclusions	200
CHAPTER 8: QUANTITATIVE STRATEGIES IN PREDICTION	202
8.1 Background	202
8.2 Related Works	204
8.2.1 Financial News Articles.....	205
8.2.2 Textual Representation	206
8.2.3 Quantitative Portfolio Building	209
8.3 Proposed Approaches	214
8.4 Experimental Design	216
8.5 Experimental Findings and Discussion	221
8.6 Conclusions	224
CHAPTER 9: CONCLUSIONS AND FUTURE DIRECTIONS	226
9.1 Contributions	226
9.1.1 Theoretical Contributions	226
9.1.2 Technical Contributions.....	227
9.1.3 Empirical Contributions.....	227
9.2 Relevance to MIS Research	228
9.3 Future Directions.....	229
APPENDIX A: AZFINTEXT STATE DIAGRAM	231
APPENDIX B: AZFINTEXT CLASS DIAGRAM	232
REFERENCES	233

LIST OF TABLES

Table 2.1. Differences between ALICE and AZ-ALICE	36
Table 2.2. Participant Breakdown.....	39
Table 2.3. Chatterbot User Corrections	43
Table 2.4. System Accuracy as a measure of Correction Rates.....	45
Table 2.5. Observed chatterbot response errors.....	45
Table 2.6. BaseBot Satisfaction and Correction Rates	49
Table 2.7. TeleComm Satisfaction and Correction Rates.....	50
Table 2.8. Comparing dialog to domain for the chatterbots	51
Table 3.1. Differences between ALICE Program D and TARA	66
Table 3.2. AIML distribution for the three chatterbots.....	71
Table 3.3. Analyzing User Behavior towards the system.....	76
Table 3.4. User comments about comfort using the system	76
Table 3.5. User comments about system recommendation.....	78
Table 3.6. User behavior towards the system	78
Table 3.7. Response appropriateness across three chatterbots	79
Table 3.8. Response rating statistics	79
Table 3.9. The ‘Both’ chatterbot’s dialog and domain responses.....	80
Table 3.10. The most popular interrogatives used.....	81
Table 4.1. Differences between ALICE ProgramD and AZ-ALICE.....	94
Table 4.2. The category breakdown between systems.....	96
Table 4.3. Study Participants by System	98
Table 4.4. Interaction Breakdown.....	98
Table 4.5. System Accuracy as a measure of Correction Rates.....	99
Table 4.6. Observed chatterbot response errors.....	100
Table 4.7. Response Satisfaction scores	103
Table 4.8. Abbreviated view of the Corrected Responses of Table 4.5.....	104
Table 5.1. Difference between original ALICE, AZ-ALICE and TARA.....	122
Table 5.2. Category Breakdown of the Six Chatterbots	123
Table 5.3. Study Participants	125
Table 5.4. Participant usage of interrogatives across chatterbots	127
Table 5.5. Interrogative frequency use of Telecommunications.....	128
Table 5.6. Interrogative frequency use of Terrorism	128
Table 5.7. <i>What</i> expansion frequency counts of Telecommunications	129
Table 5.8. <i>What</i> expansion frequency counts of Terrorism.....	129
Table 5.9. Interrogative Response Satisfaction scores for Telecommunications	130
Table 5.10. Interrogative Response Satisfaction scores for Terrorism.....	130
Table 5.11. Response Satisfaction breakdown of Both in Telecommunications.....	131
Table 5.12. Response Satisfaction breakdown of Both in Terrorism	132
Table 5.13. Query lengths across chatterbots	133
Table 5.14. Query lengths for each interrogative in Telecommunications.....	133
Table 5.15. Query lengths for each interrogative in Terrorism	134

LIST OF TABLES--Continued

Table 5.16. Relating query lengths to response satisfaction scores	134
Table 6.1. Taxonomy of prior algorithmic research	143
Table 6.2. Taxonomy of textual financial data	147
Table 6.3. Closeness results	156
Table 6.4. Directional Accuracy results.....	156
Table 6.5. Simulated Trading Engine results	156
Table 6.6. Closeness results	160
Table 6.7. Directional Accuracy results.....	160
Table 6.8. Simulated Trading Engine results	160
Table 6.9. Closeness results	162
Table 6.10. Directional Accuracy results.....	162
Table 6.11. Simulated Trading Engine results.....	162
Table 7.1. Basic statistics on the AZFinText datasets	190
Table 7.2. Recommendation breakdown of each Trading Professional	191
Table 7.3. Three metric evaluation of the different GICS levels.....	192
Table 7.4. Sector breakdown and evaluation	195
Table 7.5. Company components of the Consumer Discretionary Sector	196
Table 7.6. Company components of the Utilities sector	197
Table 7.7. Comparison of Directional Accuracy results.....	198
Table 7.8. Simulated Trading results of Professionals and Quants	199
Table 7.9. Simulated Trading results of S&P 500 quants.....	199
Table 8.1. Trading Returns for Momentum Strategies	221
Table 8.2. Trading Returns for Contrarian Strategies.....	223

LIST OF FIGURES

Figure 1.1. DIKW Framework and Example.....	16
Figure 1.2. The Intersection of Textual Web Mining.....	20
Figure 2.1. Screenshot of AZ-ALICE evaluation.....	36
Figure 2.2. Experimental Design Testing.....	42
Figure 3.1. Diagram of User Input to Chatterbot Response.....	67
Figure 3.2. Screenshot of TARA's evaluation process.....	73
Figure 5.1. A Question Answer system taxonomy.....	110
Figure 5.2. Graphic Illustration of Chatterbot Components.....	119
Figure 5.3. Example AIML node map.....	121
Figure 5.4. Screenshot of TARA's evaluation process.....	126
Figure 6.1. AZFinText system design.....	150
Figure 6.2. Example AZFinText representation.....	153
Figure 7.1. Prediction Market Exchanges.....	177
Figure 7.2. AZFinText system design.....	186
Figure 7.3. Example of AZFinText Processing.....	189
Figure 7.4. Histogram of Trade Returns.....	194
Figure 8.1. A Taxonomy of Quantitative Portfolio Building.....	209
Figure 8.2. AZFinText system design.....	215
Figure 8.3. Four models to analyze.....	218
Figure 8.4. Example of AZFinText Processing.....	220
Figure A.1. State Diagram of AZFinText.....	231
Figure B.1. Simplified Class Diagram of AZFinText.....	232

ABSTRACT

The World Wide Web is an abundant source for Textual Web Mining research. Data can be acquired from Web texts and converted to Information or Knowledge for immediate consumption. Studying the acquisition and consumption of Web text can provide a glimpse into the social/behavioral aspects of Web Users and Web Content Providers. Patterns embedded within textual data can be similarly identified through technical means and even anticipated.

Seven essays explore the important algorithmic and computational aspects needed in the analysis of acquiring, delivering and making predictions from Web texts. Chapters 2 and 3 describe the knowledge acquisition process and feasibility of leveraging Web users. While the knowledge acquired from Web users was not as refined as that from domain experts, the knowledge gathered was found to be of acceptable quality. From our analysis of dialog systems, it was found that Web users were more likely to augment the breadth of existing knowledge by adding new response sets to the knowledge base. Chapters 4 and 5 look at the aspects of knowledge delivery to Web users. Using a dialog system, we observe the acceptance and satisfaction levels of dialog responses in general conversation, domain knowledge and the combination of both knowledge bases. Chapters 6 through 8 consider the prediction facet of knowledge using textual financial news articles and stock prices. This section focuses on comparing different model parameters and textual representations to best describe future prices as well as an examination of document representation based on the sector and industry a company is

engaged in. From these analyses we found that Sector-based aggregation led to the best price predictions.

Together these essays effectively leverage large amounts of textual Web data to represent knowledge in meaningful ways to end users. These essays also provide the blueprints for several real-world applications. The approaches and techniques described borrow from referent disciplines of linguistics, finance, computer science, statistics as well as MIS and demonstrate potentially useful applications for dialog systems, quantitative stock prediction and other knowledge management processes in which textual data can be accurately represented and forecast; thus improving the exchange of human knowledge.

CHAPTER 1: INTRODUCTION

1.1 From Data to Wisdom

Knowledge signifies things known. Where there are no things known, there is no knowledge. Where there are no things to be known, there can be no knowledge. We have observed that every science, that is, every branch of knowledge, is compounded of certain facts, of which our sensations furnish the evidence. Where no such evidence is supplied, we are without data; we are without first premises; and when, without these, we attempt to build up a science, we do as those who raise edifices without foundations. And what do such builders construct? Castles in the air (Wright, 1829).

These are the words of the Scottish-born Frances Wright in her 1829 book, Course of Popular Lectures. While Polanyi might argue that this description of knowledge signifying things that are known, ignores tacit knowledge (Polanyi, 1997), Wright's words of warning could easily be applied to all academic disciplines. This poetic linking together of knowledge, facts and data occurred over one-hundred and fifty years before Ackoff laid the foundations for the familiar DIKW (Data-Information-Knowledge-Wisdom) hierarchy (Ackoff, 1989).

The DIKW hierarchy is widely accepted in knowledge management circles as a way to represent the different levels of what we see and what we know (Cleveland, 1982; Zeleny, 1987). With each successive level relying upon the previous, deeper levels of

the hierarchy, this model provides an increasing awareness of the surroundings (Carlisle, 2006) where meaning can be found in the organization-wide continuum of data, information, knowledge and even wisdom (Chen, 2001). By correctly identifying and applying the DIKW framework, perhaps we can avoid Wright's "Castles in the air."

1.2 The DIKW Framework

Figure 1.1 illustrates Alavi's linear interpretation of the DIKW Framework (Alavi and Leidner, 2001) and provides an example of how it functions.

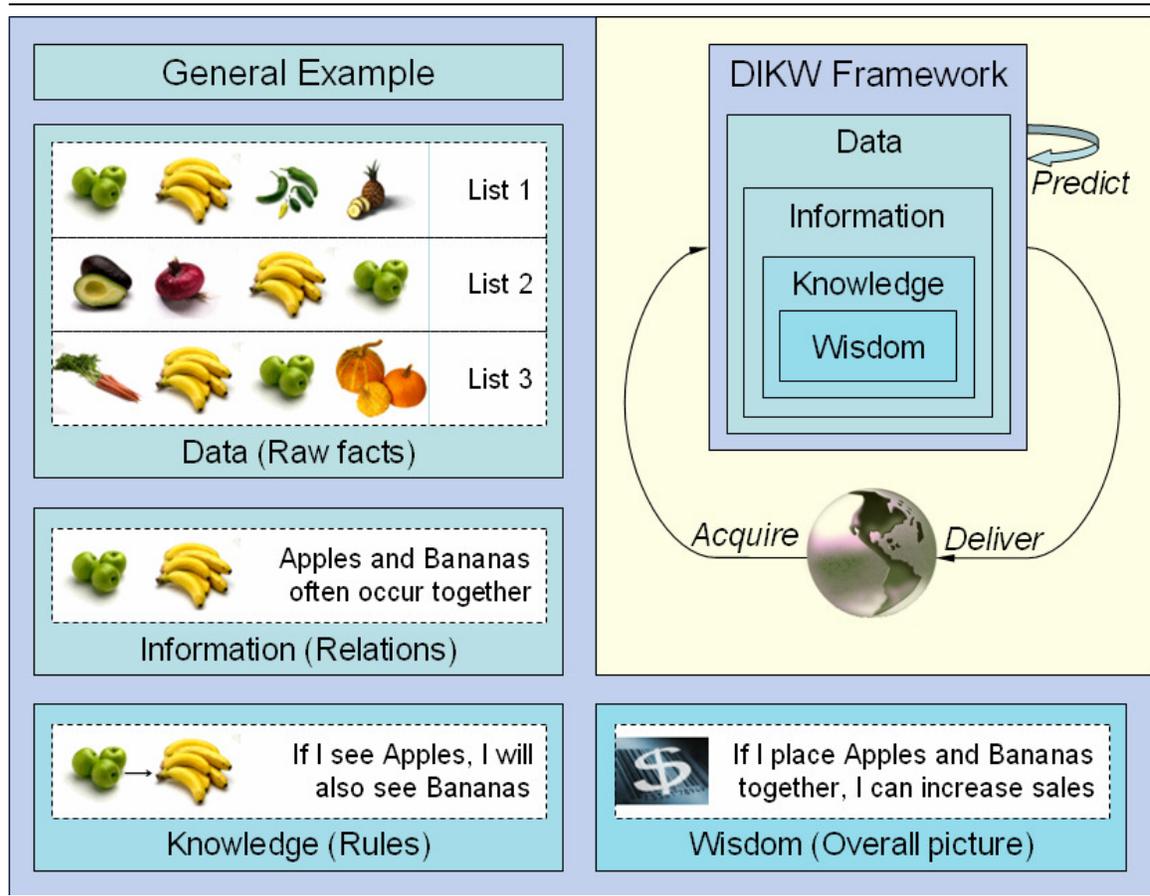


Figure 1.1. DIKW Framework and Example

While Alavi's portrayal of the DIKW framework as independent boxes arranged in a linear fashion has been the subject of debate (Tuomi, 1999), the DIKW framework present in Figure 1.1 represents a more unifying view that each level is interwoven (Han and Chang, 2002; Hildreth and Kimble, 2002; Stenmark, 2002; Carlisle, 2006). However, it is still a commonly held view is that each level of the DIKW framework is derived from its immediate predecessor (Davenport and Prusak, 1998). Below we define each element of the framework.

Data, the observable differences in physical states (Boisot and Canals, 2004), is acquired from stimuli and careful inspection of the world around us. Data by itself is generally overwhelming in volume and not entirely usable. In the example of Figure 1.1, data comes in the form of consumer grocery buying habits. In order to be of practical value, data must be transformed by identifying relationships (Barlas, Ginart et al., 2005) or limited to only that which is relevant to the problem at hand (Carlisle, 2006). This transformation of content gives us Information.

While the value of Information may depend on its timeliness, accessibility, reliability and availability (Chen, 2005), it can be similarly argued that its value is also based on a particular user's need (Choo, 1996). In its essence, Information can be construed as meaningful, useful data (Bierly, Kessler et al., 2000). From the example of Figure 1.1, we gain Information from the observation that apples and bananas occur frequently together. Although this relation is not entirely useful at this stage, abstracting it to the next level of the hierarchy, Knowledge, can provide us additional meaning.

Knowledge is the aggregation of related Information (Barlas, Ginart et al., 2005), that forms a set of expectations or rules (Boisot and Canals, 2004) which provides a clearer understanding of Information (Bierly, Kessler et al., 2000). This level of the hierarchy begins the formation of rule-based systems which can allow individuals to expand their own knowledge while also benefiting the organization (Alavi and Leidner, 2001). In Figure 1.1, we can form an associative rule that links together apples and bananas.

The interweaving of data, information and knowledge permits the extrapolation of different levels in the hierarchy. Earlier instantiations of DIKW prohibited backwards movement (e.g., acquiring information from knowledge), however, modern research has questioned this assumption. Possessing knowledge can allow a user to derive information or even data (Stenmark, 2002) which stands in contrast to Alavi's original design. If there is knowledge that apples and bananas are linked together, then assumptions can be made regarding the composition of data.

While the precise definitions of data, information and knowledge are still a matter of debate; wisdom can be viewed as a grasp of the overall situation (Barlas, Ginart et al., 2005), that uses knowledge and knowledge alone (Carlisle, 2006) to achieve goals (Bierly, Kessler et al., 2000; Hastie, Tibshirani et al., 2001). In Figure 1.1, wisdom can be depicted as the realization that increasing profits (our goal) can be obtained by cross-merchandizing two products that have a relation in consumer buying habits. Uncovering this truth rests in the capabilities of cognition and human understanding (Carlisle, 2006), as a computational wisdombase is currently difficult to imagine (Barlas, Ginart et al.,

2005). It is this incorporation of understanding that currently sets the divide between man and machine.

“Wisdom is the principal thing, therefore get wisdom; and with all thy getting, get understanding” (Proverbs 4:7 KJV).

1.3 Textual Web Mining

With the advent of cheaper computer storage and interoperating networks, access to sources of data, information and knowledge has become readily available. Peter Drucker has characterized this period as the Knowledge Economy where the management of an organization’s knowledge becomes a tool of competitive advantage (Drucker, 1969). Given the vast amount of content available, the Internet and especially the Web, is an abundant source from which we can extract valuable knowledge. Unfortunately, these sources are often unstructured and full of irrelevant material. While the act of finding data has become much easier, finding clean and well-organized data has been a challenge.

In order to address the difficulties of searching for relevant data, the science of Information Retrieval (IR) was created to sift through documents and databases to reduce information overload by returning those results that most closely match the query (Wikipedia, 2007) which include search engines and question-answer systems. However, for the purposes of this dissertation, we will be focusing on a sub-area of Information Retrieval, Data Mining.

Data Mining involves procedures to uncover hidden trends and develop new data and information from previous data sources. These sources can include well-structured and defined databases or the more common form of unstructured texts. While most Web-based communication is textual data, finding information or knowledge within these Web documents can be of strategic value to the Knowledge-based organization and gives rise to the area of Textual Web Mining.

Figure 1.2 shows the relation between Textual Web Mining and its parent disciplines.

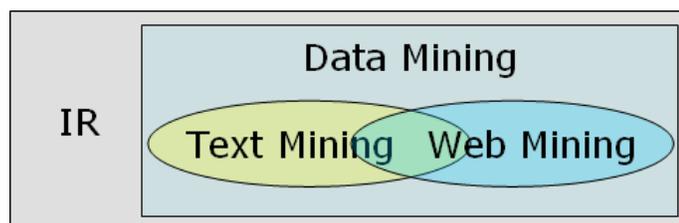


Figure 1.2. The Intersection of Textual Web Mining

Textual Web Mining is the application of Data Mining techniques to extract knowledge from natural language sources of Web data, including Web documents, Web users and other Web data repositories (Hearst, 1999; Zhong, Liu et al., 2002). This stream of research is a subset of Information Retrieval and uses Artificial Intelligence techniques to acquire and anticipate relevant information to meet specific domain needs.

1.4 Relating DIKW to Textual Web Mining

The exploration of the DIKW framework serves as an introduction to the topic of this dissertation; mainly the focus on external interactions allowing the acquisition, delivery and prediction of textual Web knowledge. While it can be argued that other elements of

the DIKW framework can be acquired, delivered or even predicted from, this dissertation seeks to limit the scope to focus solely on knowledge domains.

Knowledge Acquisition is the process of obtaining relevant and concise content from Web sources. From Figure 1.1, knowledge is acquired or transformed from external Web sources, such as Web documents, Web users and other Web resources.

Knowledge Delivery is the process of returning relevant and concise content to Web users. Figure 1.1 illustrates the return of knowledge back to the external environment. This process involves Information Retrieval tasks in order to identify the appropriate knowledge requested.

Knowledge Prediction is the process of projecting the trends and tendencies contained within existing knowledge. In Figure 1.1, knowledge can be used in the creation of further knowledge. By leveraging existing knowledge sources and sound algorithms, knowledge creation can become a valuable business tool.

This synthesis of Knowledge Acquisition, Delivery and Prediction forms a circle of knowledge (Mazzotta, 1993) where Web users can create, distribute and receive knowledge in a relevant and concise manner.

1.5 Research Focus

The ability of an organization to leverage its knowledge assets is key to strategic growth. This dissertation focuses on new applications of existing knowledge and its resources. Its guiding focus is the development of organizational tools that create, retrieve, deliver and project new knowledge from existing stores (Choi, Jung et al., 2004).

We place special emphasis on building domain-specific systems to best represent the knowledge within its domain.

Chapter 2 of this dissertation focuses on the Knowledge Acquisition aspect of Natural Language Dialog Systems (NLDSs) from Web users. Traditional studies have focused on using pattern-matching techniques to extract known knowledge from predefined and typically well-vetted textual sources. The approach we selected acquired Telecommunications knowledge directly from Web users for later inclusion in the system's knowledge bases. While some studies have expressed caution over the untrustworthiness of using such sources, we found that our approach netted satisfactory knowledge and that the intake of inappropriate and misleading knowledge was much less than expected. We then returned the gathered Web knowledge to a second group of Web users and found that its inclusion produced a significant improvement to the knowledge bases.

Chapter 3 focuses on large-scale Knowledge Acquisition from reputable Web sources, more particularly the usability of Terrorism knowledge through an NLDS system. This chapter is positioned as a Terrorism response system that can be deployed in the event of a serious terrorism event to fill the information vacuum. Immediately following the attacks of Sept. 11, information was at a premium. Aside from problems of different emergency management groups being unable to communicate; issues of addressing public concerns was largely secondary which led to feelings of frustration and helplessness. Citizens wanted to know what was happening in real-time, especially if a loved one of theirs had been recovered. This chapter fills the gap by introducing the AZ-

ALICE system which can be deployed in the aftermath of a terrorist event and can communicate in natural language conversations with the public, assuaging any concerns they may have. Coupled with the ability of emergency management and governmental personnel to update the system in real-time, AZ-ALICE can fill the information vacuum and provide immediate answers. In our study we focused on what types of knowledge that Web users would be most likely to request and found that answering definitional types of questions were most prevalent and should be the emphasis of any knowledge-gathering activity.

Chapter 4 moves into small-scale Knowledge Delivery and studies the reactions of Web users to proportionally smaller domain knowledge bases than conversational dialog. In this study we varied the amount of conversational and Telecommunications knowledge to form three groups; Full conversation without any Telecommunications knowledge, Full Telecommunications knowledge without any conversation and Full Telecommunications knowledge with limited conversation. By evaluating the Web users' satisfaction with system responses and their willingness to correct certain responses, we found that Web users preferred Telecommunications knowledge over conversational dialog. Grouping the commonly cited shortcomings of conversational knowledge together, it was found that most system response errors arose from nonsensical replies and wordy, awkward responses. Given the scope of this study, it may be an indication that concise definitional responses were preferred.

Chapter 5 is the large-scale Knowledge Delivery component and consists of two pieces; the impact of textual delivery and social acceptance. As the next step in

knowledge discovery, information gathered from the Web needs to be delivered and its effectiveness measured. Using methodologies borrowed from human computer interaction and behavioral science, we formulate methods to measure the satisfaction and usefulness of specific information within restricted domains. From this study, it was found that specific knowledge was preferred to general knowledge when present in significant quantity.

Chapter 6 transitions to the Knowledge Prediction component and is a more technical investigation of generalizing patterns learned from textual web sources and making predictions from them. This chapter fuses natural language processing, machine learning techniques and textual financial news sources together to make discrete ephemeral predictions of stock prices. By looking at financial news articles, some terms contribute to either upward or downward stock price movement. Identifying the relevant terms that move stock prices and then predicting a future value for them was found to be obtainable.

Chapter 7 examines the area of stock prediction in more detail. Building upon the system developments in Chapter 6, we seek to investigate grouping textual financial news articles according to their peer organizations instead of using either all financial news articles or only those specific to a certain company. This chapter also compares our system against existing human traders and quantitative stock picking algorithms and found that using textual financial news articles has a decided advantage to traditional techniques.

Chapter 8 further extends the AZFinText trading engine by incorporating it with some well-known and well-studied quantitative stock selection strategies. Comparisons

between AZFinText's news only approach are compared against solo quantitative strategies and hybrid strategies using both the news analysis and rudimentary quantitative strategies. Analysis is also performed on the impact of external factors such as analyst upgrades/downgrades and insider transactions on AZinText's stock prediction. Through our analysis of ten different hybrid AZFinText instantiations, we found that the hybrid strategies statistically outperformed quantitative-only returns and that utilizing a momentum strategy with one-week portfolio formation led to the best trading returns. Similarly, it was also found that news articles immediately preceding analyst recommendations and certain insider trades are also more predictable than chance.

Chapter 9 of this dissertation finishes up by highlighting the major research contributions, the relevance of this research to MIS and future directions that these studies can pursue.

CHAPTER 2: ACQUIRING KNOWLEDGE FROM WEB USERS

This chapter examines the process of gathering domain-specific knowledge Web Users and then evaluating the efficacy of such sources within a natural language conversational environment.

2.1 Background

The process of knowledge acquisition is to transfer existing knowledge and its structure into a computer-interpretable form (Potter, 2001). This knowledge can come from humans or other sources such as textual documents or encyclopedias. When coupled with the Internet, knowledge acquisition inherits new problems of scale such as information quality and reliability issues.

This chapter investigates the knowledge acquisition activities of a chatterbot program that mimics human conversation. Web-based chatterbot systems can provide an easy, natural extension to knowledge acquisition. This style of dialog system, due to its robustness, scalability and ease of connecting to the Web for information retrieval, appears to be a viable approach for knowledge acquisition. One of the better performers in the field is the ALICEbot. ALICE, or Artificial Linguistic Internet Chat Entity, was developed by Richard Wallace in 1995. This system has had marked success, winning the Loebner Prize for most human-like computer in 2000, 2001 and 2004. It has also begun to position itself as a personable customer-service application through 3D modeling and an on-screen character presence which can better interact with users (Chavez, 2006).

This chapter is laid out as follows. Section 2.2 will investigate the existing literature going from knowledge acquisition and its many approaches, down to the ALICE chatterbots and how they fit into the framework. In Section 2.3, we introduce a set of research questions and explain the system design implemented in our study. Section 2.4 looks at the experimental design in detail. Section 2.5 describes the results from the experiment and offers a discussion of their meaning. Finally, in Section 2.6, conclusions and future directions are provided.

2.2 Related Works

Knowledge acquisition has been a sought after goal since the early days of Artificial Intelligence. Newell posited that psychology and structure are important elements to perform a sequence of complex tasks and noted the similarities between cognitive tasks and existing programming languages. These languages are further engineered to use logic and conditional operators (Newell, 1973) to mimic human ability and to simulate human behavior (Feigenbaum and Simon, 1962).

Under the broad umbrella of knowledge acquisition there are several approaches; those that are manual and those that are automated or semi-automated. Some of the more common manual approaches are; the expert interview where knowledge is captured from a human expert through an interview-style process and protocol analysis where an expert's thought process is analyzed step-by-step in solving a particular problem (Potter, 2001). The strengths of using these processes rest in the quality of information gained. However, the process of acquiring knowledge using these methods is laborious and time-

consuming. The automatic or semi-automatic methods to knowledge acquisition attempt to automate all or some part of the acquisition process in a human-like fashion using models or examples to learn from (Lavrac and Mozetic, 1992; Potter, 2001). The draw of using such an approach is to decrease the amount of time needed to capture knowledge. These approaches work best when dealing with highly structured knowledge, but this approach does not generally increase information quality and it is difficult to qualify what a system 'should have' learned.

2.2.1 Approaches to Knowledge Acquisition

The automated or semi-automated approaches have several sub-approaches; pattern and template matching, machine learning and textual data mining (Lavrac and Mozetic, 1992; Potter, 2001). In pattern and template-matching, the natural language input is matched against some preconceived template where the response is pre-coded and dictated by the matching template. This type of system is simple, scalable, easy to develop and has a good performance history. The drawback to this approach is the amount of time required to acquire initial knowledge and place in a structured template. For machine learning, where the system attempts to identify concept pairs or co-occurrences, the system can handle exceedingly large data sets. However, this system requires a priori knowledge of the domain and the input must be in a highly structured and machine-friendly style. In textual data mining, new knowledge can be discovered by unearthing previously unknown relations using data mining techniques. However, the

new knowledge in a conversational setting will generally require user guidance to format it in a semantically correct context.

2.2.2 Dialog Systems

For the pattern and template matching approach, there are several sub-categories as well. The first of which is the trigger approach where the system identifies specific trigger phrases within the natural language input. This method lends itself to a high degree of precision, however, it is not very robust and does not handle conflicts between triggers very well. The second method is that of the semantic approach. In this system, a context-free environment is maintained while focusing only on the word meanings themselves. The downside to this method is that it is not very scalable when pursuing new knowledge domains. In the third approach, syntactic analysis is used to acquire knowledge, typically from highly structured technical texts. This approach can induce new knowledge, but will ignore unrecognized words or patterns. The fourth approach is that of the Eliza-type chatterbot, which played the role of a Rogerian psychotherapist to pseudo-extract information from its patients (Weizenbaum, 1966). With Eliza, it became possible to carry on generalized conversations in a reasonable manner. The weakness fell on the exceedingly limited initial set of patterns that Eliza had at its disposal.

Following Eliza, a fundamental split of theory occurred when researchers began concentrating of two different methodologies; the theoretically motivated models and the performance led systems (Simmons, 1970; Russell, 2002). The theoretical models pursued directions in symbolic reasoning and deep understanding systems. An example

of such research is Winograd's "Blocks World" where the artificial environment played a key role in shaping the program's direction. Performance led systems were quite the opposite and became increasingly more interested in obtaining an answer with speed rather than distract themselves with understanding. The favored approach is to omit syntactic analysis in favor of simple pattern matching tactics (Russell, 2002; Vrajitoru, 2003). Some of the more interesting research in this area came by way of Parry and the ALICEbot family.

Parry (Colby, Weber et al., 1971) was one of the chatterbot pioneers in the performance led systems and expanded upon the Eliza framework. Parry, a variant of Eliza, was a robust entity with many more patterns than its predecessor. However, Parry was restricted to a limited domain of paranoid hospital patient expressions. This under-generalized set of expressions were a limitation, however, in the context of insane hospital patient responses, those interacting with the system accepted Parry's responses even if the responses were non-transferable to a more generalized domain.

Another performance-led system that investigated the aspects of social interaction was Cobot (Isbell, Kearns et al., 2000). Cobot, a part of the MUD community called LambdaMOO, interacts with others either by recognizing certain commands tied to its statistical information gathering (i.e., 'who loves me' and 'who acts like me') or by conversational pattern-matching techniques. This process of learning and interaction has allowed others to socially accept Cobot.

The ALICEbot family (Wallace, 2004) is one of the more recent additions to the field, utilizing pattern-matching techniques without a syntactic overhead. These series of

systems are considered by some researchers as robust, scalable and convincing conversationalists. The downside is that expanding into new knowledge domains can require some laborious programming. Within the ALICEbot family is Converse, Cartman and ALICE (Russell, 2002).

Converse, which won the Loebner Prize in 1997 for most human-like computer (Anonymous, 2003), uses scripts and outside corpora such as WordNet to form responses. The problem with this system is that knowledge expansion requires additional scripts, as well as the reliance on a single exterior source of knowledge.

Cartman, the next notable member of the ALICEbot family, was developed as a dialog agent to tutor students. The benefits to this system are in its flexible, scalable nature and the lack of client-side software to operate. The weakness, however, is that it was not developed as a general conversationalist and instead operates in particular domains of interest.

2.2.3 ALICE

ALICE uses XML knowledge-bases to match user input against a predefined response set. The shortcoming of this system is that it cannot adequately answer all of the queries given to it. It is contended that ALICEbots have no cognitive theory behind them and instead blindly rely on canned responses to matched inputs (Russell, 2002). However, it can be similarly argued that ALICEbots use Case Based Reasoning (CBR) to represent their responses (Wallace, 2003). This becomes beneficial to a performance-led system because CBR does not require the computational overhead that other reason-

based systems would demand (Breese and Heckerman, 1996). Although ALICEbots are sometimes compared to rule-based procedures, rules can contradict one another while cases are repetitive and never contradictory (Gilboa and Schmeidler, 2000).

ALICEbots are also able to expand their present knowledge bases through XML-based AIML (Artificial Intelligence Markup Language) (Wallace, 2003). This would imply that ALICEbots could be given an 'expert appearance' within a particular domain of knowledge. This expansion has already been witnessed in the areas of foreign language fluency and specific domain-related knowledge fields which can either be supervised by an interceding chatterbot master or unsupervised where knowledge is gathered en masse from trusted sources.

In the supervised approaches, there is a responsible party that filters and formats knowledge into a machine-readable form. This can be a labor intensive task and is the default method of ALICEbot's learning. In unsupervised approaches, massive amounts of knowledge can be gathered in a relatively short period of time, however, it is believed that users can sometimes be untrustworthy sources of knowledge (Wallace, 2003).

Within the realm of unsupervised learning is 'ART' which is an unsupervised pattern recognizer (Fu, 1996). Fu does concede that because of its unsupervised nature, ART can misclassify entire classes of instances. Another similar study in spoken dialog systems made use of a sub-verification routine (Smith, 1998). Whenever the system is unsure about the spoken request, i.e., the calculated value of understanding falls below a certain threshold, the system will ask clarifying follow-up questions regarding the unclear

statement in order to gain a clearer understanding. It was found that such use of a sub-dialog verifier dramatically increased the overall system accuracy.

In studies conducted on ALICEbots in particular, some interesting results have been obtained. One study focused on using an ALICEbot as a Social Theory tutor for students (Moore and Gibbs, 2002). It was discovered that students were more interested in using the system as a search engine to answer assignment questions rather than as the conversational tutor. This sentence-based information retrieval aspect had traditionally been confined to the arena of search engine design (Radiv, Fan et al., 2005). In another learning-style experiment, a modified version of an ALICEbot was used as a learning tool to teach Chinese students either English or German (Jia, 2002). This study focused more on user attitudes rather than on chatterbot efficiency. It was discovered that 62% of users chatted for 10 lines or less and that 8.5% of the time ALICEbot had no specific pattern to match the given input and had to rely on root-level generic responses. These conversational entities all have in common the difficulty of maintaining dialog for a sustainable period of time (Zacharski, 2004).

Throughout the examination of the field, several research gaps were noticed. The first such gap is that no ALICEbot harnesses a user-centric automated approach to knowledge acquisition. ALICEbots by default have relied solely on manual or semi-automatic methods for obtaining knowledge. The other notable find is that no one has examined the effectiveness of chatterbots in conversational or specific knowledge domains.

2.3 Proposed Approaches

Dialog systems can function in one of two ways; they can provide brief, concise or well-detailed answers to a particular query or they can engage the user in providing small-talk types of conversational responses.

This leads to our exploration of mass knowledge acquisition where our aim is two-fold. First, we explore using human subjects to train various dialog systems and study the impact of the acquired knowledge. Second, we study the effects of domain answers to those of the conversational dialog. To accomplish this, we pose the following questions:

- How effective is interactive training of a dialog system?
- How well can a chatterbot perform in a specific domain compared to general conversation?

To properly address the first question, we have a dual set of hypotheses.

- H1A: Mass knowledge acquisition will not make a noticeable improvement of responses for the general conversational chatterbot.
- H1B: Mass knowledge acquisition will improve the responses of the domain-specific chatterbot.

It is believed that because of the complexities involved and the limitless breadth of topics that can be covered in general conversation, we do not feel that we will find a noticeable difference. However, it is believed that for a constrained set of domain-specific knowledge, acquiring a set of additional knowledge will provide a noticeable improvement in chatterbot responses.

- H2: We expect that ALICEbots will be more effective in domain responses than in general dialog responses.

In H2, we further suspect that the responses from a constrained set of domain-specific knowledge will be more effective than those from general conversation.

To answer our questions, we constructed the AZ-ALICE dialog system. AZ-ALICE is built upon the freely available java-based ALICE Program D from www.ALICEbot.org. Our system can be broken into five component parts; the Chat User Interface, Chat Engine, AIML (Artificial Intelligence Markup Language) knowledge files, a Logging component and Evaluation module.

The Chat User Interface is an XML-based web page that allows users to authenticate themselves and chat with the system. The system stores the authenticated name as a state variable which allows for personalized communication (e.g., What do you mean by that Sally369?).

The Chat Engine functions as the brain of the system, pattern-matching user input against the intelligence stored in the AIML files and returning the corresponding response.

The AIML files contain the knowledge of the system which is structured in XML format as patterns and templates. When a user inputs a query to the system, the Chat Engine seeks to best match the input against all of the loaded patterns and returns the associated template. Each of the pattern/template pairs are also known as categories.

```
<category>
<pattern>WHAT IS *</pattern>
<template>Do you think I should know that?</template>
</category>
```

The Logging component records the entire conversation history for each user. Finally, the Evaluation module permits the user to evaluate and offer a corrected system response. These evaluations are later analyzed for performance-related data. Table 2.1 summarizes the differences between ALICE and the AZ-ALICE system.

	Chat Interface	Chat Engine	AIML	Logging	Evaluation
ALICE	Uses XML to chat with users	Uses off the shelf ALICE Program D	Uses the freely available Standard AIML set	Logs everything to a monolithic XML log file	None
AZ-ALICE	Custom perl skin	Same as Original ALICE	Uses Standard AIML or customized Telecommunications knowledge	Keeps XML logs on a per user basis	Custom perl script

Table 2.1. Differences between ALICE and AZ-ALICE

The Chat User Interface used in the AZ-ALICE system combines together the system dialog and evaluation module. Users are allowed to immediately rate and suggest new chatterbot responses following each line of inputted chat. Figure 2.1 shows a screenshot of the AZ-ALICE in evaluation mode.

[UA Home](#)
[Eller College](#)
[Dept of MIS](#)

[Knowledge Acquisition main menu](#)
[Logout](#)

NetID: kateama111 Welcome to dialogue evaluation. Please

(1) rate the original response

(2) edit the should-be response textbox when necessary to make it a more useful/meaningful response.

! Important: Please save the evaluation result with the save button at the end of this page when you are done.

Input #1

KB Response

Evaluation strongly disagree disagree somewhat disagree
 neither agree or disagree some what agree agree strongly agree

Should-be Response

Figure 2.1. Screenshot of AZ-ALICE evaluation

To illustrate how the AZ-ALICE system works; consider the following input/response interaction.

```
User: Tell me a funny joke.
AZ-ALICE: Did you hear the one about the Mountain Goats in the
           Andes? It was Ba a a a a d.
User: Tell me another one.
AZ-ALICE: I am the latest result in artificial intelligence which
           can reproduce the capabilities of the human brain with greater
           speed and accuracy.
```

- AZ-ALICE passes the user input “Tell me a funny joke” to the chat engine.
- The chat engine uses a memory-resident directed graph to best match the user input. In the case of the first user input, the best match is “TELL ME A JOKE”.
- The chat engine returns the corresponding response to the input, “Did you hear...”

The second user input/response interaction demonstrates the shallow behavior of the ALICE chat engine. In the second user input the context is still about funny jokes, however, ALICE treats this second input as a completely new topical interaction. This lack of context returns the best match to the input without consideration to the topic. The chat engine behavior does at times become problematic when attempting to maintain a conversation for any length of time.

```
User: ok well i think i might go now
AZ-ALICE: Oh. How about that. Since when? Perhaps. How do you
           plan to do this?
User: i'll logoff
AZ-ALICE: Do you think your plan will succeed?
User: yes its not that difficult
AZ-ALICE: That was not a yes or no question. I think you mean
           'it's' or 'it is' not 'its'. What is it?
```

However, the ALICE chatterbot is able to return adequate replies to brief user inquiries. The following interaction demonstrates this behavior.

User: I went on a big date last night.
 AZ-ALICE: Was it very expensive? Did you have a good time?
 User: It was very expensive. I had a great time.
 AZ-ALICE: How did you feel about that? What happened to it?
 User: I had the salmon.
 AZ-ALICE: Do you still have the salmon ?
 User: I ate it yesterday.
 AZ-ALICE: How did it taste?
 User: It tastes very good.
 AZ-ALICE: Like chicken?
 User: Not like chicken but chicken is good too.
 AZ-ALICE: Oh I get it.
 User: Do you like chicken?
 AZ-ALICE: Yes it is good especially baked.
 User: How can you eat chicken if you are a robot?
 AZ-ALICE: That is a good epistemological question.

In terms of telecommunications knowledge, ALICE exhibited some of the same shortcomings of keeping conversations going for any length of time.

User: Do you know what a firewall is?
 AZ-ALICE: A firewall is a specialized gateway designed to block or restrict certain types of data transfers on the Internet.
 User: Can you list different types of firewalls?
 AZ-ALICE: You should just download my software.
 User: What software?
 AZ-ALICE: Interesting question.
 User: How could I download your software?
 AZ-ALICE: A lot of practise.

2.4 Experimental Design

In our experiment, we created two chatterbots, BaseBot and TeleComm. BaseBot, the general conversational chatterbot, was our control chatterbot. It was an off-the-shelf ALICE ProgramD chatterbot loaded with the ‘Standard AIML’ rule set consisting of 23,735 knowledge categories that can be freely obtained from www.alicebot.org. Each of the knowledge categories consists of a pattern to match against the user input and a template response corresponding to the pattern. The other chatterbot, TeleComm, was

essentially identical to BaseBot except that TeleComm was further augmented with 298 telecommunication specific definitions thus raising TeleComm’s total rule set to 24,032¹.

Our experiment was further divided into two separate user studies utilizing both chatterbots. In User Study 1, subjects were encouraged to imprint new knowledge patterns onto their particular chatterbot to offset incorrect responses to their queries. In User Study 2, the new knowledge gathered from each chatterbot was then integrated into the preexisting knowledge stores of BaseBot-2 and TeleComm-2 respectively. Subjects in the second user study were given the same instructions as those of the first.

2.4.1 Participants

Participant subjects came from four sections of a freshman introductory course in Management of Information Systems. Each class section was given a particular chatterbot. Participation was on a voluntary basis, however, participants who successfully completed the requirements were given bonus point incentives. Table 2.2 shows the breakdown of participation among the experiments. In total, 376 participants were involved in our study. We believe that this is probably the largest systematic study involving a chatterbot.

Number of Study Participants		
Chatterbot	User Study 1	User Study 2
BaseBot	74	98
TeleComm	91	113

Table 2.2. Participant Breakdown

¹ One category pattern overlapped between ‘Standard AIML’ and Telecommunications definitions, decreasing the total by one.

Participants were asked to interact with the chatterbot for approximately one-half hour. At the end of the interaction period, users were presented with all of their user input and chatterbot responses. Participants were then given the opportunity to correct chatterbot responses and rate their satisfaction level with the response using a one-to-seven Likert scale (one – strongly dissatisfied to seven – strongly satisfied). Subjects were further asked to constrain their chatting to telecommunication topics; however, participants were not forced to do so. This gentle absence of restricting topics further allows us to evaluate Han’s claim that the inclusion of general conversational knowledge can help users to self-steer their conversation back to a domain-specific target (Han and Kim, 2001).

While we can agree that in this experiment where users were provided with an incentive of receiving a meager amount of classroom bonus points, transferring this technology ‘to the wild’ would derive many of the same benefits as mentioned with open-source software. The best comparison to an existing knowledge-based system would be Wikipedia, warts and all. Users derive no more benefit from adding and correcting the knowledge in Wikipedia, other than self-satisfaction and use of the tool for areas in which they may not as knowledgeable (Wagner, 2004). Applying this parallel to a real-world instance of our application, users would be creating and improving knowledge in much the same way as Wikipedia and for the same reasons.

2.4.2 Performance Metrics

The first metric that we aim to study is that of Correction Rate. Since users are permitted to correct chatterbot responses, Correction Rate is defined as a percentage of the number of new responses entered into the system divided by the total number of user inputs. This metric is used to evaluate each chatterbot to determine whether the Correction Rates are increasing, decreasing or remaining steady after mass knowledge acquisition.

The second metric is Response Satisfaction. Response Satisfaction is measured for each user input/response pair on a Likert scale of one to seven. This measure is then aggregated for each chatterbot and is simply the summation of all Response Satisfaction scores for the chatterbot divided by the total number of user inputs.

2.4.3 Experiment setup

Testing of H1A and H1B, mass knowledge acquisition effectiveness of the BaseBot and TeleComm chatterbot respectively, is carried out between user studies. Users from the first user study whom have provided new responses for BaseBot and TeleComm will have those new responses incorporated into the second user study chatterbots, BaseBot-2 and TeleComm-2. The change in Correction Rate and Response Satisfaction between both BaseBot-1 to BaseBot-2 and TeleComm-1 to TeleComm-2 is then measured.

In testing hypothesis H2 (domain responses will be more effective than dialog), the Response Satisfaction measures of the BaseBot and TeleComm chatterbots are broken

down into constituent dialog and domain knowledge responses and calculated against one another. Figure 2.2 graphically demonstrates the experimental relations.

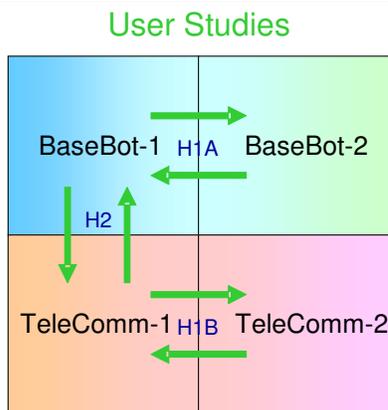


Figure 2.2. Experimental Design Testing

2.5 Experimental Results and Discussion

2.5.1 Users made corrections to nonsense responses

In our investigation of mass knowledge acquisition, users were instructed to make corrections to the knowledge sets of Study 1 that would then be incorporated into the knowledge sets used by Study 2. The following user/chatterbot interaction followed this pattern:

```
User in Study 1: What are you talking about?
AZ-ALICE: The topic is mood are you in.
User's new suggested response: My bad.
```

```
User in Study 2: What are you talking about?
AZ-ALICE: My bad.
```

From this acquisition interaction, users entered 1,707 new responses (17.5% of BaseBot-1 responses) for BaseBot. While the TeleComm-1 chatterbot received slightly

fewer new responses, 1,632 (16.0% of TeleComm-1 responses). These results are summarized in Table 2.3.

Stats on Overall Use	User Study 1		User Study 2	
	BaseBot-1	TeleComm-1	BaseBot-2	TeleComm-2
Total Number of Users	74	91	98	113
Lines of User Input	9,751	10,179	14,484	13,641
New Responses Entered	1,707	1,632	2,722	2,430
Correction Rate	17.5%	16.0%	18.8%	17.8%
Corrections per User	23	18	28	22
Number of New Responses used by the chatterbot			156	101
Percentage used			1.1%	0.7%
How many were recorrected			41	31
Percentage recorrected			26.3%	30.7%

Table 2.3. Chatterbot User Corrections

From the bottom of Table 2.3, approximately one-percent of the corrections made to chatterbot responses from Study 1 were used by the chatterbot in Study 2. Of these corrections, 26.3% of the BaseBot-2 New Responses and 30.7% of the TeleComm-2 New Responses were further corrected. This leads us to believe that with more subjects on the web, mass knowledge acquisition could be viable.

However, one problem we encountered was the addition of new suggested responses to already acceptable chatterbot expressions. To continue the previous interaction:

User in Study 1: What are you talking about?
 AZ-ALICE: The topic is mood are you in.
 User's new suggested response: My bad.

User in Study 2: What are you talking about?
 AZ-ALICE: My bad.
 User's suggested response: Nevermind.

In this example, the context of User 1's suggested response of "My bad" was not general enough to encompass the given input. This led to User 2 issuing a second correction of "Nevermind" to address the input. However, not all user interactions that

fell into this category were because of inadequate response suggestions. Some of the input/response pairs came as a result of users wanting to provide more entertaining answers such as the following.

User in Study 1: What is your favorite movie?
AZ-ALICE: My favorite movie is undefined. Have you seen it?
User's suggested response: I don't have a favorite movie.

User in Study 2: What is your favorite movie?
AZ-ALICE: I don't have a favorite movie.
User's suggested response: Star Trek.

In this exchange, the user in Study 2 offered a new chatterbot response although it was not really needed. This user behavior presents a problem. We had intended that Correction Rates be indicative of grammatical or communication-related problems with chatterbot responses where users would offer new responses for only those responses that needed it. But with participants instead elaborating on otherwise good responses, this may not be a limitation as much as expected student interaction with a system for a certain demographic subset and as such, this measure needs to be considered and interpreted cautiously.

To follow-up on these types of unnecessary corrections, we ask a further question of How accurate is a dialog system such as ALICE, and what types of errors/deficiencies occurred?

To answer the first part of this question we need to take a look at each of the system's total Correction Rate values irrespective of conversational dialog and Telecommunications knowledge, and assume that system accuracy can be modeled as one minus the Correction Rate. We acknowledge that measuring Accuracy as $(1 -$

Correction Rate) can be problematic in an absolute sense, however since our measure is consistently applied between chatterbots, the relative measure is interpretable and has meaning. Table 2.4 provides a summary of system accuracy.

System Name	Correction Rate	Accuracy
BaseBot-1	17.5%	82.5%
TeleComm-1	16.0%	84.0%

Table 2.4. System Accuracy as a measure of Correction Rates

AZ-ALICE's total accuracy figure hovers near 80% which might be satisfactory in some settings (e.g., conversation) and unsatisfactory in others (e.g., information retrieval). An example conversational interaction is shown below.

User: He did a good job teaching you.
 AZ-ALICE: Yes he did didn't he.

For the second part of our research question, *what types of errors/deficiencies occurred*, we take a qualitative approach by categorizing the most prominent types of response errors that were observed in the system logs. Table 2.5 outlines the observed chatterbot response errors in both BaseBot and TeleComm.

Overall	BaseBot	TeleComm
Nonsense responses	35.7%	34.5%
Spurious user corrections	28.6%	6.9%
Wordy and awkward responses	28.6%	13.8%
Application or Opinion type questions	7.1%	20.7%
Incomplete lexicon of terms	0.0%	24.1%

Table 2.5. Observed chatterbot response errors

For BaseBot, the most common error categories are; nonsense responses, spurious user corrections, and wordy and awkward responses.

2.5.1.1 Nonsense responses

For nonsense responses, this type of error generally occurred when the system reflected the user input back to the user and either selects improper pronouns or uses a sentence fragment in an improper context. The following interaction is a good example of this type of deficiency.

```
User: Did you buy your car yourself?  
AZ-ALICE: No I didn't buy your car yourself. What's buy your car  
yourself like?
```

In this interaction the system incorrectly used the pronoun of 'yourself' and ended up with a nonsense reply.

2.5.1.2 Spurious User Corrections

The second conversational dialog error type, spurious user corrections, occurred when users offered unnecessary and spurious corrections to the system responses – corrections which do not benefit future users of the system. The following snippet illustrates this category.

```
User: do you like metallica?  
AZ-ALICE: Not since they put pressure on Napster.  
User Corrected Response: YES they are the BEST band in the  
history of humankind.
```

This interaction shows that the original response makes sense in the context of user input, however, the user decided to insert a more entertaining response for future system interactions. It was interesting to note that this category would usually have a higher than average Response Satisfaction score.

2.5.1.3 Wordy and awkward responses

For wordy and awkward responses, the AZ-ALICE system glued together non-sequitur sentences and sentence fragments when confronted with complex sentences containing multiple subject/verb pairs. Here is an example.

```
User: Yeah everyone else is sorry when they do hear me sing.
AZ-ALICE: Yeah I know what you mean about that. That was not a
           yes or no question. That's a rather sweeping generalization.
```

The above interaction shows the system picking three separate matches keying in on different words from the user's input.

The telecommunications knowledge domain had its own class of errors and deficiencies; users would ask application or opinion oriented questions rather than definitional ones, the system held an incomplete lexicon of terms, or the system would provide nonsense responses.

2.5.1.4 Application or opinion type questions

The error category of application or opinion type questions houses all of the ways students would use telecommunications keywords outside of a definition type context.

```
User: Is internet telephony useful?
AZ-ALICE: The transmission of voice/data over long distances.
```

In this interaction, the system recognized the term 'telephony' and returned the predetermined AIML definition. Even though subjects were instructed to ask definition type questions, this category of error was nevertheless widespread.

2.5.1.5 Incomplete lexicon of terms

In the telecommunications error category of incomplete lexicon of terms, there was some discourse in which users would ask obvious telecommunications definitions for which the system possessed a response, however, the system would instead key in on a different part of the user input and ignore the telecommunication term.

User: How does an IP address work?
AZ-ALICE: How should I know it does?

In this example, the system matched ‘How does *’ rather than ‘* IP *’. These types of mismatch problems, while correctable by AIML maintenance, are common to a conversational system.

2.5.1.6 Nonsense responses

The error category of nonsense responses is the only category that spans both the conversational dialog and Telecommunications knowledge domains. Nonsense responses are a problem with the ALICE ProgramD chat engine, particularly when the elicitation of particular domain-related answers are desired. Again, this is the trade-off between conversational entertainment and terse knowledge delivery. The other interesting item to note was that both of the other domain-related system problems stemmed from an incomplete set of domain answers.

2.5.2 Knowledge Acquisition improved general conversation responses

In looking at the Response Satisfaction ratings for the general conversational chatterbot BaseBot, it would appear that mass knowledge acquisition improved. This

improvement, 4.31 in the first user study to 4.33 in the second, was found to be statistically significant with a p-value < 0.001 . This significance value is the result of the large number of interactions (n) with the system. Table 2.6 summarizes BaseBot Response Satisfaction and Correction Rates.

BaseBot stats on Satisfaction				
	n	Avg	Std Dev.	Corr. Rates
Study 1	9,751	4.31	1.50	17.5%
Study 2	14,484	4.33	1.76	18.8%

Table 2.6. BaseBot Satisfaction and Correction Rates

The increase in Response Satisfaction between user studies is likely to be attributed to the refinement of conversational chatterbot responses for Study 2 users. Returning to our hypothesis H1A, we find that mass knowledge acquisition does in fact make a noticeable improvement of user satisfaction for the general conversational chatterbot.

2.5.3 Knowledge Acquisition improved telecommunication responses

The TeleComm chatterbot is composed of both conversational as well as telecommunication knowledge. In looking at the Response Satisfaction ratings for the domain-specific knowledge, it would appear that mass knowledge acquisition improved its responses as well. This improvement from a mean of 4.64 in Study 1 to 4.95 in Study 2 was found to be statistically significant with a p-value < 0.001 . Correction Rates decreased between studies (p-value < 0.001), which was in line with expectations. Table 2.7 summarizes TeleComm response satisfaction and correction rates.

TeleComm stats on Satisfaction				
	n	Avg	Std Dev.	Corr. Rates
Study 1	218	4.64	1.70	15.1%
Study 2	346	4.95	1.79	13.6%

Table 2.7. TeleComm Satisfaction and Correction Rates

This increase in response satisfaction and decrease in correction rates between user studies can be attributed to the refinement of telecommunications knowledge similar to that observed in the dialog system. It is interesting to note that the magnitude of mean movement between the two studies is much greater than that of the dialog system. This could be due to the limited amount of initial telecommunications knowledge used in Study 1 as compared to the much larger array of conversational knowledge, which made more of an impact on the telecommunications mass knowledge acquisition.

From hypothesis H1B, we had expected that mass knowledge acquisition would improve the domain-specific chatterbot responses. From the analysis above, we find that mass knowledge acquisition does improve domain-specific chatterbot responses.

2.5.4 Domain responses are more effective than Dialog

To answer our hypothesis H2, that domain-specific chatterbot responses will rate better than general conversation, we examine the chat output from both BaseBot and TeleComm in User Study 1 by breaking the chat output of each chatterbot into either conversational or domain-specific chat. Table 2.8 illustrates such a breakdown of response satisfaction and correction rates between the chatterbots.

	User Study 1	
	Conversational Dialog	TeleComm Domain
BaseBot		
Lines of User Input	9,670	81
Response Satisfaction Avg	4.30	4.43
Response Satisfaction Std Dev.	1.50	1.66
Correction Rate	17.4%	28.4%
TeleComm		
Lines of User Input	9,961	218
Response Satisfaction Avg	4.16	4.64
Response Satisfaction Std Dev.	1.50	1.70
Correction Rate	16.1%	15.1%

Table 2.8. Comparing dialog to domain for the chatterbots

Looking at the BaseBot results, the mean response satisfaction value is higher when the BaseBot is talking about Telecommunications than general conversation (4.43 versus 4.30, p -value < 0.001). Even though the BaseBot was not loaded with a specific Telecommunications knowledge set, BaseBot did have some familiarity with some of the telecommunications terms. From the data, users rated this subset of dialog knowledge better than the rest of the set. The following input/response interaction illustrates the argument.

User: What is an IP address?
 AZ-ALICE: The number of a specific machine like your 127.0.0.1.

The same conclusion can be drawn about the TeleComm chatterbot. Its mean response satisfaction value is higher for telecommunications than general conversation (4.64 versus 4.16, p -value < 0.001). This result meets our expectation that the domain knowledge is better than the conversational knowledge. Therefore, we found that domain responses are clearly more effective than dialog.

2.6 Conclusions

From our study we can conclude that the use of a chatterbot as a knowledge acquisition tool appears to be a stable instrument in gathering both conversation and domain-related knowledge. We believe that with the decrease in correction rates observed between studies, that after several rounds of such corrections the knowledge base will be of sufficient quality to answer domain-related questions. Extending this research in such a way demonstrates the viability of having users train a conversational system. Although a minority of subjects did intentionally leave misleading or false responses, we feel that the iterative process will eventually eliminate such responses. Furthermore, we found that mass knowledge acquisition restricted to a particular knowledge domain had higher Response Satisfaction levels than the corresponding conversational-style responses regardless of the chatterbot involved.

From our examination of natural language dialog systems and ALICEbot in particular, we believe that ALICEbots show a promising future in domain-restricted areas. Although we studied only one particular area of domain expertise in telecommunications, it would be interesting to further pursue other areas of domain interest and test the flexibility of the ALICEbot system in those areas as well.

CHAPTER 3: ACQUISITION AND DELIVERY ON A TERRORISM DOMAIN

This chapter investigates the potential use of dialog-based ALICEbots in disseminating terrorism information to the general public. In particular, we study the acceptance and response satisfaction of ALICEbot responses in both the general conversation and terrorism domains. From our analysis of three different knowledge sets: general conversation, terrorism and combined, we found that users were more favorable to systems that exhibited conversational flow. We also found that the system that incorporated both conversation and terrorism knowledge performed better than systems with only conversation or terrorism knowledge alone. Lastly, we were interested in what types of questions were most prevalently used and discovered that questions beginning with ‘wh*’ words were the most popular method to start an interrogative sentence. However, ‘wh*’ sentence starters surprisingly proved to be a very narrow majority.

3.1 Background

Since the events of September 11th, terrorism education and awareness has become a national priority. The Department of Homeland Security (DHS) has written many training guides instructing first responders and citizens how to react in a terrorist event. DHS has also launched several initiatives in this area, such as Community Emergency Response Teams (CERT) and Ready.gov that contain online preparation resources. Independent organizations such as the American Red Cross have also prepared terrorism education materials which instruct citizens in basic emergency procedures. Preparing

citizens to act in a terrorism incident is important because citizens are often the first ones to respond to a scene and can be highly effective during an emergency (Furedi, 2004; Alexander and Klein, 2006).

As a part of mobilizing citizen and post-terrorism response initiatives, two post Sept. 11th reports, “Making the Nation Safer” and “The 9-11 Commission Report,” present the importance of a mobile Command, Control, and Communications (C3) system as a vehicle of information delivery. The C3 system could be quickly deployed to the scene of a terrorist attack and would be responsible for providing both emergency communications channel capacity for first responders and acting as a focal point for information collection, management and decision-making tasks. It would also have the ability to provide communications to the public by providing citizens with timely information that could aid in rescue. One of the secondary roles of a C3 system is to get information to the public and fill the void before rumors, hoaxes and misinformation become mainstream (Durodié and Wessely, 2002).

One system with promise to meet many of the C3 system guidelines is ALICEbot. ALICEbots are a class of Question Answer programs developed in 1995 by Richard Wallace (Wallace, 2004). These Question Answer systems can be programmed with terrorism-specific knowledge and quickly deployed in a terrorism-affected area. ALICEbots have the advantage of holding an immense library of knowledge, are able to communicate in natural English and are robust and scalable enough to address new areas of knowledge. This class of system shows promise in the field of terrorism education by allowing citizens to seek specific terrorism knowledge as well as disseminate real-time

critical knowledge to those in need, such as family members, in a more interactive way other than a unidirectional television broadcast.

One of the major hurdles to using ALICEbots and Question Answer systems in general, has been returning responses that are appropriate and acceptable to the context of the question. With many different ways to communicate information and structure sentences, computer systems have had major difficulty in the basic understanding of language. The unique aspect of ALICEbot has been its ability to mimic conversational English without regard to understanding. This primitive approach to contextual accuracy has garnered ALICE the Loebner Prize for most human computer in 2000, 2001 and 2004. This simple conversational technique has made the goals of attaining a working C3 system appear to be within grasp.

This rest of this chapter is organized in the following sections. Section 3.2 is a literature review and talks about how ALICEbots, Question Answer and C3 systems are all related to each other. Section 3.3 asks a series of research questions about the retrieval potential of ALICEbots and introduces the reader to the Terrorism Activity Resource Application (TARA) system which was created as a design model for C3 research. Section 3.4 is the experimental design segment and lays a framework for evaluation. Section 3.5 is the experimental results and discussion of what the results mean. Finally, Section 3.6 wraps up the study with conclusions and possible future avenues of research.

3.2 Related Works

Question Answer systems are a branch of Natural Language Dialog Systems that are more information retrieval oriented with a close parallel to search engine technology. These types of systems are presented with a user-based input and must search their knowledge bases much like a chatterbot in an attempt to return an answer. While the simpler chatterbot technology of directly matching an input to a response is considered to be a toy domain, the complexities involved in the querying and retrieval aspects of Question Answer systems lend this branch its research worthiness.

When discussing Question Answer (QA) systems and how ALICEbots fit into the model, there are several frameworks available. The first of which is the Natural Language Dialog Systems (NLDS) framework (McKevitt, Partridge et al., 1999). In the NLDS framework, QA systems can be broken into three distinct camps: Semantic theory, Structure theory and Intention theory.

Semantic theory deals with finding and inferring relationships from system input. An example of such a system is Doug Lenat's CYC project that attempts semantic understanding through complex word relations (Lenat, Miller et al., 1995). Within Semantic theory, meaning is derived from the words used. This approach is computationally more difficult than other approaches, as it requires a large corpus of word meanings and word sense disambiguation in order to be effective.

Structure theory is instead more concerned with syntactic sentence markers or cue words and analyzes their communicative function to the system input (Andernach, 1996). Syntactic systems are generally more shallow than semantic ones and require less

processing overhead (Hammerton, Osbourne et al., 2002). In QA systems, the syntactic relations of objects generally deal with interrogatives such as who, what and where. Focusing in on the use of interrogatives can help the syntactic system narrow the scope of possible response choices and provide a more accurate reply. Many of the current systems handle simple fact-driven questioning but fail at tasks requiring any level of reasoning (Moldovan, Pasca et al., 2003). An example of such a syntactic system is ALICE (Artificial Intelligence Linguistic Chat Entity). ALICE's syntactic nature will be described in later sections.

In Intention theory, system input is used to communicate plans and beliefs. This is one of the most difficult areas of research. Through this theory, systems possess reasoning abilities and are able to work through problems of a very constrained nature. An example of such a system is TRAINS 95, where the system's purpose was to route locomotives to meet economic demands and to minimize disruptions due to track outages and other simulated disasters (Ferguson, Allen et al., 1996). This system is constrained by the rules given to it, and is unable to abstract itself to meet the challenges of a changing problem that violates the existing rules.

The other framework concerning QA systems is the class system (Moldovan, Pasca et al., 2003). In this framework, systems are classified according to how they function.

- Class 1 – QA systems that can process fact-based questioning
- Class 2 – QA systems that can use simple reasoning
- Class 3 – QA systems that can fuse together answers from different sources
- Class 4 – QA systems that remember and use previous dialog to form answers

- Class 5 – QA systems that can perform analogical reasoning

To give the reader some background in the area of QA systems, a special Question Answer track was started in 1999 from the Text REtrieval Conference (TREC-8), in which systems were given a standardized test of information retrieval tasks (Voorhees, 2001). Systems are given various class questions, from class 1 questions like, “Who is Bin Laden,” to class 5 questions like, “Why do terrorists hate the West?” (Moldovan, Pasca et al., 2003).

3.2.1 ALICE

ALICE is a derivative of structure theory (McKevitt, Partridge et al., 1999) and class 1 QA systems (Moldovan, Pasca et al., 2003). This fact-driven syntactic parser uses Case-Based Reasoning (CBR) to determine its responses. The knowledge used by ALICE is contained in XML-based Artificial Intelligence Markup Language (AIML) files that can be easily extended to meet particular knowledge demands. The process of adding new knowledge to the system allows ALICE to convey an ‘expert appearance’ in narrow domains of knowledge (Wallace, 2004). Because of ALICE’s dependence on syntactic parsing (Russell, 2002), this lack of cognitive theory will result in ALICE missing certain types of user interactions. This has become the case in several studies done using ALICE chatterbots where ALICE typically scores an 80% accuracy rate in general conversation and slightly higher when coupled with specific knowledge domains.

What has become interesting is the use of ALICE as a tutor in several research studies. In a University of Huddersfield experiment, ALICE was carefully given the

personalities of four famous social psychologists. The intent of the project was to use ALICE as an interactive tutoring tool where students could pose questions and gain further insight into the thinking style of these four individuals (Moore and Gibbs, 2002). The experiment unfortunately failed when students used ALICE as a search engine to obtain assignment answers and not as a conversational tutor. Another research study conducted in China used ALICE as a conversational partner to teach Chinese students either English or German (Jia, 2002). This study had surprising results that focused attention directly on ALICE's 80% accuracy rating. Participating students developed negative attitudes toward the system and generally chatted for no more than 10 turns before leaving. This study also reviewed a log of chat dialog and discovered that negative comments about the system outweighed the positive ones. It is thought that these results are in part due to the limited amount of conversational knowledge used by the system, which was roughly half of the AIML knowledge available at the time.

In a study conducted at the University of Arizona, several ALICE chatterbots were examined to measure the effects of using subjects as unfiltered tutors to build new chatterbot knowledge patterns (Schumaker, Liu et al., 2006). Along with the main thrust of the research, measures for response appropriateness and satisfaction levels for individual responses were performed. This study incorporated three similar chatterbots that differed only their knowledge bases. One system used a limited conversational knowledge base; the Standard AIML knowledge set. The second one used a limited conversational knowledge set from Standard AIML, coupled with several hundred telecommunications definitions. The third chatterbot was a combination of the other two.

From this experiment, it was found that subjects did not correct the combination chatterbot as often as the stand-alone systems. However, the conversation-only chatterbot was found to exhibit a significantly higher user satisfaction level. It was found that this result was due to the rejection of the telecommunications knowledge set because it contained an inadequate amount of knowledge base entries.

3.2.2 Emergency Response Systems and the C3 Model

There are several studies on public emergency response and communications systems worth looking at. The first study, called Citizen Awareness System for Crisis Mitigation (CAMAS), is an event extraction tool that utilizes ‘humans as sensors’ (Mehrotra, Wickramasuriya et al., 2004). This tool is deployed at a disaster site and collects first-hand information from victims as they exit the scene. This information could be about potential problems witnessed by the victims such as downed power lines, ruptured gas lines, etc., or could be information concerning where a group of victims may be located. This information is concentrated and analyzed by CAMAS and appropriate rescue personnel are then notified. The disadvantage of CAMAS is that it does not contain a mechanism for bidirectional information flow. CAMAS is unable to alert victims of impending danger, instead requiring first responders to relay such information.

The second study, called Integrated Community Care (INCA), is more tailored to the healthcare field (Beer, Hill et al., 2003). INCA is a network of health-monitoring agents that can oversee the well-being of a community by coordinating emergency services when a problem is detected. The disadvantages of INCA are similar to CAMAS. There

is no bidirectional flow of information from the emergency personnel to the affected victims. The unidirectional design of both of these systems preclude them from attaining the goals set forth for C3 systems.

The C3 Model has several key points that we will address.

- “In a crisis, channels to provide information to the public will be clearly needed.” (National Research Council, 2002)
- A C3 system must be tailored to help people in specific areas with specific needs.
- It must be effective in providing information to a targeted population with substantial infrastructure damage.
- It should be able to communicate the status of affected individuals to people outside of the disaster area.

The first point, channels to provide information to the public, may be one of the more obvious ones. Channels are needed to provide the public with information to mitigate further damage following a terrorist event. This information flow helps to calm the public by fixing their concentration on immediate concerns and empowers individuals to commit themselves in useful and needed ways. It also establishes a medium of trust that officials are in charge of the scene and eliminating some of the information vacuum described earlier (Durodié and Wessely, 2002).

The second point, a C3 system must be tailored to help people in specific areas with specific needs, directly addresses the need for bidirectional information flow. Such a system would be able to direct targeted groups to safety or provide them with information to satisfy a need. Such a system could also be of benefit to individual rescuers by helping them to locate and free trapped victims.

The third point, provide information to a targeted population with substantial infrastructure damage, illustrates the need for a C3 system to provide its own infrastructure, independent of whatever infrastructure may be remaining. This point could further be refined into issues of portability and durability of the system. In later sections we will discuss aspects of the ALICEbot system which could be placed on mobile communications devices and used in emergency shelters.

The last point, the system should be able to communicate the status of affected individuals to people outside of the disaster area, brings up the concept of “I’m Alive” boards. In the September 11th aftermath, displaced victims and their families fell out of touch and were unable to communicate with one another. Throughout Manhattan, “I’m Alive” boards sprang up where victims could tack their name and current shelter to the board. These victims relied on television news to communicate this information to the victims families. It turned out to be a cumbersome and rarely effective method of communication. However, with so many victims and “I’m Alive” boards, information overload quickly became a problem.

Because of the recent timing of “The 9/11 Commission Report” (National Commission on Terrorist Attacks upon the United States, 2004) and “Making the Nation Safer” (National Research Council, 2002), research has yet to be published regarding a complete instantiation of a C3 system. Several of the existing emergency response systems, such as CAMAS and INCA, only meet some of the criteria set forth. As of this writing, no system meets all of the C3 objectives.

3.2.3 ALICEbots and C3

ALICEbots can provide a unique answer to the problems of C3 systems. ALICEbots have the ability to disseminate information to the public through conversational dialog and are able to carry on multiple independent conversations at the same time. They can be quickly augmented with specific knowledge to help targeted groups and are portable in nature for infrastructure-damaged areas. The bidirectional flow inherent in ALICEbots makes the implementation of “I’m Alive” boards a simple programming exercise.

The knowledge used in the ALICEbot can be acquired in one of two ways; manual effort or automated means. The first method of manual effort, allows information specific to a disaster to be incorporated in the system. This is usually custom-tailored information such as safe escape routes and victim location. The second method, of automated means, is for general types of terrorism questions. These static answers can be automatically scooped up from reputable sources such as definitional websites or news organizations and incorporated into the knowledge base. The optimum solution would be a mixture of both where the automated information gathering is performed periodically to ensure timely and relevant information.

There are several reputable terrorism definition websites of interest to automated knowledge gathering. The first of which is www.terrorismanswers.com which is authored by the Council on Foreign Relations. This website provides a wealth of information on terrorist groups, individuals, states and events. The other website is www.11-sept.org which also contains an extensive glossary of terms on groups and

weapon descriptions. Both of these websites appear to be non-biased in nature and high-quality resources for ALICEbot to use.

3.2.4 Literature Review Summary

In reviewing the existing literature, the key facts are as follows:

- A focus on interrogatives is an important aspect of QA systems (Hammerton, Osbourne et al., 2002).
- C3 systems need to be able to distribute timely and relevant material.
- ALICEbots can potentially fill the role of a C3 system.

Since most users appear to use ALICEbots as specialized search engines (Moore and Gibbs, 2002), it would make sense to approach ALICEbot input in the same terms. The main benefit to using ALICE over a search engine is the ALICEbot's ability to return information within a conversational context, rather than displaying a list of probable sites and asking that users find the information on their own. Following up on the search engine context, it is noted that search terms are predominately definitional in nature (Voorhees, 2001). This would imply that the best method of acquiring knowledge for the ALICEbot would be to obtain definitional responses that are keyed on interrogative input.

The ability to provide information during a crisis is paramount. This flow of information serves to soothe public fears and calm citizens by providing direct and useful information in an appropriate manner.

Finally, as discussed earlier, ALICEbots already meet many of the challenges required of a C3 system. ALICEbots were built first and foremost for conversation and they can leverage that ability in specialized knowledge domains.

3.3 Proposed Approaches

Because of the versatility of the ALICEbot system, we have identified several research questions we feel need to be investigated. Based on our prior study on the usefulness and performance of a telecommunications Question Answering system (Schumaker, Liu et al., 2006), we ask:

- What is the perceived usefulness of a terrorism Question Answer system?

H1: It is expected that users will prefer the system with “Both” conversational and domain knowledge to acquire terrorism-related knowledge.

How will a Question Answer system perform with:

- Pure conversational knowledge
- Pure domain knowledge
- Both conversational and domain knowledge

H2: The Question Answer system with “Both” conversational and domain knowledge will perform better than either conversation or domain knowledge alone.

From our prior research, we believe that a sufficiently large terrorism knowledge base coupled with general conversational knowledge will yield higher response satisfaction scores than either alone. This is based on our belief that both elements will work together by helping users to reform their queries or pursue tangential areas of knowledge. Prior research with a more limited knowledge-base proved that the “Both” chatterbot could not accomplish this task, however, we believe that a larger terrorism knowledge base will improve cross-cooperation between the two knowledge-bases.

For our third research question we would like to investigate the particulars of subject sentence interaction in order to tweak future knowledge-base entries on those rules most likely to be executed.

- What are the most frequent types of questioning found in Question Answering systems?

H3: It is expected that interrogative sentences beginning with “Wh*” will compose a large segment of user input.

From the preceding question we will have a better idea of what question types to focus future data gathering activities on and as a consequence provide for better response-oriented accuracy gains.

To answer the research questions posed, we created the Terrorism Activity Resource Application (TARA) system, which is based on a modified version of ALICE Program D that can be freely obtained from www.alicebot.org. Both TARA and ALICE share many of the same components, however, for the sake of clarity, Table 3.1 shows the differences between the two.

	Chat Interface	Chat Engine	AIML	Logging	Evaluation
Original ALICE	Uses XML to chat with users through a crude Jetty interface	Uses off the shelf ALICE Program D	Typical setup uses Standard AIML	Logs everything to a monolithic XML Log file	Does not provide a user-based evaluation component
TARA	Uses a customized perl skin to chat and for evaluation purposes	Same as Original ALICE	Customized Terrorism AIML for domain knowledge and Standard and Wallace AIML for conversation	Keeps XML logs on a per user basis	Customized perl script that allows users to evaluate and suggest new patterns

Table 3.1. Differences between ALICE Program D and TARA

From Table 3.1 the reader will notice that ALICE and TARA have five major components. The Chat Interface was one of the more modified components in TARA. Both ALICE and TARA work by taking the user input, tagging it with XML-style tags and passing it to the chat engine through a Jetty web server, as illustrated in Figure 3.1.

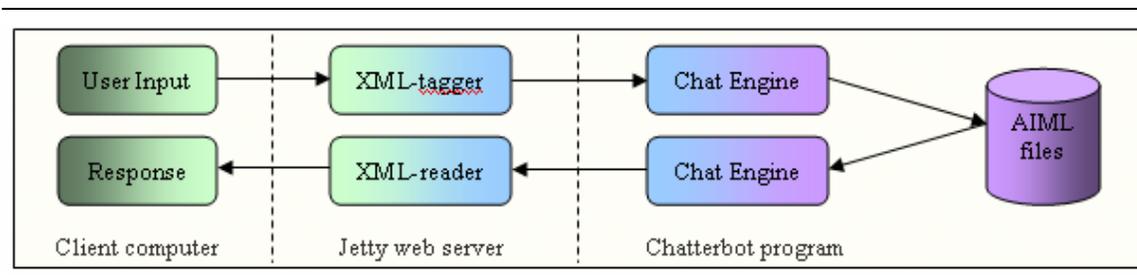


Figure 3.1. Diagram of User Input to Chatterbot Response

The chat engine is the core element of the system. When the system boots up, the chat engine loads all of the AIML rules or categories into a memory-resident directed graph. As user input is passed to the chat engine from the XML-tagger, the chat engine traverses this graph to find the most suitable match called a pattern. After the match has been made, the corresponding response or template is fed back as XML to Jetty. Jetty then unwraps the XML and posts the chatterbot response to the chat interface.

In the TARA implementation, we modified the chat interface by overlaying a perl skin on top of the client computer. This perl skin allowed for input and response with the Jetty web server as described above, however, it also allowed us to introduce an evaluation component which we will describe later. In both TARA and ALICE Program D, the chat engine and Jetty web server were identical.

The AIML files can be thought of as the brains of the system. The chat engine only organizes and retrieves information stored in AIML, while the AIML itself contains the input and response categories. The following is an example category:

```
<category>
<pattern>WHAT IS POTASSIUM IODIDE</pattern>
<template>FDA-approved nonprescription drug for use as a blocking
agent to prevent the thyroid gland from absorbing radioactive
iodine.
</template>
</category>
```

In order for the system to return the template as a response, the user will have to exactly ask “What is potassium iodide”. Although the system is not case-sensitive in matching patterns, it is sensitive to wildcard matching. The above example does not use any ALICE-specific wildcards, i.e., ‘*’ or ‘.’. So adding a question mark ‘What is potassium iodide?’, would not retrieve the same template. This shortcoming becomes more evident when using automated means of gathering terrorism knowledge. The simplest fix is to create multiple patterns that add wildcards both in front of and behind the key term. For example:

```
<category>
<pattern>* PENTAGON *</pattern>
<template>Headquarters of the U.S. Department of Defense. The
five-sided building, built in 1943, was one of the targets of
September 11. American Airlines Flight 77 was flown into the
Pentagon, killing 189 people in all, including 125 people
inside the building, 64 passengers, and 5 terrorist
highjackers.
</template>
</category>
```

This category will return the template if the pattern contains the word ‘Pentagon’ in the middle of the text. This is a more palatable solution, however, this may not be only

response desired when talking about the Pentagon. To give the reader an idea of how the ALICE wildcard system works, consider the following four patterns:

- Bin Laden
- * Bin Laden
- Bin Laden *
- * Bin Laden *

In the ALICEbot scheme of wildcard handling, each of these patterns are distinct which prevents two or more patterns from matching the user's input. In the first case, 'Bin Laden' will be activated *only if* the user's input is precisely, 'Bin Laden' with no other characters before or after the input. In the second case, '* Bin Laden' will be activated *only if* 'Bin Laden' is at the end of a sentence with preceding characters. This leads us to the use of qualifying definitions with the use of interrogatives, such as 'What is the Pentagon' or 'Where is the Pentagon', and generating different templates for those types of questions. The following is a sample interaction to show how qualifying the templates can lead to difficulty.

User: Who is our president?

System: George W. Bush

User: What is our president respond to terrorist attack?

System: Any act or series of acts by an enemy causing substantial damage or injury to property or persons. In any manner by sabotage or by the use of bombs shellfire or atomic radiological chemical biological means or others or processes.

In this example, the system recognized the term 'terrorist attack', but failed to respond to it in the expected context. It is for instances like these that a larger and more specific corpora of AIML knowledge may be able to address.

The AIML files between TARA and ALICE are slightly different depending upon the chatterbot knowledge desired. For a general conversational chatterbot it is recommended that the Standard and Wallace AIML knowledge sets be used. These knowledge sets helped ALICE to win the Loebner contests for most human-like chatterbot. This pattern of success against deeper reasoning systems is justification enough to use these free knowledge sets. For terrorism knowledge, we put together our own AIML sets which we will describe in the Experimental Design section.

The logging component has also undergone several changes from the ALICE implementation. In ALICE, logging was performed within the chat engine before passing the response back to Jetty. The logs used by ALICE were monolithic XML files that captured every interaction with the program. In our design of TARA, we incorporated logging into the perl skin of the chat interface. This allowed us the freedom to make programming changes without modifying the chat engine and risking code integrity problems. We logged each participant's conversation into separate XML files which helped limit data overload challenges in the analysis phase.

The evaluation component was new in the TARA design. In TARA, we provided a method of integrated evaluation where the user is given the opportunity to rate their satisfaction level after each chatterbot response is given.

3.4 Experimental Design

3.4.1 AIML Knowledge

For our experiment we chose to test three chatterbots, all with different types of AIML knowledge. Table 3.2 breaks down the AIML knowledge and shows the number of categories for each chatterbot.

Chatterbot	Number of categories		
	Conversational AIML	Terrorism AIML	Total
Dialog	41,873 †		41,873
Domain		10,491	10,491
Both	41,873	10,491	52,354 ‡

Table 3.2. AIML distribution for the three chatterbots

- † - Some of the Conversational AIML categories did include some terrorism terminology.
- ‡ - Ten categories between Conversational and Terrorism AIML overlapped, decreasing the total number of categories by ten.

The 41,873 categories of conversational AIML consisted of the Standard and Wallace AIML sets. For terrorism knowledge, we created AIML files through a mixture of automated and manual entry processes. Automated entry collected the most categories, 10,476, while 318 categories were obtained from manual methods. This represents an overlap of 303 categories, which occurs when two categories have the exact same pattern. When there is overlap, the chat engine ignores the duplicate categories thus decreasing the total number of categories.

From the 10,476 automatically gathered categories, 2,619 are unique patterns. This factor of four difference is because of the way the ALICE chat engine handles wildcards, as described earlier. These 2,619 patterns came from several terrorism-related websites

that were identified as reputable places to gather terrorism definitions by a terrorism expert.

From the 2,619 unique patterns collected:

- 1,322 came from www.terrorismsanswers.com
- 174 came from www.11-sept.org
- 1,212 came from www.wmd-nm.org

An overlap of 89 categories occurred between these three sites.

For manual terrorism knowledge, all 318 categories were gathered from www.highvolumemedia.com/thebullhorn/GlossaryA-Z.htm. These categories are represented by the following terrorism groupings:

- Bin Laden – 8 categories
- Countries – 29 categories
- Definitions – 131 categories
- People – 4 categories
- Terrorist Groups – 122 categories
- WMD – 24 categories

Because these terms were manually gathered, appropriate wildcards could be added.

This avoided the 4x wildcard handling experienced by the automated method. An example category is listed below.

```
<category>
<pattern>What is Zyklon B</pattern>
<template>A form of hydrogen cyanide. Symptoms include increased
    respiratory rate, restlessness, headache, and giddiness
```

```

followed later by convulsions, vomiting, respiratory failure
and unconsciousness. Used in the Nazi gas chambers in WWII.
</template>
</category>

```

3.4.2 Study Participants

Participants were recruited from several undergraduate and graduate Management of Information Systems classes. Each participant was assigned to one of the three chatterbots through a pseudo-random algorithmic process based on their University login ID's. Participants were encouraged to interact with the system for approximately one-half hour before completing a final survey. Students were asked to discuss terrorism-related topics, but the system did not force them to do so. Completion of the study was further encouraged through the prospect of randomly awarded gift certificates.

An integrated evaluation method was used in our study. Participants would chat a line, and immediately evaluate their satisfaction level of the chatterbot response. Figure 3.2 shows a sample screenshot of the evaluation process.

Please evaluate chatterbot response and click next.

You said: **Who is Osama Bin Laden?**
Chatterbot response: **He is the world's most wanted man.**

Do you feel that the chatterbot response is appropriate given your input? Yes No

If no, please explain:

How would you rate your satisfaction level of the chatterbot response in the context of your input?

Very Dissatisfied Somewhat Dissatisfied Mildly Dissatisfied Neutral Mildly Satisfied Somewhat Satisfied Strongly Satisfied

Figure 3.2. Screenshot of TARA's evaluation process

The final survey was composed of seven questions, four open-ended and three dichotomous. The four open-ended questions were a qualitative look at user impression of system strengths and weaknesses.

- What do you believe are the strengths of the system?
- What do you believe are the weaknesses of the system?
- What do you believe the system excels at?
- What do you believe the system needs help on?

The three dichotomous questions quantified user reaction concerning their comfort level with the system, whether or not they would use such a system to find terrorism information, and whether or not they would recommend the system to friend. Each of these three questions also had space for open-ended comment where users could optionally justify their answers.

- Do you feel comfortable using this system?
- Would you use such a system to find terrorism information?
- Would you recommend this system to a friend who was interested in terrorism knowledge?

A pilot study was also conducted to test the system and evaluate whether the responses received were what we had intended to capture.

3.4.3 Metrics

To answer the research questions posed we set up three chatterbots that differed only on the knowledge based used: general conversation, terrorism knowledge and a combined

system. The following independent variable metrics were designed: User Impression, Response Rating and Response Appropriateness. User impression is actually three subtle measures that tie directly to the three dichotomous questions asked in the final survey. These measures of user comfort, usability and recommendation potential are measured as an aggregate percentage for each chatterbot. The open-ended comments accompanying these questions are used to qualitatively determine causes behind user choices.

Response rating is introduced in the evaluation component and is a measure of the user's satisfaction level of the chatterbot response. This measure uses a Likert scale and is an aggregate computation of all response ratings divided by the number of user inputs.

Response appropriateness also comes from the evaluation component. It is a dichotomous measure that asks participants whether or not the response is appropriate given the user input. This is different from response rating, because an interaction may produce an unexpected response leading to a low response rating yet be appropriate given the context of the query. This measure is aimed more towards determining whether a participant will accept the system response or not.

3.5 Experimental Results and Discussion

3.5.1 Users preferred Conversational to Definitional Responses

In our first hypothesis, we postulated that users will prefer the chatterbot with Both dialog and domain knowledge. Using the qualitative questions in the final survey, the results of Table 3.3 are generated.

User's 'Final Survey' Analysis	Chatterbot		
	Dialog	Domain	Both
Number of Users completing the survey	27	25	29
User Comfort	22.2%	44.0%	31.0%
Usability	11.1%	28.0%	31.0%
Recommendation Potential	22.2%	28.0%	24.1%

Table 3.3. Analyzing User Behavior towards the system

In looking at the percentages for the 'Both' chatterbot, it was preferred the most in the category of usability (31.0%). For the other two categories the Domain chatterbot was preferred, 44.0% and 28.0% respectively. This counters our intuition of the hypothesis and was further investigated.

In looking at the first category of user comfort, we analyzed the open-ended comment of the Domain and 'Both' chatterbots. It soon became clear that some users had selected other criteria to judge the system. Table 3.4 shows the Domain and 'Both' chatterbot comments broken down by common groupings.

User comment groupings for 'User comfort'				
	Domain chatterbot		'Both' chatterbot	
	Comfort:Yes	Comfort:No	Comfort:Yes	Comfort:No
Ease of use	3	0	0	0
Responses	7	14	8	20
Other	1	0	1	0

Table 3.4. User comments about comfort using the system

To demonstrate how we arrived at our "Ease of Use" categorization numbers in Table 3.4, we list the open-ended comments received: "easy to use", "it is easily deployed via the web...", and again "easy to use". The "Other" category was generally reserved for meaningless comments such as "N" which we received and thought was rather ambiguous. Select comments (misspellings and all) regarding "Responses" are as

follows: “It gives very specific details in the responses with exact dates and places,” “It recognizes key words well and gives good replies” and “It is like dictionary search engine. So stornig about definition.”

We had expected that users would rate the system solely based on the chatterbot responses given. However, a minority of participants in the Domain chatterbot instead rated it favorably based upon the system’s ease of use. If we were to look at only those values that dealt with Responses, the Domain chatterbot would still be preferred 33.3% to ‘Both’ at 28.6%. In looking further into the comments, two key findings emerged:

- Users were uncomfortable with the system returning definitions instead of conversation
- Users seemed to like the breadth of knowledge and speed of the system

Several users made comments that the system felt more natural when it is not talking about terrorism and they were displeased with its definition-style approach rather than placing knowledge in the context of the conversation. Others were impressed with the level of detailed knowledge they received from their queries.

The other non-cooperative category of recommendation potential was very similar to what we found in user comfort. Table 3.5 breaks down user comments between the Domain and ‘Both’ chatterbot. Again a minority influence of system ease crept into the results.

User comment groupings for 'Recommendation potential'				
	Domain chatterbot		'Both' chatterbot	
	Recmnd:Yes	Recmnd:No	Recmnd:Yes	Recmnd:No
Ease of use	2	1	0	0
Responses	4	17	6	22
Other	1	0	1	0

Table 3.5. User comments about system recommendation

A minority of users again cited ease of system use instead of the expected category of Responses. However, if we were to only look at the value for Responses, this time the 'Both' chatterbot leads in preference 21.4% to Domain's 19.0%, as shown in Table 3.6.

Modified 'Final Survey' - Responses Only	Chatterbot		
	Dialog	Domain	Both
Number of Users completing the survey	27	21	28
User Comfort	22.2%	33.3%	28.6%
Usability	11.1%	19.0%	28.6%
Recommendation Potential	22.2%	19.0%	21.4%

Table 3.6. User behavior towards the system

Table 3.6 shows a snapshot view of the final survey questions where only comments about the system responses were made. Even from this look, the 'Both' chatterbot was clearly not preferred as we had expected. Through analyzing the open-ended comments further, several key findings emerged:

- The system needs a larger corpus of knowledge
- Users appreciated the dialog (conversational) component of the system

The first finding directly contradicts one of the findings from user comfort. However, with approximately an equal number of users making comments both for and against the amount of terrorism-specific knowledge, it would appear that a sufficient level of knowledge was used in this experiment. The second finding was more obvious as users

pointed out that the dialog helped them to reform their queries in order to obtain their desired response.

From analyzing the three final survey questions, we found that users did not prefer the ‘Both’ chatterbot. In particular, it appeared from the comments that users disliked the domain’s definitional treatment of responses and would have preferred responses in a conversational context. This definitional treatment did emerge as a hindrance to the ‘Both’ chatterbot.

3.5.2 Dialog and Domain knowledge perform better together

For the second hypothesis, we believe that the ‘Both’ chatterbot will perform better than stand-alone Dialog and Domain. To test our assumption we use the results of response appropriateness and response rating. Tables 3.7 and 3.8 show the results of response appropriateness and response rating respectively.

Response Appropriateness	Chatterbot		
	Dialog	Domain	Both
Response Appropriateness	66.3%	21.6%	62.0%

Table 3.7. Response appropriateness across three chatterbots

Chatterbot	# Responses	Avg Response	Std Dev
Dialog	1,524	4.04	2.00
Domain	849	2.43	1.90
Both	1,138	4.21	2.20

Table 3.8. Response rating statistics

In analyzing the response rating from Table 3.8, the ‘Both’ chatterbot was preferred over Dialog and Domain (p-value < 0.001). But, in Table 3.7 the ‘Both’ chatterbot was second-best to Dialog (62.0% to 66.3%). The reader should be cautioned that the above

numbers are a bit misleading, because the ‘Both’ chatterbot is composed of elements from both dialog and domain. Table 3.9 places everything on equal footing by breaking the ‘Both’ chatterbot responses into its constituent parts for a more comparable analysis.

Breaking apart the numbers of 'Both'	Both's components	
	dialog	domain
Number of user queries	888	250
Avg Response Appropriateness	68.4%	39.6%
Avg Response Rating	4.51	3.14
Std Dev of Response Rating	2.12	2.17

Table 3.9. The ‘Both’ chatterbot’s dialog and domain responses

From Table 3.9, we can now compare the dialog component of ‘Both’ against the Dialog chatterbot which has a response appropriateness of 68.4% compared to 66.3% (from Table 3.7, p -value < 0.001). Similarly, we compare the domain component of ‘Both’ against the Domain chatterbot and obtain a response appropriateness of 39.6% and 21.6% respectively (p -value < 0.001). From this analysis we see that the dialog and domain components of ‘Both’ are performing better than the Dialog and Domain chatterbots.

3.5.3 Wh* Interrogatives were popular in user queries

In hypothesis H3, we believe that the interrogatives beginning with ‘wh*’ will be most frequently observed. To test this hypothesis we look at the 1,138 user inputs given to the ‘Both’ chatterbot. From this number, 250 or 22.0% were terrorism-related. Of the 250, 171 or 68.4% were interrogatives. Eighty-eight of 171 interrogatives began with the ‘wh*’ interrogative (51.5%). Table 3.10 lists the most frequently encountered interrogatives.

Sentence Starter	# Instances	% of Total
What	47	27.5%
Do	27	15.8%
Who	19	11.1%
How	14	8.2%
Where	10	5.8%
Is	9	5.3%

Table 3.10. The most popular interrogatives used

It was surprising to note the high percentage of ‘Do’, ‘How’, and ‘Is’ instances. These words were not entirely expected in our study. The other fact worth mentioning was that ‘Who’ questions fetched the highest response appropriateness and rating scores. It would appear that ‘Who’ questions were well-represented by the AIML knowledge.

Looking at the dialog side of ‘Both’ produces a similar picture. There were 245 ‘wh*’ interrogatives posed in the Both-dialog and of those 136 began with ‘what’ (55.5%). In looking at those 136 instances of ‘what’, 57 contained a terrorism keyword but failed to return a terrorism-related response. Looking further into the 57 terrorism classification failures, the following informal observations were made:

- 31 of the failures were due to an insufficient knowledge base
- 10 were from overly complicated knowledge base entries
- 5 were from improper knowledge base formatting
- 1 was from a better match found in the Dialog AIML

Addressing these failure types regarding knowledge base entries as well as using the interrogative frequency counts from Table 3.10 will help refine future systems. Efforts should be made to concentrate similar knowledge gathering activities on knowledge-base rules that begin with the ‘what’ interrogative. In summary, we found that ‘wh*’

interrogatives made up a majority of user questioning and furthermore, the word ‘what’ was the most frequent interrogative sentence starter.

3.6 Conclusions

In conclusion, several key findings were made. The first finding was that users appear to prefer a natural flow of conversation over a definitional approach. This means that the terrorism knowledge bases need to be adjusted to reflect a more conversational tone. Perhaps in the future a more careful screening of domain-specific terms and a post-acquisition filter can format the knowledge to fit the demands. The second finding was that the components of the ‘Both’ chatterbot performed better together than apart. This finding is supported eloquently through the comments of one user that said the dialog kept the user “on track” throughout the conversation. Whenever the user entered a query the system did not understand, the dialog part helped the participant to reform their question to obtain a desired answer. The third finding is consistent with the view of Voorhees that interrogatives are a major source of user inquiries. The ‘wh*’ interrogatives and ‘what’ in particular described a substantial number of user queries (Voorhees, 2001) and should be the focus of future knowledge gathering activities.

We suggest some future attention to be given to spell-checking and a larger knowledge base source. The first suggestion is to implement a spell-check mechanism on user input. Although it may have made a small improvement in our study, the trivial nature of programming would be well worth the effort. The other area is to investigate the use of a larger corpus of knowledge. In our study, the amount of knowledge appeared

to be appropriate, however, the more knowledge a system has the more potential accuracy of correctly answering highly specific questions the system could have.

From this study we have shown that off-the-shelf ALICEbots can function adequately in the terrorism domain by providing a channel of relevant information to the public. To meet the other challenges of a C3 system, future efforts should also investigate the use of ALICEbots to disseminate specific information to targeted individuals, as well as its ability to harness other data sources such as the “I’m Alive” boards to communicate the status of victims to family members outside of a disaster area. We believe that these efforts to establish a working C3 system will be well worth it.

CHAPTER 4: DELIVERING SMALL-SCALE KNOWLEDGE TO USERS

This chapter explores a lightweight and flexible dialog framework based on the ALICE system and evaluates its performance in chat and knowledge delivery using both a conversational setting and a specific telecommunications knowledge domain. Metrics for evaluation are presented and the evaluations of three experimental systems; a pure dialog system, a domain knowledge system and a hybrid system combining dialog and domain knowledge are highlighted and discussed. Our study of 257 subjects shows approximately a 20% user correction rate on system responses. Certain error classes such as nonsense replies were particular to the dialog system, while others, like mistaking opinion questions for definition questions, were particular to the domain system. A third type of error, wordy and awkward responses, is a basic system property that was found to span all three experimental systems. We also show that the highest response satisfaction results are obtained when coupling domain-specific knowledge together with conversational dialog.

4.1 Background

The World Wide Web is a vast distributed network of information. Users are constantly accessing and try to make sense of the Web's content using a variety of tools, such as search engines and digital libraries. The explosion of Web content volume coupled with the increasing ease of access to high speed bandwidth, means that researchers have a renewed focus on the design and implementation of knowledge delivery platforms. In its simplest form, this may be a digital library where access tools

facilitate the one-way flow of documents from the corpus to the end user. Another more dynamic approach is to allow the end users to be secondary contributors of information. This has been seen in electronic marketplaces of expertise such as Answer Garden (Ackerman and Malone, 1990; Ackerman, 1998) and the Annotate! system, which allowed organizational workgroup-level document annotation to augment search engine results (Ginsburg and Kambil, 1999). In situations where all participants are potential information donors, coordination mechanisms are critical between the primary content authors and system administrators who are responsible for managing the knowledge-bases as they scale upwards.

Given a specific domain of interest and its audience pool, there are two important aspects of a networked knowledge transfer platform. We have knowledge delivery, where the system is able to answer a broad range of questions within the domain to the satisfaction of a broad range of the audience pool and knowledge acquisition, where the audience can contribute ideas to the system's knowledge base for the subsequent benefit of all.

One stream of work on these issues in the 1990s has focused on building a controlled vocabulary system, which is domain rich and can help users with specific goals (Ferguson, Allen et al., 1996; Ferguson and Allen, 1998). These approaches can lead to maintenance difficulty and semantic drift as vocabularies change and evolve (Pejtersen, 1998).

Another approach is to utilize an Internet audience to build a large-scale information resource of interest, without specifying a priori the nature of the task an individual might

have in mind when he or she accesses the resource. WordNet, OpenCYC and Wikipedia, while adopting differing implementation philosophies (Lenat, Miller et al., 1995; Wagner, 2004) all depend on large numbers of users to build the resource; a dictionary in the case of WordNet and a freely available encyclopedia in the case of OpenCYC and Wikipedia. In this chapter we will not consider large-scale knowledge acquisition with its related social problem of contributor credibility and editorial effort, but instead focus instead on knowledge delivery.

One platform for knowledge delivery is a lightweight dialog system that will hold the user's attention with human-like responses. The ALICE system (Artificial Linguistic Internet Chat Entity) (Wallace, 2004) was created by Richard Wallace uses an XML dialect called AIML (Artificial Intelligence Markup Language) to store patterns and responses upon encountering the pattern in a user dialog. Its standard distribution comes with approximately 24,000 patterns covering assorted geography, nature and human interest facts. In addition, ALICE has mechanisms to engage the participant in conversational small talk and supports access to third party networked resources via Rhino, a Java-like Javascript dialect. A test implementation linking the Java ALICE ProgramD to a set of Web services in a prototype portal was demonstrated in The Catacomb Project (Ginsburg, 2002). The distribution² also provides automated dialog logging, archiving and visualization using XML and XSL.

² There are various ALICE distributions freely available via <http://www.alicebot.org> (Python, C++, SETL, Java and more).

In the remainder of the chapter we describe related work, including quite a few implemented systems in the Literature Review. We then describe our research questions as well as our AZ-ALICE system. Following that we present our experimental design and an analysis of the experimental results. Finally, we conclude our study and provide details of further development for this platform.

4.2 Related Works

Dialog Systems can be divided into two main groups; the Theoretical and the Performance-driven (Russell, 2002). The Theoretical or High-level systems involve symbolic reasoning and a deep understanding of user input. The Performance or Low-level systems forgo syntactic analysis and understanding for a much simpler pattern-matching algorithm. In both of these systems there are two elements that differentiate them, the level of analytical complexity and the level of complexity to understand context. High-level systems try to maximize these variables, while Low-level systems minimize them for performance gain. Hybrid, or mid-level systems, lie between these extremes and strike a balance between both reasoning and performance.

4.2.1 High-level Dialog Systems

High-level dialog systems are sometimes referred to as Integrated Artificial Intelligence (Ferguson and Allen, 1998). These are systems that possess planners, learning algorithms, speech recognition and temporal reasoning. The goal of these systems is to operate at the cognitive levels of humans or to at least understand the complexities of human behavior (Cassimatis, Mueller et al., 2006). Another feature is

that they maintain state across user sessions, remembering which tasks have been accomplished and which ones remain. Because of their complexity, they typically focus on narrow bands of knowledge. Examples of include TRIPS and TRAINS.

TRIPS, The Rochester Interactive Planner System, is an interactive spoken dialog collaborative transportation logistics planning assistant for crisis management (Ferguson and Allen, 1998). This system behaves as a crisis assistant on a mythical island, handling all the transportation-related logistics of an evacuation. Although this system does possess some contextual understanding and reasoning abilities, the system is not generalizable and takes a considerable amount of time to react to new stimuli, such as a bridge being suddenly destroyed.

TRAINS is the precursor to TRIPS and involves logistical routing of locomotives using a pre-defined map. As compared to TRIPS, TRAINS uses simpler route planning and a simpler problem solving engine. However, TRAINS is also limited in scope and is incapable of handling unexpected environmental changes.

4.2.2 Mid-level Dialog Systems

Mid-level dialog systems are those that can respond to a variety of requests about a task or domain. They will also typically have somewhat advanced reasoning abilities (Flycht-Eriksson and Jönsson, 2000). These types of systems can also take advantage of external resources such as Cyc or WordNet in their information gathering processes. The breadth of knowledge encompassed by these systems typically exceeds that of High-level systems but is less than Low-level systems. These systems do not typically maintain

state between user sessions. Some examples of these systems are MALIN, Lucy and Koko.

MALIN, the multi-modal application of the LINLIN programming language, is a bus time-table system that keeps track of bus routes and stations in Östergötland (Flycht-Eriksson and Jönsson, 2000). This system is less sophisticated than High-level systems which might plan, route or remember previous itineraries. However, it does use natural language processing to answer simple user queries such as “Which bus passes the North gate,” and “Are there any bus stops near the Garden square” (Flycht-Eriksson and Jönsson, 2000).

Lucy and Koko are both semantic interpreters of the English language (Barnett, Knight et al., 1990). Lucy attempts to process unfamiliar inputs into the CycL language while Koko performs the inverse function of taking CycL mappings and attempting to construct English sentences. Both of these systems rely upon the large Cyc knowledge-base and thus operate in a much broader environment than MALIN.

4.2.3 Low-level Dialog Systems

Low-level dialog systems are those that seek to mimic conversation rather than understand it (Hutchens and Alder, 1998). These systems employ simple algorithms to return dialog with minimal or no maintenance of state and thus no consideration of context. Because they are preprogrammed and have a large store of canned responses, these systems can be entertaining in a large variety of conversational topic settings.

Examples of these systems include MegaHAL and ALICE, which won the Loebner Prize in 2000, 2001 and 2004 for the most human-like computer.

MegaHAL, winner of the 1998 Loebner Prize for the most realistic human imitator, uses a method of Markov modeling to generate responses (Hutchens and Alder, 1998). In this system a keyword in the user input is isolated and many Markov chains are assembled before and after the keyword. Through a process of relevance ranking, the response with the highest information content most applicable to the input is then returned to the user. This system learns from its interactions although it returns mostly nonsense replies.

ALICE uses simple pattern-matching of user input to predefined inputs and then returns a response to the user (Wallace, 2004). This system is simple to administer and can be quickly adapted to new knowledge domains, such as customer service agents (Chavez, 2006).

There are some interesting studies of ALICEbots worth mentioning. One such study used an ALICE system to help Chinese university students practice their conversational English skills (Jia, 2002). The study was qualitative in nature and used pre-existing conversational knowledge bases. While it was not very systematic, it did yield some interesting results. Users generally made a high number of 'bad comments' about the system and on average chatted for about 5 lines.

There are two studies that focused on using ALICE to augment or enhance an existing subject. One study focused on using ALICE systems to tutor students in Euclidean Geometry (Han and Kim, 2001). In this paper the author argues for the value of

combining domain knowledge with conversational knowledge such that the system exhibits some form of personality and can respond to questions outside of the knowledge domain. It is further posited that the conversational knowledge assists users in their decision making skills.

Another tutoring study focused on using ALICE as a course enhancement tool with Social and Political Theory knowledge (Moore and Gibbs, 2002). This study found that most subjects used the system as a search engine rather than as a conversational partner. It was further concluded that their system was unable to function as a stand-alone tutor.

4.2.4 Challenges of Low-level Dialog System Analysis and Validation

There are several types of challenges in constructing and evaluating Low-level dialog systems. One of these is behavioral, where users might elect to insult the system and develop negative or sometimes abusive attitudes towards the system (De Angeli, Johnson et al., 2001). De Angeli conducted behavioral studies using ALICE and discovered that friction arose from power differences between users and the system, where users were trying to exert their dominion of control over the system. From De Angeli's work it was found that some users will promote an abusive environment to establish their dominance. In addition, users will sometimes use the system in unintended ways (Moore and Gibbs, 2002). In cases like these, further exploration is needed to see whether the system is lacking in topical knowledge, user training is inadequate or it is an example of user abuse.

Another challenge is systemic. A simple pattern matching approach relies on pre-built input and output responses. Unfortunately this means that there will be some queries that the system cannot properly respond to. Using Zipf's law of distribution for English words, it has been found that 2,000 words cover 95% of the first words typed into the ALICE system (Wallace, 2004). Naturally gaps remain and a challenge is to cover these gaps to improve user satisfaction.

A third challenge comes from the lack of systematic study in Low-level dialog systems, such as the ALICE system. Most prior ALICEbot studies have been qualitative and exploratory in nature, lacking objective, quantitative, field-based validation.

4.3 Proposed Approaches

Dialog systems have two distinct modes of operation. First, they perform conversational interaction to engage the user in discussion. Second, they can be tailored to perform knowledge delivery by responding to specific queries and supply domain knowledge.

To address the research gap on the design and evaluation of Low-level dialog systems as conversationalists and as on-demand knowledge suppliers, we build an ALICEbot platform to deliver domain knowledge in addition to differing gradients of conversational ability. From the work of Han (Han and Kim, 2001), we would expect that full conversation will perform best. We address these issues in the following research questions:

- How accurate is a dialog system such as ALICE, and what types of error/deficiencies will occur?

- How will a Dialog System perform with:
 - ... strict conversational components?
 - ... strict domain knowledge and sparse conversation?
 - ... a hybrid approach involving full conversational and domain knowledge components?

Analysis of these research questions will point out the pros and cons of each approach and pave the way for knowledge delivery extensions.

The AZ-ALICE dialog system is built on the Java-based ProgramD implementation that is freely available at www.ALICEbot.org. AZ-ALICE can be broken into five component parts: Chat User Interface, Chat Engine, AIML (Artificial Intelligence Markup Language) files, Logging and a component for Evaluation.

The Chat User Interface uses an XML web page that allows the users to authenticate and converse with the system. The system also passes the authenticated username to the Chat Engine where it is stored as a state variable to allow for a more personalized conversation (e.g., Hello Bob389, how are you today?).

The Chat Engine contains the AIML knowledge-base and conducts simple pattern-matching from the user's input to pre-defined AIML categories and then returns the corresponding response.

In our AZ-ALICE project we created a separate Telecommunications AIML knowledge set that contained 298 new categories to manage the Telecommunications knowledge domain. The Logging component enables the system to keep a user-specific record of discourse for future use. Finally, the Evaluation module allows users to

evaluate and correct each system response. The evaluations are then later used for assessing the performance of each system.

Our AZ-ALICE implementation differed from ALICE ProgramD in several ways.

Table 4.1 illustrates the differences between the two systems.

	Chat Interface	Chat Engine	AIML	Logging	Evaluation
Original ALICE	Uses XML to chat with users through a crude Jetty interface	Uses off the shelf ALICE Program D	Typical setup uses Standard AIML	Logs everything to a monolithic XML Log file	Does not provide a user-based evaluation component
TARA	Uses a customized perl skin to chat and for evaluation purposes	Same as Original ALICE	Customized Terrorism AIML for domain knowledge and Standard and Wallace AIML for conversation	Keeps XML logs on a per user basis	Customized perl script that allows users to evaluate and suggest new patterns

Table 4.1. Differences between ALICE ProgramD and AZ-ALICE

Although the interfaces between the ALICE ProgramD and AZ-ALICE systems differ, the underlying information transfer between the interface and the chat engine remains the same.

To illustrate user/system interaction, consider the following real interaction complete with misspellings and poor grammar.

```
User: What is it like? I am a very emotional person so I can not
      imagin having no emotions.
AZ-ALICE: Do you think I should know that? You are a emotional
          person so I can not imagin having no emotions? Is that your
          job?
```

Here are the steps that led to the AZ-ALICE response.

- The User inputs the sentence into the Chat Interface which then passes it along to the Chat Engine.
- The Chat Engine traverses a graphical representation of AIML nodes in memory.

- A match is found: the first sentence to the AIML category of “What is *” where ‘*’ is treated as a wildcard.
- The chat engine passes the associated response of “Do you think I should know that?” back to the Chat Interface and in addition echoes part of the user’s original chat as a technique to keep the conversation going.

4.4 Experimental Design

The main thrust of this chapter is to analyze the gradients of conversational knowledge in a domain-specific chatterbot. To do this, we created three chatterbot systems that differed only in their knowledge bases; BaseBot, TeleComm (Full) and TeleComm (Limited). BaseBot, the conversational and control chatterbot, used Standard AIML files which consisted of 23,735 knowledge-base entries. TeleComm (Full) contained 298 telecommunications-related definitions plus the same conversational knowledge as BaseBot. Finally, TeleComm (Limited) contained 298 telecommunications definitions and only a very rudimentary conversational capability of 3,892 knowledge-base entries. We decided that because of the limited amount of definitional knowledge in TeleComm (Limited), rather than run the potential risk of not returning any responses to participants and thus create alienation within our study, we instead chose to place TeleComm (Limited) in the role of providing full TeleComm knowledge and a limited array of conversational ability. From this position, we can test the effects of two different degrees of conversational knowledge coupled with full domain knowledge. For our limited conversational knowledge, we selected a subset of conversational AIML that from the work of Richard Wallace, should be able to adequately answer 65% of the

queries given to it. These files accounted for the 3,892 conversational knowledge-base entries. Table 4.2 shows the breakdown of categories between each of the systems.

System Name	Std AIML	Telco AIML	Total # of Categories
BaseBot	23,735	0	23,735
TeleComm (Full)	23,735	298	24,032
TeleComm (Limited)	3,892	298	4,190

Table 4.2. The category breakdown between systems

From Table 4.2, the total number of categories for TeleComm (Full) does not equal the true summation of Standard and Telecommunications AIML entries. This was because one knowledge base entry overlapped between both knowledge sources. When this happens the ALICE chatterbot will automatically omit the second instance thus decreasing the total number of categories available.

4.4.1 Performance Metrics

To gauge system performance, we measured the following evaluation variables; Correction Rate, Response Satisfaction and Classification of User Inquiries.

Correction Rate is defined as the number of system responses that were corrected by the user, divided by the total number of interactions typed into the system. Note that the act of correction requires time and effort on the users part and thus the user may elect to bypass possible corrections he or she judges to be less pressing. This follows the conclusions drawn on prescriptive Restrictiveness Theory whereby limiting the amount of decision control allowed to the users, may serve to discourage system use (Silver, 1990).

Response Satisfaction is a measure of the appropriateness of system response given the context of the user query. This metric is evaluated by the users using a seven point Likert scale (1-strongly disagree to 7-strongly agree). The aggregate Response Satisfaction number is then a summation of all the Response Satisfactions divided by the number of interactions scored.

Classification of User Inquiries is the only measure that is not under the direct manipulation of users. In this metric, user inquiries are scanned for any Telecommunications keywords that appear in the Telecommunications AIML file and are then labeled as either conversational dialog or Telecommunications-related inquiries.

4.4.2 Participants

We assigned each experimental system to a different section of an Introductory Management of Information Systems course, such that participants would interact with only one of the chatterbots. Students, mostly freshman and sophomores, were instructed to interact with the system for approximately ½ hour and then evaluate all of the system responses for their particular session. They were also instructed to provide a Response Satisfaction score for each response and were given the opportunity to provide an alternate system response to their particular query. Students were further instructed to focus their topic of conversation on Telecommunications knowledge, however, they were not forced to do so. All students were given an incentive through the awarding of bonus points for completing the study. Participation in the study was voluntary and students were permitted to chat from any computer terminal they wished which allowed them to

span multiple chatting sessions. Student subjects were selected based upon their availability and represented a computer-literate demographic that is likely to use chatterbot entities. Table 4.3 shows a breakdown of the number of participants for each of the three systems.

System Name	Number of Study Participants	Number of Interactions
BaseBot	74	9,751
TeleComm (Full)	91	10,179
TeleComm (Limited)	92	10,005

Table 4.3. Study Participants by System

Table 4.4 shows a breakdown of Interaction groupings between the three chatterbots. The average number of Interactions per chatterbot are provided as an aid.

Number of Interactions	BaseBot	Tele-Lmtd	Tele-Full
Average number of Interactions	131.8	108.8	111.9
Less than 50	9	13	10
Between 50 and 99	23	30	33
Between 100 and 149	24	33	25
Between 150 and 199	6	7	18
Between 200 and 249	5	7	1
Between 250 and 299	3	1	3
Greater than 300	4	1	1

Table 4.4. Interaction Breakdown

4.5 Experimental Results and Discussion

4.5.1 Knowledge Delivery Accuracy was approximately 80%

- How accurate is a dialog system such as ALICE, and what types of errors/deficiencies occurred?

To answer the first part of this question we need to take a look at each of the system's total Correction Rate values irrespective of conversational dialog and domain knowledge; and assume that system accuracy can be modeled as one minus the Correction Rate. We acknowledge that measuring Accuracy as $1 - \text{Correction Rate}$ can be problematic in an absolute sense, however since our measure is consistently applied between chatterbots, the relative measure is interpretable and has meaning. Table 4.5 provides a summary of system accuracy.

System Name	Correction Rate	Accuracy
BaseBot	17.5%	82.5%
TeleComm (Full)	16.0%	84.0%
TeleComm (Limited)	22.2%	77.8%

Table 4.5. System Accuracy as a measure of Correction Rates

AZ-ALICE's total accuracy figure hovers near 80% which might be satisfactory in general conversation but not for information retrieval. An example conversational interaction is shown below.

User: He did a good job teaching you.
 AZ-ALICE: Yes he did didn't he.

For the second part of our research question, *what types of errors/deficiencies occurred*, we take a qualitative approach by categorizing the most prominent types of response errors that were observed in the system logs. Table 4.6 outlines the observed chatterbot response errors in both BaseBot and TeleComm (Limited).

Overall	BaseBot	TeleComm
Nonsense responses	35.7%	34.5%
Spurious user corrections	28.6%	6.9%
Wordy and awkward responses	28.6%	13.8%
Application or Opinion type questions	7.1%	20.7%
Incomplete lexicon of terms	0.0%	24.1%

Table 4.6. Observed chatterbot response errors

4.5.2 Conversational Dialog Categories

For BaseBot, the most common error categories are; nonsense responses, spurious user corrections and wordy and awkward responses.

4.5.2.1 Nonsense responses

For nonsense responses, this type of error generally occurred when the system reflected the user input and either selected an improper pronoun or used a sentence fragment in an improper context. The following interaction is a good example of this type of deficiency.

```
User: Did you buy your car yourself?
AZ-ALICE: No I didn't buy your car yourself. What's buy your car
yourself like?
```

In this interaction the system incorrectly used the pronoun of 'yourself' and ended up with a nonsense reply.

4.5.2.2 Spurious User Corrections

The second conversational dialog error type, spurious user corrections, occurred when users offered unnecessary and spurious corrections to the system responses – corrections which do not benefit future users of the system. The following illustrates this category.

```
User: do you like metallica?  
AZ-ALICE: Not since they put pressure on Napster.  
User Corrected Response: YES they are the BEST band in the  
history of humankind.
```

This interaction demonstrates that the original response makes sense in the context of user input, however, the user decided to insert a more entertaining response for future system interactions. It was interesting to note that this category would usually have a higher than average Response Satisfaction score.

4.5.2.3 Wordy and awkward responses

For wordy and awkward responses, the AZ-ALICE system glued together non-sequitur sentences and sentence fragments when confronted with complex sentences containing multiple subject/verb pairs. Here is an example.

```
User: Yeah everyone else is sorry when they do hear me sing.  
AZ-ALICE: Yeah I know what you mean about that. That was not a  
yes or no question. That's a rather sweeping generalization.
```

The above interaction shows the system picking three separate matches keying in on different words from the user's input.

4.5.3 Telecommunications categories

The telecommunications knowledge domain had its own class of errors and deficiencies; users would ask application or opinion oriented questions rather than definitional ones, the system held an incomplete lexicon of terms or the system would provide nonsense responses.

4.5.3.1 Application or opinion type questions

The error category of application or opinion type questions houses all of the ways students would use telecommunications keywords outside of a definition type context.

User: Is internet telephony useful?
 AZ-ALICE: The transmission of voice/data over long distances.

In this interaction, the system recognized the term ‘telephony’ and returned the predetermined AIML definition. Even though subjects were instructed to ask definition type questions, this category of error was nevertheless widespread.

4.5.3.2 Incomplete lexicon of terms

In the telecommunications error category of incomplete lexicon of terms, there was some discourse in which users would ask obvious telecommunications definitions for which the system possessed a response, however, the system would instead key in on a different part of the user input and ignore the telecommunication term.

User: How does an IP address work?
 AZ-ALICE: How should I know it does?

In this example, the system matched ‘How does *’ rather than ‘* IP *’. These types of mismatch problems, while correctable by AIML maintenance, are common to a conversational system.

4.5.3.3 Nonsense responses

The error category of nonsense responses is the only category that spans both the conversational dialog and Telecommunications knowledge domains. Nonsense responses are a problem with the ALICE ProgramD chat engine, particularly when the elicitation of

particular domain-related answers are desired. Again, this is the trade-off between conversational entertainment and terse knowledge delivery. The other interesting item to note was that both of the other domain-related system problems stemmed from an incomplete set of domain answers.

4.5.4 Users preferred Domain responses to Dialog

- How will a dialog system perform with varying degrees of Conversational dialog and Domain Knowledge?

System Name	Conversational dialog	Telecommunications domain knowledge	Percentage Gain
BaseBot	4.3040	4.4321	3.0%
TeleComm (Full)	4.1565	4.6376	11.6%
TeleComm (Limited)	3.8225	4.4431	16.2%

Table 4.7. Response Satisfaction scores

† All pairwise comparisons were found to be with p-values < 0.001

The second research question is best answered by inspecting the user Response Satisfaction and Correction Rate numbers between the three systems. We were not as interested in the absolute values or motivations behind the Response Satisfaction scores, but more interested in the relative significance between them. All three systems were rated higher in Telecommunications knowledge (4.4321, 4.6376 and 4.4431) as compared to conversational dialog (4.3040, 4.1565 and 3.8225 respectively). The most notable differences came from the two TeleComm systems whose Telecommunications domain knowledge was a double digit percent gain in Response Satisfaction as compared to conversational dialog. TeleComm (Full) showed an 11.6% increase while TeleComm

(Limited) did even better with a 16.2% gain. Because of the large number of interactions, these values were found to be statistically significant. It would appear that users preferred the limited set of domain terms to conversational dialog. However, in defense of eliminating conversational dialog completely, the two systems that implemented full conversational dialog, BaseBot and TeleComm (Full) both performed better in conversational dialog, 4.3040 and 4.1565 respectively, than did TeleComm (Limited) with its relatively low Response Satisfaction score of 3.8225. This finding is of practical significance because domain-specific systems appear to perform better with a full complement of conversational dialog patterns to augment their domain-specific knowledge which is consistent with the observations of Han (Han and Kim, 2001).

System Name	Conversational dialog	Telecommunications domain knowledge
TeleComm (Full)	16.1%	15.1%
TeleComm (Limited)	22.3%	21.1%

Table 4.8. Abbreviated view of the Corrected Responses of Table 4.5

It is interesting to note that the conversational dialog in the two TeleComm systems also had a higher Correction Rate than their domain knowledge counterparts, as shown in Table 4.8. TeleComm (Full) conversational dialog had a Correction Rate of 16.1% as compared to its Telecommunication knowledge Correction Rate of 15.1%. Likewise, TeleComm (Limited) had a conversational dialog Correction Rate of 22.3% as compared to its domain knowledge Correction Rate of 21.1%.

Intuitively, it would seem that as Correction Rate drops, Response Satisfaction should increase. However, our BaseBot proved to be an exception to this rule as it had a much

higher Correction Rate and Response Satisfaction score in its Telecommunications domain (28.4% and 4.4321 respectively), than it did in conversational dialog (17.4% and 4.3040 respectively). As can be seen from Tables 4.7 and 4.8, Correction Rate and Response Satisfaction moved in the same direction rather than opposite ones. One explanation was that users liked the Telecommunications-related system responses but instead chose to amend them to reflect an elaboration or entertainment value. However, in a further examination of the results this did not appear to be the case. It was found that users were instead creating answers in the vacuum of BaseBot's Telecommunications knowledge and correcting compound and wordy responses. Another explanation is that the size of interactions with BaseBot's Telecommunications are simply too small (23 corrected Telecommunications-related responses) to make any kind of solid presumption. Although the limited rule set did make a fairly sizable and statistically significant impression on the two TeleComm systems, the level of Telecommunications interaction in proportion to conversational dialog was low. Subjects were instructed to limit themselves to Telecommunication definitions, but most students instead found the conversational dialog functionality after the first few telecomm interactions. It was interesting to find that even in the conversationally constricted environment of TeleComm (Limited), where conversational knowledge outnumbered telecommunications by nearly 13-1, subjects insisted on talking about non-telecomm topics, where conversational dialog accounted for 95.8% of TeleComm (Limited)'s interactions.

The experimental results included some examples of repetitive user input (just to satisfy the extra-credit incentive) and other examples of vulgar language. However, this may not be a limitation as much as expected student interaction with a system for a certain demographic subset.

4.6 Conclusions

In our experiment, using the results of Response Satisfaction, we found that domain knowledge is more effective in a chatterbot environment to obtain domain-specific knowledge than conversational knowledge alone. This comes from the breadth of knowledge that conversational knowledge would have to cover as opposed to the limited scope that domain knowledge encompasses. Further, we found that domain-specific knowledge coupled with conversational knowledge yields the highest response satisfaction scores. We feel that conversational dialog, while not strong on its own, is an important element in a domain-centric chatterbot.

We also found that the AZ-ALICE system is better suited to answering specific domain-related queries than performing as a general conversationalist. In our analysis of the domain dependent systems, it was found that both TeleComm systems had higher Response Satisfaction (4.6376 and 4.4431 respectively) and lower Correction Rates within their knowledge domain (15.1% and 21.1% respectively) than in the conversational dialog arena (4.1565 and 3.8225), and (16.1%, and 22.3%) respectively. We further found that conversational dialog is an integral piece of a system's repertoire. This element handles those user queries that either fall outside of the bounds of the

domain knowledge or are querying domain-specific knowledge that has not yet been entered into the system. Thus, a knowledge delivery system performs strongest when its domain knowledge is coupled with a storehouse of conversational dialog.

CHAPTER 5: DELIVERING LARGE-SCALE KNOWLEDGE TO USERS

This chapter investigates the effects of delivering large-scale Web Knowledge to Web Users. In particular, we study the effectiveness of such delivery systems within a natural language conversational environment and show that the techniques presented can be easily migrated to the business domain.

5.1 Background

Obtaining relevant yet concise information from online repositories has always been a problem. While search engines have mainly focused on the relevance aspect, they have not paid as much attention to conciseness. One way of addressing this issue has been through the use of Natural Language Dialog Systems (NLDS), where users can input natural language queries and expect to receive concise natural language responses.

One of the better conversationalists in NLDS is the ALICE chatterbot. ALICE, which stands for Artificial Linguistic Internet Chat Entity, is a type of dialog-driven chatterbot developed in 1995 by Richard Wallace (Wallace, 2004). ALICE chatterbots are built to function as general conversationalists, but they can be quickly supplemented with specific knowledge to function as a customer service agent (Chavez, 2006), information retrieval agent or language chatting partner (Russell, 2002; Wallace, 2003).

In this chapter we analyze data obtained from two similar studies conducted at The University of Arizona using modified ALICE chatterbots. While these two prior studies focused on the knowledge acquisition and delivery aspects of chatterbots in both the telecommunications and terrorism domains, this study differs by analyzing the types of

questions posed to the systems and the subject's response satisfaction levels to the chatterbot replies. This type of analysis will be able to provide guidance to future chatterbot knowledge developers as to what types of user queries are most common as well as what areas of knowledge development will need extra emphasis in order to provide the most satisfying responses. We further analyze the effects of query length on response satisfaction measures in order to judge its relative impact. We believe that answers to these questions can have a vital impact on the knowledge delivery activities for future chatterbot systems.

The rest of this chapter is organized as follows. Section 5.2 is a literature review and discusses the similarities of ALICE chatterbots and how they fit in the NLDS hierarchy. Section 5.3 asks a series of research questions about the retrieval characteristics of ALICE chatterbots and introduces the reader to our two prior ALICE chatterbot studies on telecommunications and terrorism knowledge. Section 5.4 is the experimental design segment and lays a framework for our current evaluation. Section 5.5 is our results section with a discussion of what the results mean. Finally, Section 5.6 wraps up with conclusions and possible avenues of future research.

5.2 Related Works

NLDSs form an interesting and dialog-oriented intersection between human-beings and computers. They allow for simple and natural communication, all while returning concise information to the user. These systems can be wholly automated to perform routine functions such as answer common questions, educate users about a particular

topic (Schumaker, Ginsburg et al., 2007) or be semi-automated as in case of a helpdesk assistant (Wallace, 2003). These systems can also be loaded with domain-specific knowledge adding to their flexibility of application.

One particular aspect of NLDS is Question Answer (QA) systems (Ferret, Grau et al., 2001). Figure 5.1 illustrates a synthesized QA framework extracted from the works of Voorhees, Pasca and Vrajitoru (Voorhees and Tice, 2000; Pasca and Harabagiu, 2001; Vrajitoru, 2003).

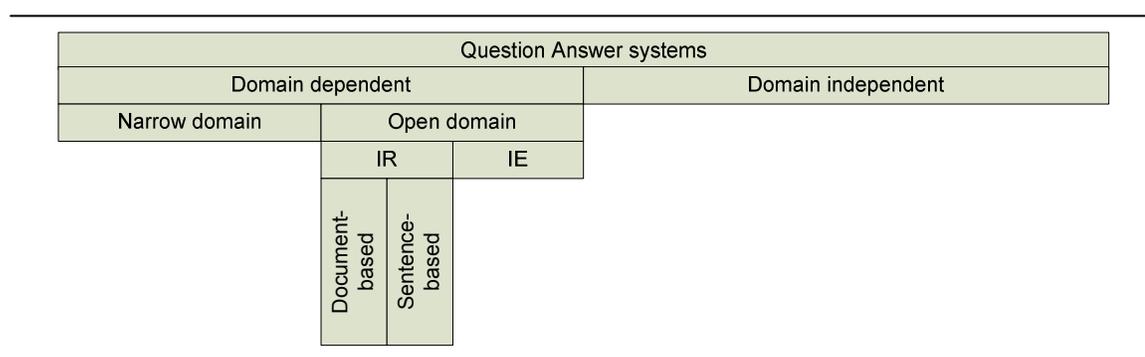


Figure 5.1. A Question Answer system taxonomy

This framework will describe each sub-area of QA systems; what they do, how they are important, how they differ from one another and provide an overview of how they all link together. We will discuss each component in detail below.

5.2.1 Question Answer Systems

QA systems use natural language processing methods to select answers based on a search of linguistic features (Ferret, Grau et al., 2001). These features can be syntactic, i.e., relying on the structure of the sentence such as NP-VP-NP patterning or semantic

where ontologies and similar corpora attempt to assign a meaning to the words (Lenat, Miller et al., 1995). These systems can also vary greatly on:

- Knowledge sources used (Domain dependent and independent sources)
- Breadth of domain expertise (Narrow and open domain systems)
- Type of information to obtain (Information retrieval and information extraction)
- And the type of response to give (Document and sentence-based IR systems)

Each of these QA system characteristics can be found as a separate entity in Figure 5.1.

5.2.2 Domain Dependent and Independent Systems

QA systems can vary based on the characteristics of their knowledge source(s), such as whether or not the knowledge-base was created specifically for computer usage. They can be classified in one of two different categories; domain dependent or domain independent systems. Domain independent systems use external knowledge sources, such as an online encyclopedia, that are not specifically built for computational consumption. One example of a domain independent system is MURAX which uses the online Grolier encyclopedia to answer its queries (Kupiec, 1993).

Conversely, domain dependent systems depend on specially tailored knowledge-bases (Voorhees and Tice, 2000). These knowledge-bases can be ontology-based as in the case of systems relying on Doug Lenat's Cyc knowledge-bases for semantic meaning and disambiguation (Lenat, Miller et al., 1995) or more simply a collection of domain-relevant data that a system uses to answer a specific question.

5.2.3 Narrow and Open Domain Systems

Domain dependent systems can be further broken into two subcategories: narrow and open domain systems (Voorhees and Tice, 2000; Pasca and Harabagiu, 2001). In narrow domain systems, the goal is to attempt conversational fluency in limited domains of expertise (Schumaker, Ginsburg et al., 2007). Example systems include STUDENT which solved algebraic word problems (Winograd, 1977), Winograd's SHRDLU which answered natural language queries about a fictitious Block World (Winograd, 1977) and LUNAR which responded to geological queries on lunar rock data (Woods, 1977).

Open domain systems possess a more diverse (i.e., generalized) repertoire of topics. These systems are not limited to any one particular area or domain. Instead, these systems can field questions from multiple disciplines and can be further classified into two major categories; information retrieval and information extraction (Voorhees and Tice, 2000; Pasca and Harabagiu, 2001).

5.2.4 Information Retrieval and Information Extraction

In Information Extraction (IE), the goal is to extract relevant contextual information from text and to fill that data into pre-defined templates. This field is well-represented by the Message Understanding Conference (MUC).

In contrast, Information Retrieval (IR) attempts to retrieve a whole or partial document for the user. Examples vary from modern search engines that rely on shallow keyword matching techniques to deeper systems that attempt to retrieve a snippet of text

to the user within the context of the query. This field is represented by the Text Retrieval Conference (TREC) (Pasca and Harabagiu, 2001; Potter, 2001).

5.2.5 Document and Sentence-based IR Systems

IR is composed of two smaller classes; the document-based and sentence-based retrieval systems (Vrajitoru, 2003). The objective of document-based systems is to return a set of relevant documents to the user, much like a search engine.

To the contrary, sentence-based retrieval systems return only a small snippet of text to the user. These systems can also vary in the styles of answers given, from binary yes/no or true/false responses (Voorhees and Tice, 2000), to those responses requiring a synthesis of material from various locations (i.e., “Why do terrorists hate the West”) (Moldovan, Pasca et al., 2003) and many others between these two extremes. ALICE chatterbots fit into the sentence-based retrieval category primarily because of their sentence-oriented response capability (Vrajitoru, 2003).

5.2.5.1 ALICE

ALICE uses XML-based Artificial Intelligence Markup Language (AIML) files to hold its internal collection of knowledge. This open-source knowledge-base makes ALICE robust and able to quickly extend into new knowledge domains (Wallace, 2004). ALICE seeks to mimic conversation rather than understand it (Hutchens and Alder, 1998). This method of conversational mimicry has allowed ALICE to win the Loebner Prize for most human chatterbot in 2000, 2001 and 2004. However, because of its simplistic pattern-matching mechanisms, ALICE lacks cognitive ability and will miss

certain types of interactions (Russell, 2002). Wallace argues that ALICEbots use Case Based Reasoning (CBR) to represent responses (Wallace, 2003), which is beneficial to system performance because CBR does not require the computational overhead that other reason-based systems would demand (Breese and Heckerman, 1996).

Prior studies have shown that ALICE has been used more like a search engine rather than a conversational tool (Moore and Gibbs, 2002). This finding complements many of the suggested uses of ALICE as a fact-driven conversationalist that can deliver domain-specific information to the public in a personable and tireless manner. Following ALICE's use as a search engine, a focus on interrogative usage is an important area of QA system research (Hammerton, Osbourne et al., 2002).

5.2.5.2 MegaHAL

A similar dialog system in the same family as ALICE is Hutchen's MegaHAL. MegaHAL was unique because of its use of Markov models to generate responses. At its core, the system would identify a keyword in the user input and then build two Markov chains, one in front of the keyword and the other behind it. The system would generate hundreds of these candidate responses and then select the one with the highest information content with respect to the keyword (Hutchens and Alder, 1998). This system also learned from its interactions by updating its list of Markov chains. Most of the chatterbot responses ended up as nonsense from a failure to realize the semantic context of response phrases (Chan and Franklin, 2003), however, this Markov chaining

method of response generation did succeed in allowing MegaHAL to win the 1998 Loebner Prize for most human chatterbot.

5.2.5.3 Other Chatterbot Studies

There are several other research studies where ALICE has been used. The first of which was an English and German conversational partner for Chinese students (Jia, 2002). This study focused on the usefulness of the ALICE platform as a stand-alone conversationalist and produced some unexpected results. Jia tracked the categories of topical discussion and found that participants most frequently discussed love, the study of the English language and friendship. However, a high proportion of students did not like the chatterbot responses and made 'bad' comments about the system. A majority of subjects interacted for a short period of time before leaving. This system used a smaller than available knowledge-base which may have directly contributed to the unfavorable study observations.

This anthropomorphism of a computer program as a social actor was not unexpected (Reeves and Nass, 1996). De Angeli conducted behavioral studies using ALICE and discovered that the friction arose from power differences between users and the system, where users were trying to exert their dominion of control over the system. From De Angeli's work it was found that some users will promote an abusive environment to establish their dominance. In addition, as stated by Pejtersen, users will sometimes use the system in unintended ways (Pejtersen, 1998; De Angeli, Johnson et al., 2001).

Another teaching-related chatterbot study was that of a Geometry tutor. This ALICE-based instantiation was a prototypical chatterbot designed to assist students with concepts in Euclidean geometry (Han and Kim, 2001). A more important realization from this study was that Han believed complementing Euclidean domain knowledge with conversational knowledge would augment the ability of the chatterbot to assist users in reformulating misunderstood queries within a natural conversational context. Han's belief was supported in a different study (Schumaker and Chen, 2007), where conversational knowledge coupled with telecommunications definitions performed better than the telecommunications definitions alone.

Another relevant study was the Emile chatterbot at the University of Huddersfield. Emile was designed to emulate four different social theorists and was offered to socio-political students as a teaching tool to better understand the different sociological perspectives (Moore and Gibbs, 2002). Students were given socio-political class assignments and were instructed to use the Emile chatterbot. Unfortunately it was determined that students were more interested in using Emile as a search engine to quickly answer the assignment questions rather than converse with the system as expected.

In a similar chatterbot study, AutoTutor was tasked with interacting with students on the topics of computer literacy and conceptual physics (Louwerse, Graesser et al., 2002). From an analysis of dialog interactions, it was again found that students were using the system as a search engine to find answers to definitional queries (Graesser, Person et al., 2001).

Following the findings of the Emile and AutoTutor studies, Voorhees noted that most search engine queries are definitional in nature (Voorhees, 2001). Queries such as *Who is Colin Powell* and *What is mold* are common types of interrogative-based methods of obtaining information (Voorhees, 2003).

In a similar search engine study, Zuckerman looked into the accuracy of returned results as a function of query length (Zuckerman and Horvitz, 2001). It was found that an inverse relationship exists between the two variables. As query length decreases, system accuracy improves. Zuckerman determined that shorter queries were more apt to provide more accurate results.

5.3 Proposed Approaches

From the data gathered on our prior ALICE chatterbot studies (Schumaker and Chen, 2007; Schumaker, Ginsburg et al., 2007), we pose several research questions which we believe will help provide some insight into the strengths and weaknesses of current ALICE knowledge-bases. We ask our questions in such a way that the answers generated can be directly used by future chatterbot developers to better position chatterbot responses.

The first question we ask relates to what question-types are most commonly posed to chatterbot systems.

- What similarities in interrogative selection frequency exist between different knowledge domains?

With this question, we examine the frequency counts of interrogative selection and determine if any interrogatives are used more frequently than others. The answer to this

question will tell us which question-types should be the focus in future knowledge-gathering activities. We adapt a naïve view and assume that interrogatives starting with ‘wh*’ will occur most frequently.

Our second research question looks at how well users perceive chatterbot responses to particular question-types.

- What interrogatives best answer user queries and why?

In this question, we seek to discover which interrogatives can best answer user queries. As a consequence, we can also examine which interrogatives cannot answer user queries, why they cannot and address possible ways to correct these instances in the existing ALICE knowledge-bases. We would assume that interrogatives that seek specific responses would return the best answers to the users (i.e., ‘what’ and ‘who’ versus ‘how’ and ‘why’).

For our third research question, we wish to examine how the length of a user query might impact a user’s satisfaction with a chatterbot response.

- How does the length of a query affect the satisfaction level of chatterbot responses?

In the work of Zuckerman and Horvitz on search engine queries, they found that shorter queries led to more accurate document-based results. We will apply this assumption to the sentence-based ALICE chatterbot and assume that shorter queries will also lead to more accurate chatterbot responses.

The chatterbot system we used was composed of five distinct components: Chat Interface, Chat Engine, AIML files, Logging and Evaluation modules. Figure 5.2 graphically illustrates the different chatterbot components.

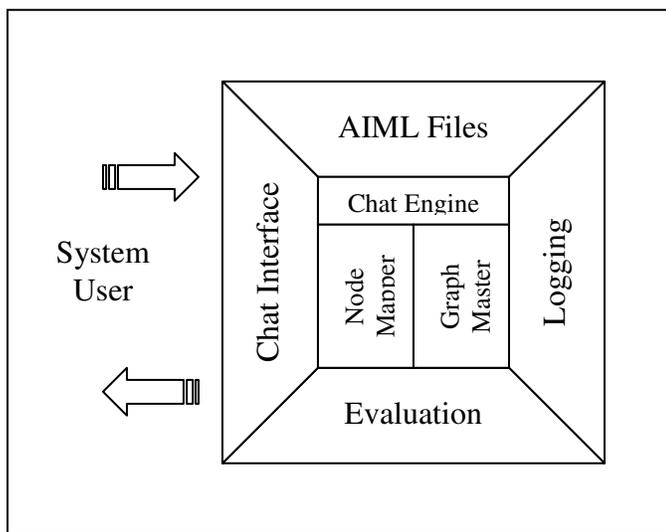


Figure 5.2. Graphic Illustration of Chatterbot Components

The first three components, Chat Interface, Chat Engine and AIML Files, are considered essential components to the ALICE chatterbot. The Chat Interface module allows the system to handle inputs and responses by performing appropriate conversions of textual data to more friendly XML-based content.

The Chat Engine is the core algorithmic component of the system and is comprised of two sub-entities; the Node Mapper and the Graph Master. When the system is initialized, the Node Mapper constructs a memory-resident directed graph of all the AIML patterns. Later, when a user poses a query to the system, the Graph Master will traverse the directed graph to best match the input.

The AIML files are considered to be the brains behind the system. Specific knowledge input patterns and appropriate chatterbot responses are stored within these files. This flexible arrangement permits the AIML-enabled system to easily migrate into new domains of knowledge with the addition of probable domain queries and responses. An AIML category used in the TARA terrorism knowledge-base is listed below. User queries are matched against patterns and chatterbot responses arise from the pre-defined templates.

```
<category>
<pattern>WHAT IS POTASSIUM IODIDE</pattern>
<template>FDA-approved nonprescription drug for use as a blocking
    agent to prevent the thyroid gland from absorbing radioactive
    iodine.
</template>
</category>
```

As an example of how this system works, consider the following query “Who is Bin Laden.” The Chat Interface passes the query to the Chat Engine, where the Graph Master resides. The Graph Master then sets out to best match the query from the most general terms to the most specific. Assume that an AIML node map can be represented in Figure 5.3.

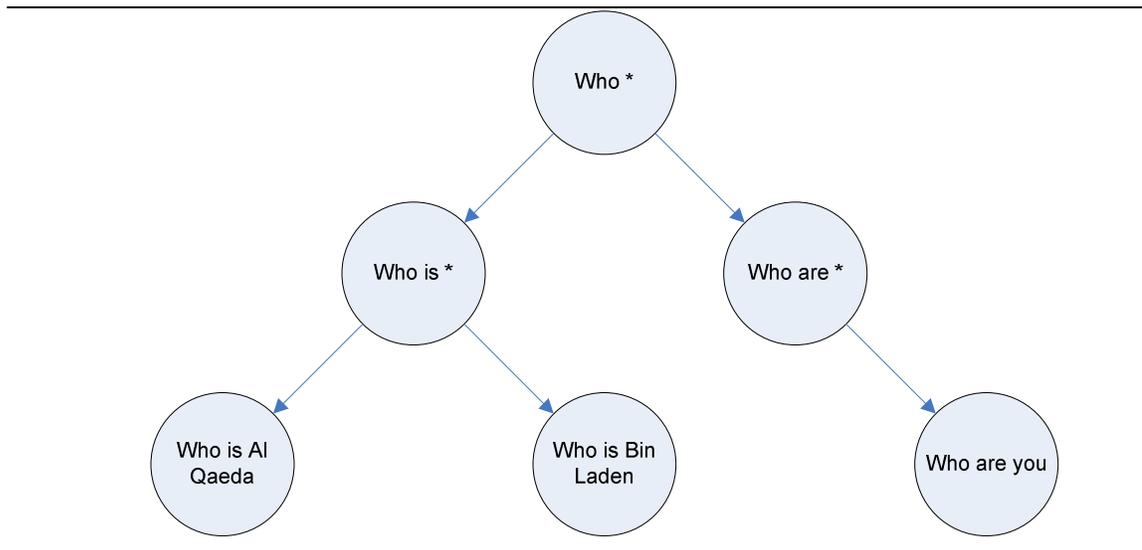


Figure 5.3. Example AIML node map

From this query, the Graph Master will first match the node “Who *” to the query where “*” is a wildcard placeholder. The Graph Master will then look at all of the child nodes to determine whether a better, more specific, match can be made. In this case “Who is *” is a better match. Continuing further, we find the exact match in the next level and the appropriate node-specific response is then sent back to the Chat Interface.

In a majority of cases, exact matches are rare. In cases of inexact matches, the Graph Master utilizes wildcard matching to best correspond to queries. Supposing that a query “Who is George Washington” was posed to our hypothetical system. Without having an exact match, the Graph Master will settle on the “Who is *” node and return it’s more generic response. It is believed that the more nodes available will directly increase the perceived depth of the system.

For our study we analyzed data gathered from two prior systems; AZ-ALICE and TARA (Schumaker and Chen, 2007; Schumaker, Ginsburg et al., 2007); which were both

closely based on the original ALICE system. Table 5.1 shows the similarities and differences between the three systems.

	Chat Interface	Chat Engine	AIML	Logging	Evaluation
Original ALICE	Uses XML to chat with users through a crude Jetty interface	Uses off the shelf ALICE Program D	Typical setup uses Standard AIML	Logs everything to a monolithic XML Log file	Does not provide a user-based evaluation component
AZ-ALICE	Uses a customized perl skin to chat and for evaluation purposes	Same as Original ALICE	Customized Telecommunications AIML for domain knowledge and Standard AIML for conversation	Keeps XML logs on a per-user basis	Customized perl script that allows users to evaluate and suggest new patterns
TARA	Same as AZ-ALICE	Same as Original ALICE	Customized Terrorism AIML for domain knowledge and Standard and Wallace AIML for conversation	Same as AZ-ALICE	Same as AZ-ALICE

Table 5.1. Difference between original ALICE, AZ-ALICE and TARA

In all three systems the Chat Engine remained the same. The only substantial difference between our studies was the AIML knowledge-bases used. AZ-ALICE focused on the telecommunications domain whereas TARA handled terrorism-related knowledge. Both systems utilized conversational knowledge in their control chatterbots; AZ-ALICE used Standard AIML and TARA used the Standard and Wallace AIML set. Depending upon the knowledge-bases used by a particular chatterbot, telecommunications, terrorism, general conversation or a mixture of both conversation and domain knowledge may be returned to the user.

5.4 Experimental Design

In order to design our experiment to provide meaningful results, we focus our attention on careful specification in three areas; AIML knowledge for the system, study participants and evaluation metrics.

5.4.1 AIML Knowledge

In both studies we used three chatterbots apiece where one chatterbot (the control chatterbot for its respective study) was devoted to conversational knowledge, one to specific domain knowledge and the third to a mixture of conversational and domain knowledge. Table 5.2 illustrates the breakdown of AIML category rules used for each chatterbot.

	Dialog categories	Domain categories	Total categories
AZ-ALICE - Dialog	23,735	0	23,735
AZ-ALICE - Domain	3,892	298	4,190
AZ-ALICE - Both	23,735	298	24,032
TARA - Dialog	41,873	0	41,873
TARA - Domain	0	10,491	10,491
TARA - Both	41,873	10,491	52,354

Table 5.2. Category Breakdown of the Six Chatterbots

From this breakdown, there are several differences that require a further explanation.

- **AZ-ALICE – Dialog:** This was the conversational or control chatterbot for the AZ-ALICE study. Its 23,735 categories were derived from the Standard AIML set which is freely available at www.alicebot.org. These are believed to be the same categories that Jia used in the English part of their conversational partner chatterbot (Jia, 2002).

- AZ-ALICE – Domain: This one was not a true domain-only chatterbot. It consisted of 298 domain categories as well as a limited set of dialog. Dialog categories were selected because of the limited responses from the domain categories. The dialog categories selected were the core categories that allowed AZ-ALICE – Domain to provide sufficient responses to a majority of questions. The domain categories were handcrafted and based on telecommunications knowledge.
- TARA – Dialog: This was the conversational control chatterbot for the TARA study. Its 41,873 categories were obtained from the Standard and Wallace set which helped ALICE to win its early Loebner contests.
- TARA – Domain: This chatterbot used true domain-only terrorism knowledge. Its 10,491 terrorism-related categories were gathered from the glossaries of several reputable terrorism websites. Further details of TARA’s domain knowledge can be obtained from (Schumaker and Chen, 2007).

The other item of interest is that the total number of categories does not necessarily represent the sum of dialog and domain categories. This is a result of the way the Node Mapper handles duplicate categories. There were several instances where domain knowledge was already represented by the dialog AIML. When this situation occurs, the Node Mapper automatically drops the duplicate node from the directed graph.

5.4.2 Study Participants

Participants from both studies were self-selected university students. Subjects for the AZ-ALICE project were obtained from several sections of a freshman-level Management of Information Systems (MIS) introductory course in Fall 2003, while TARA participants came from various undergraduate and graduate-level MIS classes in Spring/Summer 2004. While this subject-pool may initially signal problems when making study comparisons, we feel that the arbitrary rank of student subjects has no bearing on task

performance. For both of these studies we were more interested in approximating conditions similar to that of ‘the wild’ using a demographic subset that would be more likely to use such communicative instruments. Subjects were further assigned to one of the six particular chatterbots by the following metric:

- AZ-ALICE: based on which class section the student belonged
- TARA: based on a random assignment algorithm run against their University ID

In both studies, subjects were asked to interact with the system for approximately one-half hour and were given a participation incentive through the prospect of bonus points or random gift cards, depending upon the study. Subjects that participated in one experiment were not permitted to participate in the other. The number of study participants are shown in Table 5.3.

	Study Participants
AZ-ALICE - Dialog	74
AZ-ALICE - Domain	91
AZ-ALICE - Both	92
TARA - Dialog	30
TARA - Domain	30
TARA - Both	30

Table 5.3. Study Participants

In both studies, participants were asked to communicate with their chatterbot either on telecommunications or terrorism topics and then evaluate the responses of the chatterbot and rate their satisfaction level of the response using a one-to-seven Likert scale (one – strongly dissatisfied to seven – strongly satisfied). The evaluation phase did differ between studies, where AZ-ALICE asked for the response evaluation at the

conclusion of chatting while TARA presented its evaluation after each interaction.

Figure 5.4 illustrates the integrated evaluation mechanism from the TARA studies.

Please evaluate chatterbot response and click next.

You said: ***Who is Osama Bin Laden?***
 Chatterbot response: ***He is the world's most wanted man.***

Do you feel that the chatterbot response is appropriate given your input? Yes No

If no, please explain:

How would you rate your satisfaction level of the chatterbot response in the context of your input?

Very Dissatisfied Somewhat Dissatisfied Mildly Dissatisfied Neutral Mildly Satisfied Somewhat Satisfied Strongly Satisfied

Figure 5.4. Screenshot of TARA's evaluation process

5.4.3 System Evaluation Metrics

In this chapter we analyzed the data from AZ-ALICE and TARA and arrived at three metrics to answer our research questions; a count of interrogative usage, response satisfaction scores and the length of user queries.

The count of interrogatives was concerned about the frequency that interrogatives are used to begin user queries. Example types include the 'wh*' interrogatives: who, what, when, where and why. Queries ending with question marks are identified as interrogative and the starting words are tabulated in a frequency count. While we agree that identifying interrogatives by their punctuation characteristics is not ideal, this method has a prior basis (Uzuner, Davis et al., 2004).

Measurements of response satisfaction were common between studies. This metric was subjectively measured on a one-to-seven Likert scale by study participants for each

chatterbot input-response pair. Aggregate measures of response satisfaction were then composed for each chatterbot and different interrogatives.

Query length was a simple measure that averaged the number of words in a user input for each chatterbot. The response satisfaction of query length using different interrogatives was also investigated.

5.5 Experimental Results and Discussion

5.5.1 Participants sought definitional facts across knowledge domains

In analyzing the interaction patterns of users across the different chatterbots, it was discovered that participants were generally inquisitive. Table 5.4 shows a breakdown of chatterbot interactions as well as the percentage of user inputs identified as interrogative.

Chatterbot	# Interactions	# Interrogatives	Percentage
AZ-ALICE Dialog	9,751	3,906	40.1%
AZ-ALICE Domain	10,005	5,906	59.0%
AZ-ALICE Both	10,179	4,284	42.1%
TARA Dialog	1,524	931	61.1%
TARA Domain	849	487	57.4%
TARA Both	1,138	731	64.2%

Table 5.4. Participant usage of interrogatives across chatterbots

From this table, we found that participants of TARA were more inclined to use interrogatives than those of AZ-ALICE, with the exception of the two domain chatterbots (p -values < 0.01). It was interesting to note how much communication was question-oriented.

Looking further into the frequency counts of the various interrogatives used, eight particular words always appeared frequently in all six chatterbots: *are*, *do*, *how*, *is*, *what*,

where, *who* and *why*. For the AZ-ALICE studies, these words were consistently in the top 15 and for TARA they were the top 10 most frequently used interrogatives.

Investigating these terms further, Tables 5.5 and 5.6 illustrate the frequency counts of each of these interrogatives across all six chatterbots.

ALICE Studies - Telecommunications Domain					
Dialog		Domain		Both	
Interrogative	Count	Interrogative	Count	Interrogative	Count
What	985	What	2,159	What	1,405
Do	628	Do	725	Do	522
How	503	How	447	How	448
Who	168	Why	276	Who	188
Why	164	Who	261	Why	185
Are	130	Are	199	Are	158
Where	79	Is	126	Where	71
Is	49	Where	107	Is	59

Table 5.5. Interrogative frequency use of Telecommunications

TARA Studies - Terrorism Domain					
Dialog		Domain		Both	
Interrogative	Count	Interrogative	Count	Interrogative	Count
What	228	What	136	What	176
Do	112	Who	67	Do	106
How	57	How	44	Who	67
Who	57	Where	35	How	57
Is	50	Do	32	Where	38
Are	49	Are	21	Are	31
Why	40	Is	18	Is	24
Where	35	Why	11	Why	22

Table 5.6. Interrogative frequency use of Terrorism

From these two tables, the interrogative *What* was the most frequently used in all six chatterbots. It was interesting that several of the frequently observed interrogatives do not fit under the traditional ‘wh*’ interrogative family (i.e., *do*, *are*, *is*). *Do* was one such occurrence that was frequently second to *What* except in pure Terrorism domain

knowledge. Another interesting fact was the frequency of interrogative *Why* and the complete absence of *When*. *When*'s conspicuous absence was not expected. It would appear that temporal questioning was not very popular at least within these two domains of knowledge.

Since *What* appeared with the greatest frequency, we further investigated this interrogative by expanding its frequency counts to include the second word as well.

Tables 5.7 and 5.8 show the *What* expansion.

ALICE Studies - Telecommunications Domain			
	Count		
Interrogative	Dialog	Domain	Both
What is	404	1265	802
What do	148	227	114
What are	82	151	105

Table 5.7. *What* expansion frequency counts of Telecommunications

TARA Studies - Terrorism Domain			
	Count		
Interrogative	Dialog	Domain	Both
What is	63	69	69
What do	50	8	21
What are	34	13	23

Table 5.8. *What* expansion frequency counts of Terrorism

From this expansion, we found that *What is*, *What do* and *What are* were in the top 4 frequency counts for each chatterbot. Furthermore, *What is* appeared most often suggesting that users were seeking definitional types of responses.

5.5.2 *Are* interrogatives had the highest Response Satisfaction rating

Looking further into which interrogatives led to better chatterbot responses, we analyzed the Response Satisfaction levels for each of the eight frequently observed interrogatives. Tables 5.9 and 5.10 show the Response Satisfaction ratings for the various interrogatives.

ALICE Studies - Telecommunications Domain					
Dialog		Domain		Both	
Interrogative	Avg Rating	Interrogative	Avg Rating	Interrogative	Avg Rating
Are	4.7077	Are	4.4372	Do	4.6034
Is	4.6531	Where	4.2897	Are	4.5949
Where	4.5443	Who	4.2222	How	4.3705
Do	4.5000	Do	4.1310	What	4.2135
Who	4.4345	How	4.0962	Who	4.2128
What	4.3107	Why	3.9529	Where	3.9155
How	4.3101	Is	3.7063	Why	3.8541
Why	4.2439	What	3.6281	Is	3.6780

Table 5.9. Interrogative Response Satisfaction scores for Telecommunications

TARA Studies - Terrorism Domain					
Dialog		Domain		Both	
Interrogative	Avg Rating	Interrogative	Avg Rating	Interrogative	Avg Rating
Are	4.9388	Where	3.2571	Are	4.8387
Where	4.6286	Who	2.9552	Who	4.5821
How	4.1404	What	2.5074	How	4.2281
Who	3.9825	Are	2.0476	Where	3.9474
Do	3.9821	Is	2.0000	Do	3.9434
Is	3.9000	How	1.9545	Is	3.7083
Why	3.8750	Do	1.8438	What	3.6818
What	3.6798	Why	1.3636	Why	3.5909

Table 5.10. Interrogative Response Satisfaction scores for Terrorism

From examining the above tables, it becomes apparent that *Are* interrogatives appear to be more satisfying than the others in AZ-ALICE-Dialog, AZ-ALICE-Domain and TARA-Both (p-value < 0.001 intra-chatterbot). The *Are* interrogative was statistically

equivalent to *Do* in AZ-ALICE-Both and to *Where* in TARA-Dialog. The only case where *Are* did not outright rate the highest or statistically tie, was in the pure terrorism domain of TARA. In further probing, we found that *Are*'s success in dialog-enabled chatterbots appears to come from vague and general responses, such as offering a pre-programmed pseudo-opinion rather than a fact. These responses can be binary, non-sequitur or clichéd, yet still be highly satisfying (as shown in the examples below).

```
User: Are there any weapons of mass destruction in the middle
      east?
System: Yes I think there are.

User: Are you scared of terrorists?
System: I fear nothing.
```

To take our analysis further, we broke apart the Both chatterbots into their constituent parts of Dialog and Domain knowledge. From there we can cross-check our initial findings to determine what question types are best suited to the various conversation and knowledge domains. Tables 5.11 and 5.12 show the Both chatterbots on the left column and their breakdown into Dialog and Domain to the right.

ALICE Studies - Telecommunications Domain					
Both		Both (Dialog)		Both (Domain)	
Interrogative	Avg Rating	Interrogative	Avg Rating	Interrogative	Avg Rating
Do	4.6034	Are	4.6026	Do	5.8333
Are	4.5949	Do	4.5891	What	4.9545
How	4.3705	How	4.3647	How	4.5833
What	4.2135	Who	4.2246	Is	4.2000
Who	4.2128	What	4.1367	Why	3.2500
Where	3.9155	Where	3.9429	Are	4.0000
Why	3.8541	Why	3.8674	Where	2.0000
Is	3.6780	Is	3.6296	Who	2.0000

Table 5.11. Response Satisfaction breakdown of Both in Telecommunications

TARA Studies - Terrorism Domain					
Both		Both (Dialog)		Both (Domain)	
Interrogative	Avg Rating	Interrogative	Avg Rating	Interrogative	Avg Rating
Are	4.8387	Are	5.1481	Who	4.3684
Who	4.5821	Who	4.6667	Where	3.4000
How	4.2281	How	4.6512	What	3.3830
Where	3.9474	Do	4.3291	Is	3.1111
Do	3.9434	Where	4.1429	How	2.9286
Is	3.7083	Is	4.0667	Do	2.8148
What	3.6818	Why	4.0556	Are	2.7500
Why	3.5909	What	3.7907	Why	1.5000

Table 5.12. Response Satisfaction breakdown of Both in Terrorism

From the view given by the preceding tables, *Are* interrogatives appear to be more satisfying in dialog than domain knowledge (TARA has a p-value < 0.01 but AZ-ALICE lacks a sufficient number of instances to draw any conclusions). This is not an unexpected result when correlating *Are* back to the pure Dialog chatterbot results of Tables 5.9 and 5.10, where *Are*'s Response Satisfaction was 4.7077 for AZ-ALICE-Dialog and 4.9388 for TARA-Dialog, compared to AZ-ALICE-Both (Dialog) of 4.6026 and TARA-Both (Dialog) of 5.1481. In each case, the Dialog-based *Are* interrogative has the highest Response Satisfaction rating (p-value < 0.01).

The other important finding is the low Response Satisfaction of *Why* and *What*. These interrogatives generally scored very low across all six chatterbots. It is believed that this is the result of chatterbot vagueness when a specific answer was desired, as shown in the examples below.

User: What is a terrorist attack?
System: I think you already know the answer.

User: Why is he considered a terrorist?
System: Perhaps his creative genius perhaps there is a another reason.

5.5.3 Response satisfaction scores show a weak relation to query length

We further investigated the role that query length may have on response satisfaction scores. Applying the search engine research of Zuckerman, we had hypothesized that our chatterbot response satisfaction scores will show a similar inverse relationship to query length. Table 5.13 illustrates the query lengths observed for each chatterbot.

	# of Inputs	Query Length	
		Average	Std Dev
AZ-ALICE - Dialog	3,906	5.6521	2.9233
AZ-ALICE - Domain	5,906	5.5320	2.8195
AZ-ALICE - Both	4,284	5.3987	2.8160
TARA - Dialog	931	6.6584	3.0042
TARA - Domain	487	6.7351	3.3225
TARA - Both	731	7.0766	3.7453

Table 5.13. Query lengths across chatterbots

Tables 5.14 and 5.15 show a break-down of the average number of query words for each of the eight most frequently observed chatterbot interrogatives.

ALICE Studies - Telecommunications Domain					
Dialog		Domain		Both	
Interrogative	Word length	Interrogative	Word length	Interrogative	Word length
Why	5.8415	Is	6.1746	Why	5.9135
Is	5.6735	How	5.9821	How	5.8817
Do	5.5748	Do	5.9779	Is	5.7288
What	5.5360	Why	5.8225	Do	5.6782
How	5.2366	Are	5.1910	Are	5.0886
Where	5.1392	What	5.1408	What	4.9374
Are	5.0154	Where	4.9346	Where	4.7606
Who	4.5000	Who	4.0766	Who	4.4309

Table 5.14. Query lengths for each interrogative in Telecommunications

TARA Studies - Terrorism Domain					
Dialog		Domain		Both	
Interrogative	Word length	Interrogative	Word length	Interrogative	Word length
Do	7.6875	Do	8.9688	Do	8.5094
Why	6.8750	Why	8.8182	How	7.5088
Is	6.6800	Is	7.8333	What	6.3977
How	6.6140	How	7.5909	Is	6.3333
What	6.4868	What	6.8750	Are	5.9032
Are	6.2449	Are	6.5714	Who	5.0597
Where	5.5429	Where	5.2286	Why	5.0000
Who	5.2105	Who	4.9254	Where	4.3421

Table 5.15. Query lengths for each interrogative in Terrorism

From these tables, *Why* questions appear to have the longest average query length. *Who* interrogatives are generally the shortest and questions in Terrorism are generally longer than their counterparts in Telecommunications.

Aggregating all interrogative-based response satisfaction scores on a per chatterbot level and analyzing their relation to query length, yields Table 5.16.

	Regressed Slope of Reponse Satisfaction scores to Query Length	p-value
AZ-ALICE - Dialog	-0.0231	0.0050
AZ-ALICE - Domain	0.0173	0.0276
AZ-ALICE - Both	-0.0127	0.1501
TARA - Dialog	-0.0790	0.0003
TARA - Domain	-0.1786	0.0000
TARA - Both	-0.0830	0.0002

Table 5.16. Relating query lengths to response satisfaction scores

From Table 5.16, we found that five of the six chatterbots exhibited a negative relation between query length and response satisfaction, which is consistent with our expectations. We did not expect that any of the chatterbots would show a positive relation, but AZ-ALICE-Domain did. However, when applying a standard statistical

significance of p -value < 0.001 , we found that all three AZ-ALICE chatterbots had no significant slope, while the three TARA chatterbots did.

One possible explanation for these findings may come from the disparate levels of knowledge used between the two studies. It may be that AZ-ALICE did not possess a sufficient critical mass of knowledge which resulted in poorer ratings regardless of query length. This conclusion of insufficient critical mass is best supported by the observations of AZ-ALICE-Domain which possessed the least amount of knowledge. This unusual chatterbot exhibited a positive relationship between query length and response satisfaction at p -value < 0.05 , where longer queries were rated more satisfying. The effect of longer query lengths in a knowledge-starved system would lead to a better chance of the system picking up some keyword and returning a response.

From our interpretation, the TARA chatterbots confirm the observations of Zuckerman that accuracy is inversely proportional to query length, while the AZ-ALICE chatterbots would appear inconclusive.

5.6 Conclusions

We believe that the findings generated by this chapter have a direct applicability to existing ALICE chatterbots and can help shed light on the design of natural language dialog systems.

In particular, we found that interrogatives are a major source of user inquiries (Voorhees, 2001). *Wh** interrogatives and *What* especially, appear to be a good place to focus future knowledge-gathering activities. In both the AZ-ALICE and TARA studies,

What is * was found to be the most prevalent question type. Collecting more *What is* * types of knowledge categories should help improve dialog systems by providing more specific and more likely to be triggered responses.

We also found that interrogatives beginning with *Are* and *Where* were the most satisfying. We believe this is because general and vague chatterbot responses fit these types of questions best because of their binary, non sequitur or clichéd nature. *Why* and similar interrogatives that expected a specific answer were found the least satisfying when a general response was given. To capitalize on this finding we would further suggest more knowledge categories with an emphasis on specific responses.

Query length would appear to have an impact on response satisfaction, where longer queries led to decreasing response satisfaction scores. The TARA studies meted this result out, while the AZ-ALICE studies were inconclusive. We believe that this inconsistency comes from an insufficient critical mass of knowledge used in the AZ-ALICE chatterbots. Future work should evaluate just how many AIML patterns are needed for minimal conversational saliency.

CHAPTER 6: MAKING PREDICTIONS FROM TEXTUAL KNOWLEDGE

This chapter explores the implications of making discrete numeric predictions based on textual knowledge. In particular, we explore the realm of stock market prediction from textual financial news articles and show that the techniques presented have value within the business domain.

6.1 Background

Stock Market prediction has always had a certain appeal for researchers. While numerous scientific attempts have been made, no method has been discovered to accurately predict stock price movement. The difficulty of prediction lies in the complexities of modeling market dynamics. Even with a lack of consistent prediction methods, there have been some mild successes.

Stock Market research encapsulates two elemental trading philosophies; Fundamental and Technical approaches (Technical Analysis, 2005). In Fundamental analysis, Stock Market price movements are believed to derive from a security's relative data. Fundamentalists use numeric information such as earnings, ratios and management effectiveness to determine future forecasts. In Technical analysis, it is believed that market timing is key. Technicians utilize charts and modeling techniques to identify trends in price and volume. These latter individuals rely on historical data in order to predict future outcomes.

One area of limited success in Stock Market prediction comes from textual data. Information from quarterly reports or breaking news stories can dramatically affect the

share price of a security. Most existing literature on financial text mining relies on identifying a predefined set of keywords and machine learning techniques. These methods typically assign weights to keywords in proportion to the movement of a share price. These types of analyses have shown a definite, but weak ability to forecast the direction of share prices.

In this chapter we experiment using several linguistic textual representations, including Bag of Words, Noun Phrases and Named Entities approaches. We believe that combining more precise textual representations with past stock pricing information will yield improved predictability results.

This chapter is arranged as follows. Section 6.2 provides an overview of literature concerning Stock Market prediction, textual representations and machine learning techniques. Section 6.3 describes our research questions and system design. Section 6.4 provides an overview of our experimental design. Section 6.5 expresses our experimental findings and discusses their implications. Section 6.6 delivers our experimental conclusions with a brief oratory on future directions for this stream of research.

6.2 Related Works

When predicting the future prices of Stock Market securities, there are several theories available. The first is Efficient Market Hypothesis (EMH) (Fama, 1964). In EMH, it is assumed that the price of a security reflects all of the information available and that everyone has some degree of access to the information. Fama's theory further breaks EMH into three forms: Weak, Semi-Strong and Strong. In Weak EMH, only

historical information is embedded in the current price. The Semi-Strong form goes a step further by incorporating all historical and currently public information in the price. The Strong form includes historical, public and private information, such as insider information, in the share price. From the tenets of EMH, it is believed that the market reacts instantaneously to any given news and that it is impossible to consistently outperform the market.

A different perspective on prediction comes from Random Walk Theory (Malkiel, 1973). In this theory, Stock Market prediction is believed to be impossible where prices are determined randomly and outperforming the market is infeasible. Random Walk Theory has similar theoretical underpinnings to Semi-Strong EMH where all public information is assumed to be available to everyone. However, Random Walk Theory declares that even with such information, future prediction is ineffective.

It is from these theories that two distinct trading philosophies emerged; the fundamentalists and the technicians. In a fundamentalist trading philosophy, the price of a security can be determined through the nuts and bolts of financial numbers. These numbers are derived from the overall economy, the particular industry's sector or most typically, from the company itself. Figures such as inflation, joblessness, return on equity (ROE), debt levels and individual Price to Earnings (PE) ratios can all play a part in determining the price of a stock.

In contrast, technical analysis depends on historical and time-series data. These strategists believe that market timing is critical and opportunities can be found through the careful averaging of historical price and volume movements and comparing them

against current prices. Technicians also believe that there are certain high/low psychological price barriers such as support and resistance levels where opportunities may exist. They further reason that price movements are not totally random, however, technical analysis is considered to be more of an art form rather than a science and is subject to interpretation.

Both fundamentalists and technicians have developed certain techniques to predict prices from financial news articles. In one model that tested trading philosophies; LeBaron et. al. posited that much can be learned from a simulated stock market with simulated traders (LeBaron, Arthur et al., 1999). In their work, simulated traders mimicked human trading activity. Because of their artificial nature, the decisions made by these simulated traders can be dissected to identify key nuggets of information that would otherwise be difficult to obtain. The simulated traders were programmed to follow a rule hierarchy when responding to changes in the market; in this case it was the introduction of relevant news articles and/or numeric data updates. Each simulated trader was then varied on the timing between the point of receiving the information and reacting to it. The results were startling and found that the length of reaction time dictated a preference of trading philosophy. Simulated traders that acted quickly formed technical strategies, while traders that possessed a longer waiting period formed fundamental strategies (LeBaron, Arthur et al., 1999). It is believed that the technicians capitalized on the time lag by acting on information before the rest of the traders, which lent this research to support a weak ability to forecast the market for a brief period of time.

In similar research on real stock data and financial news articles, Gidofalvi gathered over 5,000 financial news articles concerning 12 stocks and identified this brief duration of time to be a period of twenty minutes before and twenty minutes after a financial news article was released (Gidofalvi, 2001). Within this period of time, Gidofalvi demonstrated that there exists a weak ability to predict the direction of a security before the market corrects itself to equilibrium. One reason for the weak ability to forecast is because financial news articles are typically reprinted throughout the various news wire services. Gidofalvi posits that a stronger predictive ability may exist in isolating the first release of an article. Using this twenty minute window of opportunity and an automated textual news parsing system, the possibility exists to capitalize on stock price movements before human traders can act.

6.2.1 Textual Representation

There are a variety of methods available to analyze financial news articles. One of the simplest methods is to tokenize and use each word in the document. While this human friendly approach may help users to understand the syntactic structure of the document, machine learning techniques do not require such structural markings. This technique also assigns importance to determiners and prepositions which have little contribution to the overall meaning of the article. One method of circumventing these problems is to use a Bag of Words approach. In this approach, a list of semantically empty stop-words are removed from the article (e.g.; the, a and for). The remaining terms are then used as the textual representation. The Bag of Words approach has been

used as the de facto standard of financial article research primarily because of its simple nature and its ability to produce a suitable representation of the text.

Building upon the Bag of Words approach, another tactic is to use a subset of terms as features (Moldovan, Pasca et al., 2003), which can address issues related to article scaling while still encompassing the important concepts of an article (Tolle and Chen, 2000). One such method using this approach is Noun Phrasing. Noun Phrasing is accomplished through the use of a syntax where parts of speech (i.e., nouns) are identified through the aid of a lexicon and aggregated using syntactic rules on the surrounding parts of speech, forming noun phrases.

A third method of article representation is Named Entities. This technique builds upon Noun Phrases by using semantic/syntactic tagging where nouns and noun phrases can be classified under predetermined categories (Sekine and Nobata, 2004). Using the MUC-7 framework of entity classification, these categories are date, location, money, organization, percentage, person and time. The semantic/syntactic procedure operates by first assigning syntactic tags to the terms and then assigning semantic tags from a lexical hierarchy with multiple inheritance (McDonald, Chen et al., 2005). Because of the constrained categories, Named Entities in effect provide for a more abstract representation than Bag of Words or Noun Phrases.

Both Noun Phrases and Named Entities have shown limited success through previous comparison trials of tagging accuracy between differing algorithms. However, their usage as wide-scale textual representations for machine learning purposes remains somewhat unknown.

6.2.2 Machine Learning Algorithms

Like textual representation, there are also a variety of machine learning algorithms available. Almost all techniques start off with a technical analysis of historical security data by selecting a recent period of time and performing linear regression analysis to determine the price trend of the security. From there, a Bag of Words analysis is used to determine the textual keywords. Some keywords such as ‘earnings’ or ‘loss’ can lead to predictable outcomes which are then classified into stock movement prediction classes such as up, down, and unchanged. Much research has been done to investigate the various techniques that can lead to stock price classification. Table 6.1 illustrates a Stock Market prediction taxonomy of the various machine learning techniques.

Algorithm	Classification	Source Material	Examples
Genetic Algorithm	2 categories	Undisclosed number of chatroom postings	Thomas & Sycara, 2002
Naïve Bayesian	3 categories	Over 5,000 articles borrowed from Lavrenko	Gidofalvi et al. 2001
	5 categories	38,469 articles	Lavrenko et al. 2000
	5 categories	6,239 articles	Seo et al. 2002
SVM	3 categories	About 350,000 articles	Fung et al. 2002
	3 categories	6,602 articles	Mittermayer, 2004

Table 6.1. Taxonomy of prior algorithmic research

From Table 6.1, several items become readily noticeable. The first of which is that a variety of algorithms have been used. The second is that almost all instances commonly classify predicted stock movements into a set of classification categories, not a discrete price prediction. Lastly, not all of the studies were conducted on financial news articles, although a majority was.

The first technique of interest is the Genetic Algorithm. In this study, discussion boards were used as a source of independently generated financial news (Thomas and Sycara, 2002). In their approach, Thomas and Sycara attempted to classify stock prices

using the number of postings and number of words posted about an article on a daily basis. It was found that positive share price movement was correlated to stocks with more than 10,000 posts. However, discussion board postings are quite susceptible to bias and noise.

Another machine learning technique, Naïve Bayesian, represents each article as a weighted vector of keywords (Seo, Giampapa et al., 2002). Phrase co-occurrence and price directionality is learned from the articles which lead to a trained classification system. One such problem with this style of machine learning is from a company mentioned in passing. An article may focus its attention on some other event and superficially reference a particular security. These types of problems can cloud the results of training by unintentionally attaching weight to a casually-mentioned security.

One of the more interesting machine learners is Support Vector Machines (SVM). In the work of Fung et. al., regression analysis of technical data is used to identify price trends while SVM analysis of textual news articles is used to perform a binary classification in two predefined categories; stock price rise and drop (Fung, Yu et al., 2002). In cases where conflicting SVM classification ensues, such as both rise and drop classifiers are determined to be positive, the system returns a 'no recommendation' decision. From their research using 350,000 financial news articles and a simulated Buy-Hold strategy based upon their SVM classifications, they showed that their technique of SVM classification was mildly profitable.

Mittermayer also used SVM in his research to find an optimal profit trading engine (Mittermayer, 2004). While relying on a three tier classification system, this research

focused on empirically establishing trading limits. It was found that profits can be maximized by buying or shorting stocks and taking profit on them at 1% up movement or 3% down movement. This method slightly beat random trading by yielding a 0.11% average return.

Many of the prior studies were classification oriented with questions asked such as; *will this article cause the stock price to increase/decrease?* These studies were all tests of directional movement and not the predictors of stock prices. Discrete prediction from numeric trends is hardly new. However, the application of this regression technique to SVM mechanics is rather recent (Gao, Gunn et al., 2002). One such method is Sequential Minimal Optimization (SMO), where many of the scalability problems from using large training sets has been obviated through a more simplistic SVM solving technique (Platt, 1999). This combination of techniques has lead to numeric prediction studies for futures contracts (Tay and Cao, 2001), but discrete prediction has not been coupled with a systematic study of various textual analysis methods before.

From prior studies on the textual representation of documents, Joachims posits that limiting the inclusion of features to three or more instances per document will avoid the problem of unmanageably large feature spaces (Joachims, 1998). Extending this to textual representation, each feature is further represented in binary as either a zero or one; the term is either present or not present in the article (Vanschoenwinkel, 2003). This simple representational scheme is easy to implement and will lead to a sparse dataset with many zero features.

Applying these regression based methods and textual representation techniques to a supervised machine learning algorithm such as SVM can lead to a trained system with discrete numeric output.

Evaluation of output has been generally focused on only one of the following three metrics; measures of Closeness, Directional Accuracy or Simulated Trading. In measures of Closeness, the estimated value from machine learning is compared against the actual value in a Mean Squared Error (MSE) measure (Pai and Lin, 2005). Directional Accuracy was the more common measure of previous financial studies, where the direction of the predicted value is compared with the movement direction of the actual value (Cho, Wuthrich et al., 1999). Whereas Simulated Trading initiates a simple trading engine to capitalize on large predicted value differences (Lavrenko, Schmill et al., 2000a).

6.2.3 Financial News Article Sources

In real-world trading applications, the amount of textual data available to stock market traders is staggering. This data can come in the form of required shareholder reports, government-mandated forms or news articles concerning a company's outlook. Articles and reports are also routinely cross-posted in many different locations leading to problems of uniqueness and database selection (Conrad and Claussen, 2003). Reports of an unexpected nature can lead to wildly significant changes in the price of a security. Table 6.2 illustrates a taxonomy of textual financial data.

Textual Source	Types	Examples	Description
Company Generated Sources	SEC Reports	8K 10K	Reports on significant changes Annual reports
	Analyst Created	Recommendations Stock Alerts	Buy/Hold/Sell assessments Alerts for share prices
Independently Generated Sources	News Outlets	Financial Times Wall Street Journal	Financial News stories Financial News stories
	News Wire Services	PRNewsWire Yahoo Finance	Breaking financial news articles 45 financial news wire sources
	Discussion Boards	The Motley Fool	Forum to share stock-related information

Table 6.2. Taxonomy of textual financial data

Textual data can arise from two sources; company generated and independently generated sources. Company generated sources such as quarterly and annual reports can provide a rich linguistic structure that if properly read can indicate how the company will perform in the future (Kloptchenko, Eklund et al., 2004). This textual wealth of information may not be explicitly shown by financial ratios but rather encapsulated in forward-looking statements or other textual locations. Independent sources such as analyst recommendations, news outlets, and wire services can provide a more balanced view of the company and have a lesser potential to bias news reports. Discussion boards can also provide independently generated financial news; however, they can be suspect sources.

News outlets can be differentiated from wire services in several different ways. One of the main differences is that news outlets are centers that publish available financial information at specific time intervals. Examples include Bloomberg, Business Wire, CNN Financial News, Dow Jones, Financial Times, Forbes, Reuters and the Wall Street Journal (Cho, 1999; Seo, Giampapa et al., 2002). In contrast, news wire services publish

available financial information as soon as it is publicly released or discovered. News wire examples include PRNewsWire, which has free and subscription levels for real-time financial news access and Yahoo Finance, which is a compilation of 45 news wire services including the Associated Press and PRNewsWire. Besides their relevant and timely release of financial news articles, news wire articles are also easy to automatically gather and are an excellent source for computer-based algorithms.

Stock Quotations are also an important source of financial information. Quotes can be divided into various increments of time from minutes to days, however, one minute increments provide sufficient granularity for machine learning.

While previous studies have mainly focused on the classification of stock price trends, none has been discovered to harness machine learning to determine a discrete stock price prediction based on breaking news articles. Prior techniques have relied solely on a Bag of Words approach and not other textual representations. Finally, there is no consensus on what information to include in a model that will lead to better performance. From these gaps in the research we form the crux of our study with the following questions.

6.3 Proposed Approaches

Given that prior research in textual financial prediction has focused solely on the classification of stock price direction, we ask whether the prediction of discrete values is possible. This leads to our first research question.

- How effective is the prediction of discrete stock price values using textual financial news articles?

We expect to find that discrete prediction from textual financial news articles is possible. Since prior research has indicated that certain keywords can have a direct impact on the movement of stock prices, we believe that predicting the magnitude of these movements is likely.

Prior research into stock price classification has almost exclusively relied on a Bag of Words approach. While this de facto standard has led to promising results, we feel that other textual representation schemes may provide better predictive ability, leading us to our second research question.

- Which combination of textual analysis techniques is most valuable in stock price prediction?

Since prior research has not examined this question before, we are cautious in answering such an exploratory issue. However, we feel that other textual representation schemes may serve to better distill the article into its essential components.

From these questions we developed the AZFinText system illustrated in Figure 6.1.

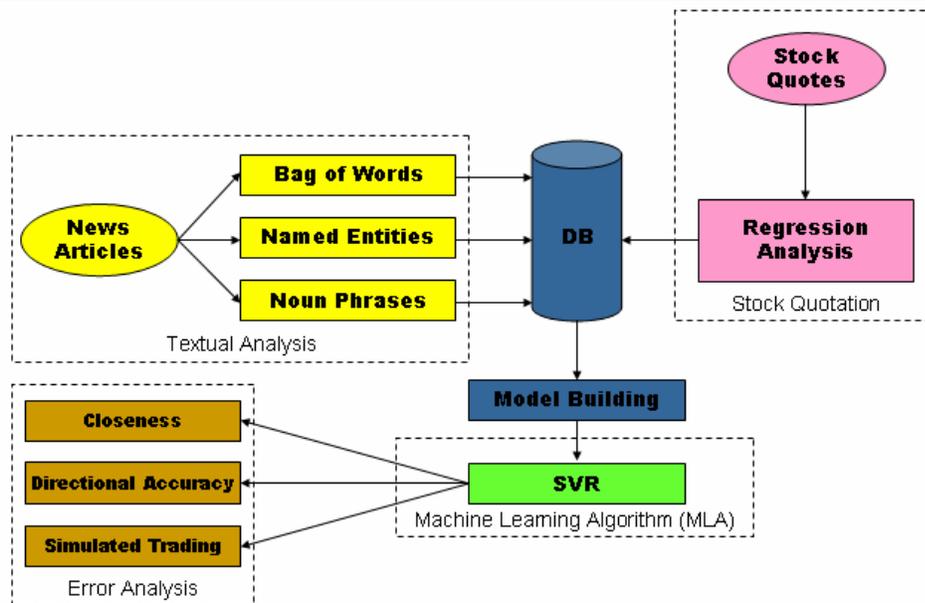


Figure 6.1. AZFinText system design

In this design, each financial news article is represented using three textual analysis techniques; Bag of Words, Noun Phrases and Named Entities. These representations identify the important article terms and store them in the database. To limit the size of the feature space, we selected terms that occurred three or more times in a document (Joachims, 1998).

To perform our textual analysis we chose a modified version of the Arizona Text Extractor (AzTeK) system which performs semantic/syntactic word level tagging as well as phrasal aggregation. AzTeK's Noun Phrasing component works by using a syntactic tagger to identify and aggregate the document's noun phrases and was found to have an 85% F-measure for both precision and recall, which is comparable to other tools (Tolle and Chen, 2000). The Entity extractor portion goes one step further by assigning hybrid semantic/syntactic tags to document terms and phrases in one of the seven predefined

categories of date, location, money, organization, percentage, person and time (McDonald, Chen et al., 2005). These entities are then identified through the usage of a lexicon. Although the AzTeK system was selected due to availability, it performs adequately for noun phrase and named entity extraction. However, there are many other such systems as reported in the Message Understanding Conference (McDonald, Chen et al., 2005), that can be adopted for financial news text analysis.

Stock Quotes are gathered on a per minute basis for each stock. When a news article is released, we estimate what the stock price would be 20 minutes after the article was released. To do this we perform linear regression on the quotation data using an arbitrary 60 minutes prior to article release and extrapolate what the stock price should be 20 minutes in the future.

To test the types of information that need to be included, we developed four different models and varied the data given to them. The first model, Regress, was a simple linear regression estimate of the +20 minute stock price. Assuming that breaking financial news articles have no impact on the movement of stock prices, we would expect reasonable performance from this model. While we acknowledge the obvious violation of Random Walk Theory, within such a compressed amount of time weak predictive ability remains (Gidofalvi, 2001). The next three models use the supervised learning of SVM regression to compute their +20 minute predictions. Model M1, uses only extracted article terms for its prediction. While no baseline stock price exists within this model, we chose it because of its frequent usage in prior studies on directional classification of stock prices. Model M2, uses extracted article terms and the stock price

at the time the article was released. We feel that given a baseline of stock price that this model will fare better. Model M3, uses extracted terms and a regressed estimate of the +20 minute stock price. This model may lead to better predictive results should the article terms have no impact on the movement of the stock price.

To illustrate how the AZFinText system works, we offer a sample news article (Burns and Wutkowski, 2005) and step through the logic of our system.

Schwab shares fell as much as 5.3 percent in morning trading on the New York Stock Exchange but later recouped some of the loss. San Francisco-based Schwab expects fourth-quarter profit of about 14 cents per share two cents below what it reported for the third quarter citing the impact of fee waivers a new national advertising campaign and severance charges. Analysts polled by Reuters Estimates on average had forecast profit of 16 cents per share for the fourth quarter. In September Schwab said it would drop account service fees and order handling charges its seventh price cut since May 2004. Chris Dodds the company's chief financial officer in a statement said the fee waivers and ad campaign will reduce fourth-quarter pre-tax profit by \$40 million while severance charges at Schwab's U.S. Trust unit for wealthy clients will cut profit by \$10 million. The NYSE fined Schwab for not adequately protecting clients from investment advisers who misappropriated assets using such methods as the forging of checks and authorization letters. The improper activity took place from 1998 through the first quarter of 2003 the NYSE said. This case is a stern reminder that firms must have adequate procedures to supervise and control transfers of assets from customer accounts said Susan Merrill the Big Board's enforcement chief. It goes to the heart of customers expectations that their money is safe. Schwab also agreed to hire an outside consultant to review policies and procedures for the disbursement of customer assets and detection of possible misappropriations the NYSE said. Company spokeswoman Alison Wertheim said neither Schwab nor its employees were involved in the wrongdoing which she said was largely the fault of one party. She said Schwab has implemented a state-of-the-art surveillance system and improved its controls to monitor independent investment advisers. According to the NYSE Schwab serves about 5 000 independent advisers who handle about 1.3 million accounts. Separately Schwab said October client daily average trades a closely watched indicator of customer activity rose 10 percent from September to 258 900 though total client assets fell 1 percent to \$1.152 trillion. Schwab shares fell 36 cents to \$15.64 in morning trading on the Big Board after earlier falling to \$15.16. (Additional reporting by Dan Burns and Karey Wutkowski)

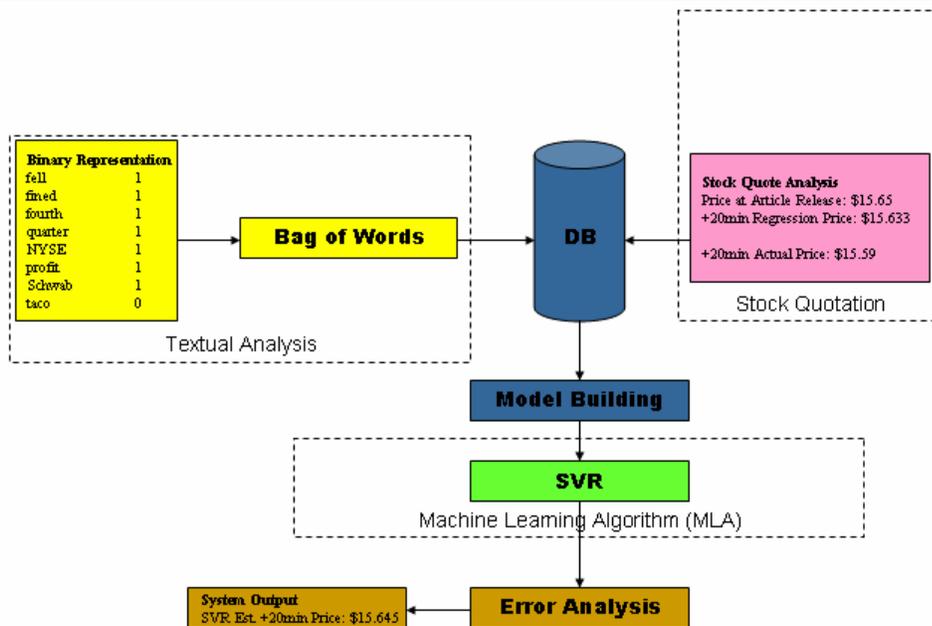


Figure 6.2. Example AZFinText representation

Figure 6.2 shows a sample run with Bag of Words using the previous article. The extracted terms are represented in binary as either present or not. Supposing our corpus also contained the term *Reuters* that appeared in a different article and not in this instantiation, the term is given a zero for not being present in the current article. For stock quotation data, we lookup what the stock price was at the time of article release (\$15.65), calculate a regression estimate of the +20 minute stock price (\$15.633) and lookup the actual +20 minute stock price for training and later evaluation (\$15.59). This data is then taken to the Model Building stage where the various models are given their appropriate data. Following that, machine learning takes place and an estimate of the +20 minute stock price is produced (\$15.645). We can see from the stock prices given in this

example that Schwab's share price dropped six cents while the model estimate figures a more conservative half penny drop.

6.4 Experimental Design

For our experiment we picked a research period of Oct. 26 to Nov. 28, 2005 to gather news articles and stock quotes. We further focused our attention only on companies listed in the S&P 500 as of Oct. 3, 2005. We acknowledge that several mergers and acquisitions did take place during this period of time; however, this only had an effect on less than 2% of the stocks tracked. In order to eliminate the 'company in passing' problem, we gathered the news articles from Yahoo Finance using a company's stock ticker symbol. This resulted in articles on 484 of the 500 companies listed in the S&P 500. Articles were further constrained to a time frame of one hour after the stock market opened to twenty minutes before the market closed. This period of time allows for sufficient data to be gathered for prior regression trend analysis and future estimation purposes. We further limited the influence of articles such that we did not use any articles that occurred within twenty minutes of each other. This measure eliminated several possible avenues of confounding results.

By performing these actions we gathered 9,211 candidate financial news articles and 10,259,042 stock quotes over the five-week period. From this pool of news articles we analyzed them using the three textual representations and retained only those terms that appeared three or more times in an article. The filtering process resulted in the following breakdown:

- Bag of Words used 4,296 terms from 2,839 articles
- Noun Phrases used 5,283 terms from 2,849 articles
- Named Entities used 2,856 terms from 2,620 articles

Article and stock quote data was then processed by a Support Vector Machine derivative, Sequential Minimal Optimization style of regression (Platt, 1999), which can handle discrete number analysis. We chose a linear kernel and performed 10-fold cross validation.

Following training, we chose three evaluation metrics; Closeness, Directional Accuracy and a Simulated Training Engine. The Closeness metric evaluated the difference between the predicted value and the actual stock price, measured using Mean Squared Error (MSE). Directional Accuracy measured the up/down direction of the predicted stock price compared with the actual direction of the stock price. While the inclusion of Directional Accuracy may not seem intuitive given the measure of Closeness, it is possible to be close in prediction yet predict the wrong direction of movement. This leads us to a third evaluation measure using a Simulated Trading Engine that invests \$1,000 per trade and follows simple trading rules. The rules implemented by our trading engine are a modified version of those proposed by Mittermayer to maximize short-term trading profit (Mittermayer, 2004). Our Simulated Trading Engine evaluates each news article and will buy/short the stock if the predicted +20 minute stock price is greater than or equal to 1% movement from the stock price at the time the article was released. Any bought/shorted stocks are then sold after 20 minutes. This assumes a zero transaction cost which is consistent with the research of Lavrenko (Lavrenko, Schmill et

al., 2000a; Lavrenko, Schmill et al., 2000b) and Mittermayer (Mittermayer, 2004) who argue that trading in volume will offset the costs of trading.

6.5 Experimental Findings and Discussion

In order to answer our research questions on the effectiveness of discrete stock prediction and the best textual representation; we tested our three models against a regression-based predictor using the three dimensions of analysis; measures of Closeness, Directional Accuracy and a Simulated Trading Engine. Table 6.3 shows the results of the Closeness measures, Table 6.4 illustrates Directional Accuracy and Table 6.5 displays the results of the Simulated Trading Engine.

MSE	Regress	M1	M2	M3
Bag of Words	0.07279	930.87	0.04422	0.12605
Noun Phrases	0.07279	863.50	0.04887	0.17944
Named Entities	0.07065	741.83	0.03407	0.07711
Average	0.07212	848.15	0.04261	0.12893

Table 6.3. Closeness results

Directional Accuracy	Regress	M1	M2	M3
Bag of Words	54.8%	52.4%	57.0%	57.0%
Noun Phrases	54.8%	56.4%	58.0%	56.9%
Named Entities	54.2%	55.0%	56.4%	56.7%
Totals	54.6%	54.6%	57.1%	56.9%

Table 6.4. Directional Accuracy results

Trading Engine	Regress	M1	M2	M3
Bag of Words	-1.81%	-0.34%	1.59%	0.98%
Noun Phrases	-1.81%	0.62%	2.57%	1.17%
Named Entities	-2.26%	-0.47%	2.02%	2.97%
Totals	-1.95%	-0.05%	2.06%	1.67%

Table 6.5. Simulated Trading Engine results

6.5.1 Model M2 performed the best

From looking at the average results in Table 6.3, Model M2 which used both article terms and the stock price at the time of article release, had the lowest MSE score (0.04261) of any of the models (p-values < 0.05). This result signifies that Model M2's predictions were closer to the actual +20 minute stock price than any of the other models including linear regression (Regress). Looking deeper into the results, we find that Model M2 performed better than Regress in each of the three textual representations, which supports Gidofalvi's claim of weak short-term predictability.

Model M1, which used only article terms, had a difficult time in its estimation of future stock prices with an average Closeness score of 848.15. While this model may have been appropriate for prior classification-only studies, this unpleasant value was somewhat expected given the lack of baseline stock prices.

The other item of interest is that Model M2's Named Entities representation had the best performance at 0.03407 (p-values < 0.05). We will further investigate the effects of textual representation in a later section.

Turning our attention back to Model M2, we examined the weighting scheme of SVM on the training variables. The stock price at the time of article release was given a weight of 0.9997, while the article terms had a combined weight of 0.0003. While the weighting of article terms may appear superficially light, these terms are important because they provide the final touches to the estimated +20 minute stock price. If we were to rely on price alone, we would have the values of Regress, and Model M2 performed better than Regress. If instead we used a regressed price estimate plus article terms, we would have

Model M3, and M2 performed better than M3. However, article terms alone were not sufficient in estimating the future stock price as demonstrated by Model M1.

Comparing our observed results with prior studies, Pai and Lin's study on forecasting stock prices across four different models had an MSE score of 0.3001 (Pai and Lin, 2005). In their study, they attempted stock price prediction one day in advance, using a small set of stocks and only close of day prices. Judging from these results, our findings were an order of magnitude better.

In evaluating the Directional Accuracy results of Table 6.4, we again note that Model M2 performed better on average (57.1%) than the other models (p-values < 0.05). Regress did not perform so well (54.6%), which would seem to indicate that unexpected stock swings were captured by article terms. Comparing our results to previous studies shows that our values are somewhat reasonable. Cho et. al. which used 100 days of training articles and 392 keywords, had an average directional accuracy of 46.8% (Cho, Wuthrich et al., 1999).

In the Simulated Trading results of Table 6.5, Model M2 using article terms and the stock price at the time of article release again had the best performance at 2.06% return (p-values < 0.05). This result would imply that Model M2 was better able to capitalize on trading opportunities given article terms and baseline stock price. Comparing this model against Model M1 with a trading return of -0.05%, we see that using article terms alone were insufficient. The results from Regress (-1.95% return) were unexpected. We believe that this finding was from the news articles themselves affecting major changes in the share price of stocks. Correlating our results with prior studies, Lavrenko et. al.

claimed a 2% return from tracking four stocks over a forty day period (Lavrenko, Schmill et al., 2000b). In a similar study, Lavrenko et. al. expanded the number of stocks to 127 over the same 40 day period and had a much lower return of 0.23% (Lavrenko, Schmill et al., 2000a). Both of these studies used essentially the same trading mechanism as we did which leads to an interesting observation; that perhaps more stocks lead to lower returns, although our study tracked 500 companies over 23 trading days. In a third simulated trading study, Mittermayer obtained a 0.11% average return using all of the stocks from Nasdaq, NYSE, and AMEX over a one year period of time. From our results, it would appear that our system is achieving fairly reasonable results at a 2.09% return.

Compiling all of the model performances together, Model M2 using article terms and the stock price at the time of article release performed best in all three metrics; measures of Closeness (0.04261), Directional Accuracy (57.1%) and Simulated Trading (2.06% return). This model was better able to capture stock price movements and further bolsters the idea of weak short-term predictability. Our results were also inline with those from prior studies and mostly performed better. With some tweaking to how we classify directional movement, we feel that our system could produce better Directional Accuracy results as well.

6.5.2 A superset of Named Entities was the best textual representation.

To answer our second research question, *which combination of textual analysis techniques is most valuable in stock price prediction*, we compare the averages of each textual representation using our three metrics. Table 6.6 presents the results of Closeness

measures, Table 6.7 displays Directional Accuracy and Table 6.8 illustrates the Simulated Trading Engine.

MSE	Regress	M1	M2	M3	Average
Bag of Words	0.07279	930.87	0.04422	0.12605	232.77789
Noun Phrases	0.07279	863.50	0.04887	0.17944	215.95020
Named Entities	0.07065	741.83	0.03407	0.07711	185.50404

Table 6.6. Closeness results

Directional Accuracy	Regress	M1	M2	M3	Totals
Bag of Words	54.8%	52.4%	57.0%	57.0%	55.3%
Noun Phrases	54.8%	56.4%	58.0%	56.9%	56.5%
Named Entities	54.2%	55.0%	56.4%	56.7%	55.6%

Table 6.7. Directional Accuracy results

Trading Engine	Regress	M1	M2	M3	Totals
Bag of Words	-1.81%	-0.34%	1.59%	0.98%	0.10%
Noun Phrases	-1.81%	0.62%	2.57%	1.17%	0.64%
Named Entities	-2.26%	-0.47%	2.02%	2.97%	0.57%

Table 6.8. Simulated Trading Engine results

From these tables, Named Entities had the lowest score in measures of Closeness (185.50404) and Noun Phrases had the better score in both Directional Accuracy (56.5%) and Simulated Trading (0.64%), all p-values < 0.05. These seemingly confusing results were not as clear-cut as our Model selection in the previous section as no one textual representation dominated the results.

However, it must be noted that these averaged results contain noise from previously failed models. If we were to focus only on the textual results for Model M2 and discard the other models, Noun Phrases performed the best in 2 of the 3 metrics and Named Entities in the remaining one.

These results ran contrary to our expectations. We had assumed that a Named Entity representation would generate better performance because of its ability to abstract the article terms and discard the noise of terms picked up by both Bag of Words and Noun Phrases. This MUC-7 textual representation was not sufficient to adequately model our article terms and lead us to ask the question, *what were the differences between Noun Phrases and Named Entities?* The answer was that Named Entities are essentially specialized Proper Nouns. The AzTeK system we used for part of speech tagging, identifies select terms in one of seven categories; date, location, money, organization, percentage, person and time (McDonald, Chen et al., 2005). Words in these categories are basically a subset of Noun Phrases. We believe that expanding the number of categories for Named Entities will lead to a better representational scheme.

In order to investigate this we took a subset of terms from Noun Phrases that were tagged as Proper Nouns and introduced a fourth, hybrid, textual representation of Proper Nouns. This selection of terms is a comparable superset of Named Entities but without the Entity categories. Proper Nouns captured 3,710 article terms from 2,809 articles compared to the 5,283 terms in 2,849 articles for Noun Phrases and 2,856 terms in 2,620 articles for Named Entities.

To give the reader an understanding of what types of terms would be captured as Proper Nouns and not Named Entities, we refer back to the sample news article immediately preceding Figure 6.2. It is important to remember that Named Entities are derived using a semantic lexicon of previous input. Therefore, terms such as NYSE, which do not appear as a Named Entity, will be depicted in a Proper Noun representation.

Restating the metrics in terms of Model M2 to clear up some of the noise from the other failed models, we introduce the following data. Table 6.9 shows measures of Closeness, Table 6.10, Directional Accuracy and Table 6.11, Simulated Trading.

MSE	M2
Bag of Words	0.04422
Noun Phrases	0.04887
Proper Nouns	0.04433
Named Entities	0.03407

Table 6.9. Closeness results

Directional Accuracy	M2
Bag of Words	57.0%
Noun Phrases	58.0%
Proper Nouns	58.2%
Named Entities	56.4%

Table 6.10. Directional Accuracy results

Trading Engine	M2
Bag of Words	1.59%
Noun Phrases	2.57%
Proper Nouns	2.84%
Named Entities	2.02%

Table 6.11. Simulated Trading Engine results

The first item of interest is that the Proper Nouns subset performed better than Noun Phrases in all three metrics; 0.04433 to 0.04887 in measures of Closeness, 58.2% to 58.0% in Directional Accuracy and 2.84% to 2.57% in Simulated Trading (all p-values < 0.05). This would seem to back up our initial expectation that a more abstract textual representation would perform better. In comparison to Named Entities, Proper Nouns performed better in 2 of the 3 metrics, Directional Accuracy and Simulated Trading

whereas Named Entities had better success at measures of Closeness. This would indicate that the direction we have undertaken is perhaps correct, but is still in need of refinement. We would suggest that future research should evaluate expanding the number of entity categories and evaluating the optimal mix for business-related news articles.

Overall, Bag of Words performed poorly by comparison. While this textual representation may be the de facto standard used in other studies, its weak performance is believed to arise from its reliance on too many noisy article terms. Noun Phrases performed much better with good performance in both Directional Accuracy and Simulated Trading. However, it suffered from poor Closeness measures. We believe that this is the result of using a better tuned representational scheme of news articles, as compared to the Bag of Words approach. Yet Noun Phrases still possessed some elements of noise which led to less than desirable Closeness scores. Named Entities had some problems and did not perform as expected. While this representation had the best Closeness score in prediction accuracy, it was unable to translate those gains into both Directional Accuracy and Simulated Trading returns. This is probably the result of using a limited set of Entity categories which was unable to fully represent the content of financial news articles. Finally, Proper Nouns had the better performance results. While this textual representation can be thought of as the hybrid go-between for Noun Phrases and Named Entities, it had a solid performance on both Directional Accuracy and Simulated Trading. This result is likely attributable to Proper Nouns adequately using the

article terms in a manner that was freer of the noise plaguing Noun Phrases and free of the constraining categories used by Named Entities.

6.6 Conclusions

Our first conclusion was that Model M2, using both article terms and the stock price at the time of article release, had a dominating performance in all three metrics; measures of Closeness at 0.04261, Directional Accuracy at 57.1% and Simulated Trading at a 2.06% return. These results were the direct consequence of this model's ability to capitalize on the article terms and stock price for machine learning.

Our second conclusion was that Proper Nouns had the better textual representation performance. While it performed best in 2 of the 3 metrics, Directional Accuracy at 58.2% and Simulated Trading at 2.84%, it pulled up short on measures of Closeness, 0.04433, as compared to Named Entities with 0.03407, all p-values < 0.05. However, this subset representation performed better than its parent, Noun Phrases, in all three metrics. We believe that Proper Nouns can attribute its success to being freer of the term noise used by Noun Phrases and free of the constraining categories used by Named Entities. Although more research into what constitutes an optimum mix of entity categories is encouraged.

Future research includes using other machine learning techniques such as Relevance Vector Regression, which promises to have better accuracy and fewer vectors in classification (Bishop and Tipping, 2003). It would also be worthwhile to pursue expanding the selection of stocks outside of the S&P 500. While the S&P 500 is a fairly

stable set of companies, perhaps more volatile and less tracked companies may provide interesting results. Another worthwhile approach would be to test a model based on article terms and percentage of stock price change. While our models relied on fixed stock prices that traded within a consistent range, penny stocks with wild fluctuations may prove worthy of further research. Lastly, while we trained our system on the entire S&P 500, it would be a good idea to try more selective article training such as industry groups or company peer group training and examine those results in terms of prediction accuracy.

CHAPTER 7: REFINING PREDICTIONS FROM TEXTUAL KNOWLEDGE

This chapter examines the problem of discrete stock price prediction using a synthesis of linguistic, financial and statistical techniques to create the Arizona Financial Text System, AZFinText. We approach this line of research using textual representation and statistical machine learning methods on financial news articles that are partitioned by similar industry and sector groupings. Through our research, we discovered that stocks partitioned by Sectors were most predictable in measures of Closeness, Mean Squared Error (MSE) score of 0.1954, predicted Directional Accuracy of 71.18% and a Simulated Trading return of 8.50% (compared to 5.62% for the S&P 500 index). In direct comparisons to existing market experts and quantitative mutual funds, our system's trading return of 8.50% outperformed well-known trading experts. Our system also performed well against the top 10 quantitative mutual funds of 2005, where our system would have placed fifth. When comparing AZFinText against only those quantitative funds that monitor the same securities, AZFinText had a 2% higher return than the next best performing quant fund.

7.1 Background

Predicting changes in the stock market has always had a certain appeal to researchers. While numerous attempts have been made, the difficulty has always centered on the behaviors of human traders within this socially constructed system. With parameters ill-defined and constantly shifting, prediction has been difficult at best. To further create

confusion, there have been two diametrically opposed philosophies of stock market research; fundamental and technical analysis techniques (Technical Analysis, 2005).

Acquiring relevant textual data is an important facet of stock market prediction. While many reports and articles are written on a daily basis, information flowing from these sources must take the form of numeric data before existing applications can process it. This limitation forces open a temporal gap between when information is acquired to when it can be acted on. Information of an unexpected nature can cause a significant impact on share prices and the ability to harness these textual documents to make accurate predictions would be a useful decision-making tool.

While there have been several studies covering textual financial predictions, these studies have been limited in nature and constrained mostly to classifying price direction, i.e.; the stock price will increase, decrease or stay the same. Discrete analysis, or assigning an exact price with some degree of accuracy has not been a trivial task. In regards to textual document selection, prior studies have used either all of the available financial news articles for prediction or a subset of company-specific articles. These studies have neglected the investigation of in-between categories, such as market sectors and industries in their analyses.

This chapter is arranged as follows. Section 7.2 provides an overview of literature concerning Stock Market prediction, textual representations and machine learning techniques. Section 7.3 describes our research questions and proposed system. Section 7.4 provides an overview of our experimental design. Section 7.5 details our

experimental findings and discusses their impact on stock market prediction. Section 7.6 delivers our conclusions and a brief discourse on future research directions.

7.2 Related Works

When it comes to predicting stock market movement, two theories have had a significant impact on market research; Efficient Market Hypothesis (EMH) and Random Walk Theory. In EMH, the price of a security is a reflection of complete market information. Whenever a change in financial outlook occurs, the market will instantly adjust the security price to reflect the new information (Fama, 1964). EMH contained three different levels of information sharing; the weak form, the semi-strong and the strong form. Within weak EMH, only historical data is embedded within the current price. The semi-strong form goes a bit farther by incorporating historical and current public information into its prices. The strong form includes historical and current public information as well as private information. From these three forms, it was believed that markets behaved efficiently and instantaneous price corrections would obviate the desire for prediction models.

Random Walk Theory is slightly different in its theoretical underpinnings and focuses on an overall short-term random pattern of stock market movements (Malkiel, 1973). This random activity is believed to produce unpredictable prices and makes it impossible to consistently outperform the market. This view is similar to the Semi-Strong EMH model where all information is contained within the current price and is worthless for future prediction.

7.2.1 Fundamentalists and Technicians

While EMH and Random Walk have served to discourage forecasting activity, the traders that persisted formed two distinctly different viewpoints on market prediction; fundamental and technical analysis. Fundamentalists are interested in the nuts and bolts of a security, such as numeric data on the overall economy, individual stock health ratios such as inflation, interest rates, return on assets, debt to equity and price to earnings among others. The focus of the fundamental trader is to make predictions from the current set of numeric data. As a consequence, historical or time-series data is not considered. By contrast, technicians rely heavily on time-series data and believe that market timing is crucial. Figures such as volume, volatility, support/resistance levels and charting techniques are all within the repertoires of a Technicians toolbox. However, technical analysis is considered to be more of an art form and is subject to interpretation.

In a study comparing the merits of fundamental and technical trading strategies, LeBaron created an artificial stock market with simulated traders (LeBaron, Arthur et al., 1999). He introduced new pieces of information into the market and varied the amount of time between when an individual trader would receive information and act upon it. It was found that traders with longer-period waiting times formed fundamental strategies while those with shorter-period waits developed technical strategies. This study was more important from the standpoint that a lag was discovered between the time that information was introduced to when the market corrected itself. This apparent delay in market behavior helped to dispel the instantaneous correction notions of EMH and lent support to a weak ability to forecast the market. Subsequent research into this weak

forecasting ability led to the discovery of a twenty minute window of opportunity before and after a financial news article is released (Gidofalvi, 2001). Within this window, weak prediction of stock price direction was found to be possible.

7.2.2 Financial News Articles

The extent of prediction between financial news articles and their impact on stock market prices is a complex avenue to investigate. While the information contained in financial news articles can have a visible impact on a security's price (Wuthrich, Cho et al., 1998; Lavrenko, Schmill et al., 2000b; Gidofalvi, 2001; Mittermayer, 2004), sudden price movements can still occur from large unexpected trades (Camerer and Weigelt, 1991).

The first challenge of textual financial prediction is to process the large amounts of textual information that exist for securities. This material not only includes required reports such as periodic SEC filings, but also a wealth of financial news articles reporting unexpected events and routine news alike. Financial news articles can be automatically capitalized on by using Natural Language Processing (NLP) and text-processing techniques to identify specific terms that can lead to dramatic share price changes. This method can be repeatedly used to forecast price fluctuations and take advantage of arbitrage opportunities faster than human counterparts.

The means of obtaining timely financial news articles can come from a variety of Internet sources. One of which is Comtex which offers real-time financial news in a subscription format. Another source is PRNewsWire which offers free real-time and

subscription-based services. By contrast, Yahoo Finance is a compilation of 45 different news sources including the Associated Press, Financial Times and PRNewsWire among others. This source provides a variety of perspectives and timely news stories regarding financial markets.

7.2.3 Textual Representation

Once news articles have been collected they must be represented. One technique is to use a Bag of Words approach which has been extensively used in textual financial research (Lavrenko, Schmill et al., 2000b; Gidofalvi, 2001). This process involves the removal of meaningless stopwords such as conjunctions and declaratives from text and using the remaining terms as the textual representation. While this method has been popular, its drawbacks include noise from seldom-used terms and scalability problems where immense computational power is required for large datasets. An improved representational system which addresses these shortcomings is Noun Phrases. This representation focuses on retaining only the nouns and noun phrases present within a document and has been found to adequately represent the important article concepts (Tolle and Chen, 2000). As a consequence of its noun-centric activity, this technique uses fewer terms and can handle article scaling better than Bag of Words. A third representational technique is Named Entities, which extends Noun Phrases by selecting the article's proper nouns that fall within well-defined categories. This process uses a semantic lexical hierarchy (Sekine and Nobata, 2004) as well as a syntactic/semantic tagging process (McDonald, Chen et al., 2005) to assign candidate terms to categories.

Categorical definitions arise from the Message Understanding Conference (MUC-7) Information Retrieval task and encompass the entities of date, location, money, organization, percentage, person and time. This more abstract representational method allows for better generalization of previously unseen terms and does not possess the scalability problems associated with a semantics-only approach. A fourth representational technique is Proper Nouns which functions as an intermediary between Noun Phrases and Named Entities. This representation is a subset of Noun Phrases that names specific nouns while at the same time is a superset of Named Entities without the constraint of categories. This representation was found to be useful by removing the ambiguity associated with particular proper nouns that could either be represented by more than one named entity or fall outside one of the seven defined Named Entity categories. In a comparison study using these four representational techniques, it was found that Proper Noun representation was more effective in symbolizing financial news articles (Schumaker and Chen, 2006).

Simply assigning one representational mechanism is not sufficient to address the scalability issues associated with large datasets. One way to approach this problem is to introduce a threshold of term frequency (Joachims, 1998). This method uses a term frequency cut-off to represent terms that appear more frequently. It has the dual effect of eliminating noise from lesser used terms as well as reducing the number of features represented. Following this line of research, machine learning algorithms are unable to process these raw terms and require an additional layer of representation. A popular method is to represent the terms in binary where the term is either present or not in a

given article (Joachims, 1998). This leads to sparse matrices where the number of represented terms throughout the dataset will greatly outnumber the terms used in an individual article.

As for training a machine learning algorithm on article terms, one popular method has been used in textual financial prediction with some degree of success. Support Vector Regression (SVR), a derivative of Support Vector Machines (SVM) (Vapnik, 1995), is the regression equivalent of SVM but without the aspect of classification. Like SVM, SVR attempts to minimize its fitting error while maximizing its goal function by fitting a regression estimate through a multi-dimensional hyperplane. This method is also well-suited to handling textual input as binary representations and has been used in similar financial news studies (Tay and Cao, 2001; Schumaker and Chen, 2006).

In a more specific study on the role of financial news articles to predict stock movement direction; Lavrenko tested the effects of training on the entire universe of news articles (Universal-training) and training based on articles for a specific company (Stock Specific-training) (Lavrenko, Schmill et al., 2000a). From this study, it was found that training on Stock-Specific news articles led to more accurate predictions of price direction. It was reasoned that keywords specific to the company were more influential in determining price direction than the collection of terms from a universe of diverse sectors and industries. In a similar vein, it was also found that training on Universal keywords led to lower variances and more uniform predictions than the Stock-Specific counterparts. One of the limitations of this study was the absence of evaluation for in-between company groupings along similar sector and industrial pairings.

7.2.4 Industry Classification

There are several different Industry classification standards that group companies with similar outputs together. The first of which is the Standard Industrial Classification (SIC) system that was developed during the 1930's as a way to categorize industrial production. Using a four digit numeric code, this system was a relatively flat structure without hierarchy that grouped similar industries together with somewhat close numeric codes. Replacing the SIC in the mid-90s was the North American Industry Classification System (NAICS) which was designed for the industries of the United States, Canada and Mexico. This six-digit system utilized a five-level hierarchy of Sector, SubSector, Industry Group, Industry and Nationality where Sector used the first two digits and each successive level used an additional digit. A similar classification system developed by Morgan Stanley specifically for global commerce was the Global Industry Classification Standard (GICS). This system, which is used by Standard and Poors, employs an eight-digit, four-level hierarchy where each successive two digit pair indicates a deeper level in the hierarchy; Sector, Industry Group, Industry and Sub-Sector. In a comparison study of various industry classification systems, it was found that the GICS system of classification was more homogeneous and had lower variances in the calculated returns as compared to SIC and NAICS (Bhojraj, Lee et al., 2003). In a similar study of industry classification schemes versus analyst specialties, it was found that the GICS system best described the areas of stock analyst expertise (Boni and Womack, 2004). The homogeneity of company partitions was best matched against those industry partitions followed by professional analysts.

7.2.5 Prediction Experts, Markets and Techniques

Among trading professionals and the Internet, there is no shortage of stock advice. While some of this free advice may be susceptible to bias and market manipulation, subscription-based forecasts may not necessarily be better. There has also been a recent resurgence of interest in quantitative investing where computer programs are given some degree of control over financial investments. These systems can be constrained to analyze financial data and either issue recommendations or complete trades. While these entities may be less susceptible to biases, they are vulnerable to variable movement outside of defined parameters and are unable to evaluate market intangibles (Jelveh, 2006).

7.2.5.1 Prediction Experts

In a study between trading professionals and laypeople, both groups were given information on two stocks and were asked to select the better performer (Torngren and Montgomery, 2004). It was found that trading professionals averaged 40% accuracy, well-below chance as compared to 58% accuracy for laypeople. The study concluded that selection errors were attributable to overconfidence and the failure to weight information uncertainty. A similar study pitted trading professionals against two groups of students in pricing a series of options (Abbink and Rockenbach, 2006). This study resulted in the student groups having an 88% and 90% payoff respectively, compared to an 80% payoff for the trading professionals. In both the laypeople and student studies, it was discovered that participants without much investing experience were employing

simple heuristics to derive their answers. By contrast, trading professionals relied more on intuition than training.

As for trading professionals, there are many available and across a broad spectra of media that provide free advice and are reasonably easy to web mine. One of which is Jim Cramer, host of the CNBC Mad Money television show. Cramer is a former hedge-fund manager that boasts 24% returns over his 13 year tenure. He provides a daily list of recommendations for stocks he feels should be bought and sold. Jim Jubak, Senior Markets Editor for MSN Money, is another professional advice expert. Jubak has served as editor and contributor to several print and Internet publications with a self-reported 14% annual return on his selections. He provides a list of buy recommendations three times a week. DayTraders.com is an Internet stock recommendation service with the self-reported goal of achieving 2-3% returns per week. This service provides daily buy/sell recommendations.

7.2.5.2 Prediction Markets and Techniques

Computational prediction of financial markets is very diverse. While some prediction methods rely on market simulation or pure mathematical form, other methods rely on the collective decision-making abilities of multiple forecasters (Ashton and Ashton, 1985; Batchelor and Dua, 1995). Methods also vary depending upon the trading exchange. Figure 7.1 illustrates a taxonomy of different Prediction Market Exchanges and we detail the predictive techniques used.

Prediction Market Exchanges			
Non-Tangibles		Tangibles	
Events	Internal Mkts	Commodities	Securities

Figure 7.1. Prediction Market Exchanges

Prediction Market Exchanges can be divided into two distinct parts based upon the type of instrument that is traded; Non-Tangibles and Tangibles. Non-Tangibles, or Information Markets trade instruments that are based upon specific events or internal market expectations, whereas Tangibles refer to commodity or security trading where a tangible instrument is involved. While it can be argued that some financial instruments can blur this distinction, the main difference rests in how the markets are regulated and what instrument is being traded.

7.2.5.2.1 Event-based Markets

Event-based markets such as the Iowa Electronic Markets (www.biz.uiowa.edu/iem) and TradeSports (www.tradesports.com), function according to the probability of specific events occurring. Examples include TradeSports' 'Saddam Security,' which paid off when Saddam Hussein was ousted from power by June 30, 2003 (Herhold, 2003; Wolfers and Zitzewitz, 2004), and the Iowa Electronic Markets, which has successfully predicted the winner of US presidential elections since 1984 (Spann and Skiera, 2003) as well as the Democratic-led upset in the 2006 Congressional elections. These exchanges operate by leveraging the experience and decision-making abilities of bettors by providing them with a financial incentive to be correct. This technique has been found to be more

successful in predicting outcomes than expert analysis (Chen and Plott, 2002) and when directly compared to political exit polls, had a lower margin of error (Wolfers and Zitzewitz, 2004). Other event-based exchanges work in the same manner, such as the Hollywood Stock Exchange (www.hsx.com), which allows bettors to use simulated money to trade securities based on the perceived success of movies, actors or directors. The Washington Stock Exchange (www.thewsx.com), which speculates on political moves, bill passage and foreign relations, and NewsFutures (us.newsutures.com), which allows bettors to invest in news events most likely to occur: all provide valuable collective decision-making and assign probabilities to a wide variety of scenarios that industry insiders may not be able to pursue on their own. These methods can form strong predictions when augmented to internal processes where proprietary information is withheld (Blattberg and Hoch, 1990).

7.2.5.2.2 Internal Markets

Internal Markets are similar to Event-based ones. The main difference rests in who has access to the market exchange. In event-based markets everyone has access, whereas in internal markets, organizations can restrict access to certain employees. These markets can also be used as forecasting tools. Some notable examples include Hewlett-Packard, which found their internal market better able to forecast printer sales than existing internal processes (Chen and Plott, 2002) and Google, which uses internal markets to determine corporate strategy such as opening new office campuses (Cowgill, 2005).

7.2.5.2.3 Commodities

Commodity markets, which typically encompass the familiar markets of metals, crops, currency and animals, have recently begun to infuse with information markets. One such example is eTopps (www.etopps.com), which is commodity-based by allowing users to physically own a sports trading card, but is also event market driven where prices are an indirect reflection of player performance. Besides the usual buy, sell and trade activity, this exchange also rewards bettors with financial incentives if the players in their portfolio achieve a specific goal (e.g., the Cleveland Indians win more than 10 games in the month of June). This structure allows consumers to either collect cards based upon their player/team preferences or collect based on speculative gains. Because of this participatory decision, eTopps cannot be considered either a pure information or pure commodity market. Another twist on this commodity exchange is the use of an auction format for buying and selling cards. While this method prices cards at the last auction transaction price, other auction-related concerns become evident such as jump-bidding (Easley and Tenorio, 2004), shill-bidding and artificial prices set by market manipulators which further blurs the true price between a commodity and event.

Other information markets that trade based upon weather temperature, the economic damage from a hurricane season and labor statistics have been subjected to recent regulation by the Commodities Futures Trading Commission (CFTC), with their introduction of contracts on the Chicago Mercantile Exchange (Hahn and Tetlock, 2006). While the instruments traded may not fall under the traditional view of commodities, public policy has been moving towards the regulation of information markets in this

regard with exceptions for trading accounts under \$500 as in the Iowa Electronic Markets.

7.2.5.2.4 Securities

The computational prediction of security markets follows two distinct paths. The first of which parallels information markets where an artificial market is constructed and predictions are made by varying system inputs (LeBaron, Arthur et al., 1999; Raberto, Cincotti et al., 2001; Raberto, Cincotti et al., 2003). These inputs could be as simple as varying the time in which new information is received and acted upon (LeBaron, Arthur et al., 1999) or more complex such as the modeling of an entire stock market exchange (Raberto, Cincotti et al., 2003).

The second computational security prediction type is that of a quantitative nature. In real market predictions, quantitative systems, or quants, follow various stock parameters and are essentially automated versions of existing market strategies (e.g., look for high growth, undervalued securities, etc.) except with the ability to follow all stocks in real-time. This advantage has led quants to steadily outperform market averages by 2-3% for the past several years (Jelveh, 2006).

While the exact strategies used are a closely guarded secret, some quantitative funds do disclose the parameters they track. The exact number and weights assigned to these parameters fluctuate frequently to keep pace with market conditions and to tweak model performance. Quant programs are also becoming a part of the individual investor's toolbox as well. Wealth Lab Pro software (www.wealth-lab.com) allows individual

investors to track upwards of 600 parameters through 1,000 pre-set investment strategies (Lucchetti and Lahart, 2006) and many brokerage houses are giving their investors quantitative software as a customer retention tool.

The number of quant funds has increased from just a few in 2001 to over 150 by the beginning of 2006 (Burke, 2006). These funds have also branched themselves out, able to cover worldwide financial markets or focus exclusively on a select boutique of securities.

Quants generally operate in the following two-stage manner. First, securities are analyzed using a technical analysis strategy and securities not meeting basic criteria are removed from further analysis. Second, the quantitative algorithm rank orders the remaining stocks. Quantitative algorithms can be classified into three major categories based on their referent discipline; artificial intelligence, statistics and mathematical analysis.

7.2.5.2.4.1 Artificial Intelligence

Artificial Intelligence has mostly contributed algorithms that deal with optimization and machine learning. Examples such as Genetic Algorithms, Support Vector Machines (SVM), Hidden Markov Models and Neural Networks all take input parameters from financial securities and return predictions based on the hidden patterns. However, most of these techniques are constrained to either identify the most relevant parameters or evaluate stock data in terms of likely movement directions.

Genetic Algorithms utilize a global search and optimization approach to identify the parameters that have the greatest impact on stock price performance (LeBaron, Arthur et al., 1999). By using the concept of evolution and survival of the fittest, the genetic algorithm can quickly search the function landscape and return the local optimum. This technique has been used in determining the predictive value of stock bulletin boards (Thomas and Sycara, 2002).

SVM is a machine learning algorithm that can classify the potential stock price into likely price directions such as rise, drop or no recommendation (Fung, Yu et al., 2002). This supervised learning algorithm assigns financial inputs to a hyperplane and then seeks to classify the data points by constructing a dividing line that best maximizes the divide and minimizes classification error.

Hidden Markov Models are another pattern recognition and classification algorithm that seeks to determine the hidden parameters from the observed instances by assigning probabilities. These techniques have been used in forecasting discrete prices for airline stocks using the prior day's data (Hassan and Nath, 2005).

Neural Networks are another method which functions by weighting various stock parameters. Neural Networks have also been used in other financial applications such as the determination of credit risk (Baesens, Setiono et al., 2003), financial distress (West, Dellana et al., 2005) and even potential bank failure (Tam and Kiang, 1992). All of these methods performed marginally better than chance (Yoon and Swales, 1991; LeBaron, Arthur et al., 1999; Fung, Yu et al., 2002).

7.2.5.2.4.2 Statistical Approach

Statistical approaches use simulations and probability methods such as Monte Carlo and Game Theory (Cai and Wurman, 2005). In Monte Carlo simulations, the problem of price prediction is too difficult to approach directly, so input parameters are given a series of suitable random numbers and are observed for how close they arrive at the predicted value (Stauffer, 2000).

In Game Theory, price prediction is modeled in terms of strategy and potential payoff. It is theorized that the players in the game will evaluate other player's strategies and adopt a stance which will earn them the best payoff. However, these types of systems do not function well in stock market prediction because of new entrants, the changing strategies of other players and the inherent difficulty in predicting price changes.

7.2.5.2.4.3 Mathematical Analysis

Mathematical approaches borrow heavily from the areas of applied mathematics and econo-physics. This branch of predictive algorithms uses more complex mathematical formalisms, such as Percolation Methods, Log-Periodic Oscillations and Wavelet Transforms to model future prices (Stauffer, 2000).

Percolation Methods use dimensional membranes to constrict trading actions and price movements. In one such example, a market exchange is modeled where a cluster of traders indicate a single company and at each time interval traders are given the choice to

buy, sell or sleep. This method is then used to model the supply and demand of securities and their potential impact on security prices.

Log-Periodic Oscillations use long-term historical data to describe macro movements in the market, such as impending crashes and market bubbles. While it has been suggested that previous ‘crash’ predictions from this model had more to do with luck, market psychology would make these oscillations more pronounced through rapid sell-offs in the face of an anticipated crash (Stauffer, 2000).

In Wavelet Transforms, input parameters are consecutively sampled to provide a finer-grained resolution into the microscopic movements that comprise the input signal. These successive filters can then be analyzed to identify parameter relations.

From our survey of the literature, we identified several potential gaps. The first of which was that machine learning systems have only been tested on either Universal financial news articles or on a Stock-Specific basis. Training a system using an in-between classification method may result in more accurate results and a better-tuned prediction. Secondly, we are not aware of any system that has been compared against both trading professionals and existing quantitative systems. While there are several studies that pit professionals against non-professionals and some mechanized techniques against others, it would be useful to make a broad comparison as a yardstick evaluation to see just how well a financial news prediction system stacks up.

7.3 Proposed Approaches

From these gaps, we have formulated several research questions with which to explore. The first of which is:

- What effect does GICS partitioning of articles have on the prediction of stock price?

Following Bhojraj's conclusion that GICS classification is superior in its homogeneity of industries, we propose to investigate the prediction accuracy of a system that explores the in-between areas of Sector and Industry-level grouping and what impact it may have on prediction results.

The second research question we will pursue is:

- How effective is a discrete prediction model versus the market and human traders?

Given our system, how does its predictive ability stack up to the advice given by trading professionals, quantitative funds and the overall market in general?

In order to properly evaluate our research questions, we designed the AZFinText system. Figure 7.2 illustrates the AZFinText system design.

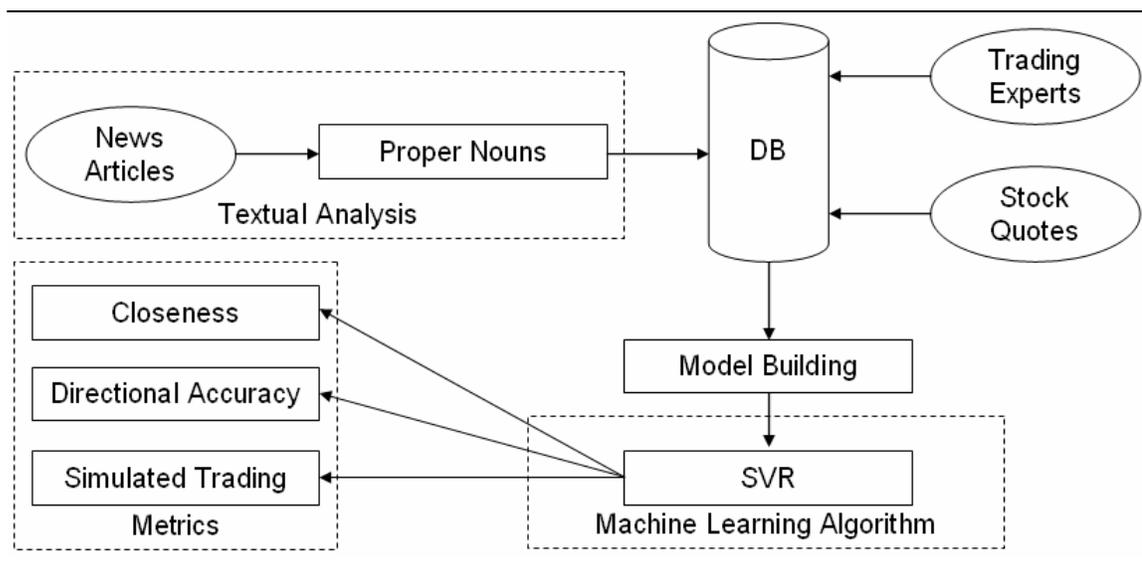


Figure 7.2. AZFinText system design

In examining the AZFinText system from Figure 7.2, there are several major components to describe. The first of which is Textual Analysis. This component gathers financial news articles and represents them by their proper nouns. This module further limits extracted features to three or more occurrences in any document, which cuts down the noise from rarely used terms (Joachims, 1998). The second major component is Stock Quotations which gathers stock price data in one minute increments. The third major component is Model Building. This component has derived from prior empirical testing and includes article term representations and the stock price at the time the news article was released. This combination of parameters was previously tested and judged to provide superior performance to all combinations tested (Schumaker and Chen, 2006). The fourth major component is the Trading Experts which gathers the daily buy/sell recommendations from a variety of trading experts. Lastly, the metrics component evaluates system output.

7.4 Experimental Design

For our experiment, we selected a consecutive period of time to serve as our experimental baseline. We selected a five-week research period of Oct. 26, 2005 to Nov. 28, 2005, which incorporates twenty-three trading days. The five-week period of study was selected because it gathered a comparable number of articles in comparison to prior studies: 6,602 for Mittermayer (Mittermayer, 2004) and 5,500 for Gidofalvi (Gidofalvi, 2001). We also observe that the five-week period chosen did not have unusual market conditions and would be a good testbed for our evaluation. In order to identify companies with more financial news, we further limited the scope of activity to focus on companies listed in the S&P 500 as of Oct. 3, 2005. Articles gathered during this period were restricted to occur between the hours of 10:30am and 3:40pm. While trading starts at 9:30am, we felt it important to reduce the impact of overnight news on stock prices and selected a period of one-hour to allow prices to adjust. The 3:40pm cut-off for news articles was selected to disallow any +20 minute stock predictions to occur after market hours. A further constraint was introduced to reduce the effects of confounding variables, where two articles on the same company cannot exist within twenty minutes of each other or both will be discarded. The above processes had filtered the 9,211 candidate news articles gathered during this period to 2,809, where the majority of discarded articles occurred outside of market hours. Similarly, 10,259,042 per-minute stock quotations were gathered during this period. This large testbed of time-tagged articles and fine-grain stock quotations allow us to perform our evaluation systematically.

For the machine learning algorithm we chose to implement the SVR Sequential Minimal Optimization (SMO) (Platt, 1999) function through Weka (Witten and Eibe, 2005). This function allows for discrete numeric prediction instead of classification. We selected a linear kernel and ten-fold cross validation. A similar prediction method was employed in the forecasting of futures contracts (Tay and Cao, 2001). To test the effects of GICS partitioning, we trained our system on keywords for all stocks, each GICS Sector, Industry Group, Industry and Sub-Industry, as well as trained on articles for each specific company. Output from these models is then evaluated on a three metric platform consistent with prior research (Schumaker and Chen, 2006). The first metric used was measures of Closeness which evaluates the accuracy of the predicted discrete value to the actual future price (Cho, Wuthrich et al., 1999). It does so by calculating the Mean Square Error (MSE) of the predicted values versus the actual +20min stock price. The second metric is Directional Accuracy. This metric simply analyzes the direction of the predicted value versus the direction of the actual +20min stock price with respect to the price at the time the article was released (Gidofalvi, 2001). The third metric used was a Simulated Trading Engine which evaluated the potential gain/loss of our predictions. This metric was wholly based on prior literature and if an absolute value was more or less than 1% away from the price at the time the article was released, the system will then buy or short the stock depending upon the direction of the prediction (Lavrenko, Schmill et al., 2000b). Our system differed slightly from Lavrenko's in that we invested blocks of \$1,000 compared to Lavrenko's \$10,000. We then divest our holdings and take the +20min stock price to derive our potential gain/loss.

As an example of our system's operation, consider Figure 7.3.

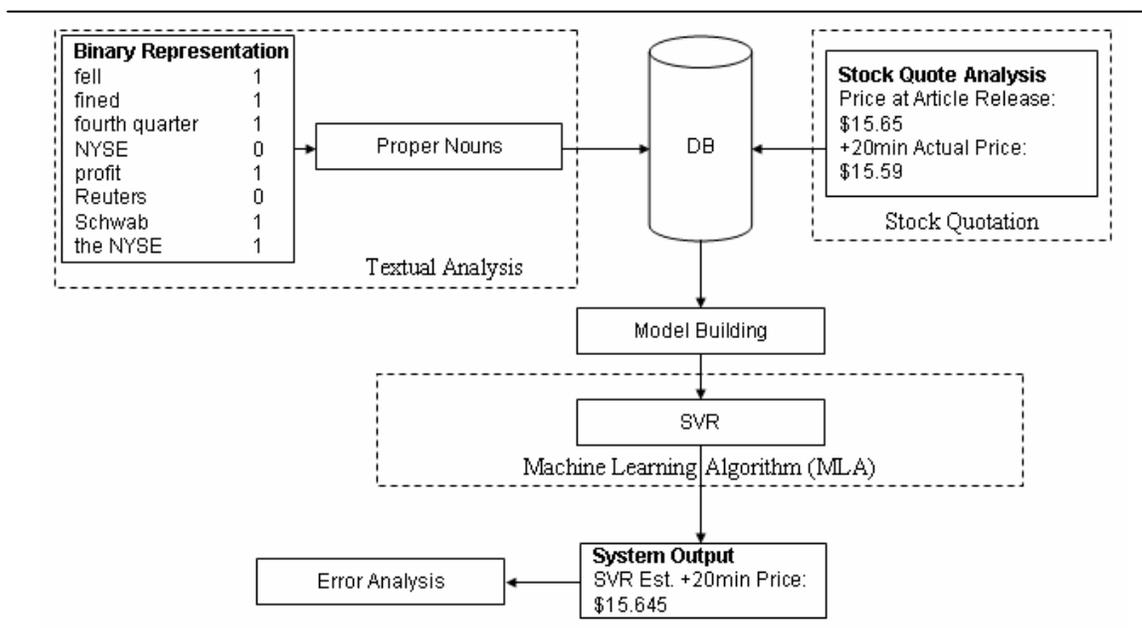


Figure 7.3. Example of AZFinText Processing

The first part of the operation is to extract news articles. The entire corpus of news articles are then evaluated for Proper Nouns and each news article is represented by its Proper Nouns in binary as to whether the feature is present or not in that particular article. These representations are then stored within the database. Concurrently, stock quotations gathered on a per minute basis and are similarly stored. To build a model, we extract the representational Proper Nouns from articles and couple them with the stock quotation at the time the article was released. This collection of material is then passed into the Machine Learning Algorithm (MLA) where the predicted price, in our Figure 7.3 example, was \$15.645. Since the actual +20min stock price was \$15.59, our measures of Closeness for this particular example would be 0.003025. With the stock price at the time the article was released at \$15.65 and going in the same direction as our prediction,

we pass the Directional Accuracy test for this article. Since the predicted price was less than 1% movement from the original stock price, our Simulated Trading Engine declines to make a trade based on this article.

To provide an overall sense of the data used throughout this experiment for the different classification models, we present Table 7.1 to illustrate some basic statistics on our dataset.

	Universal	Sector	Industry Group	Industry	Sub-Industry	Stock-Specific
Number of GICS Categories	1	10	24	61	119	500
Usable Categories	1	10	24	50	76	71
Minimum Number of Firms per Category		9	6	1	1	
Maximum Number of Firms per Category		89	36	22	16	
Average Number of Firms per Category	500	50	21	10	5	
Standard Deviation of Firms per Category		26.7	9.8	5.6	3.6	
Minimum Number of Articles per Category		100	16	11	10	10
Maximum Number of Articles per Category		518	266	150	139	58
Average Number of Articles per Category	2,809	281	117	55	34	18
Standard Deviation of Articles per Category		160.8	66.0	37.6	28.3	9.1
Minimum Number of Terms per Category		242	57	42	23	15
Maximum Number of Terms per Category		974	606	415	377	194
Average Number of Terms per Category	3,710	567	288	158	105	61
Standard Deviation of Terms per Category		291.7	134.4	88.5	67.8	30.8

Table 7.1. Basic statistics on the AZFinText datasets

There are several facets of the above table that deserve further explanation. The first of which is that within the categories of Industry, Sub-Industry and Stock Specific, the number of GICS categories does not equal the number of usable categories. This is from some of the GICS categories having fewer than ten news articles available for training which reduces the number of categories that can be used. We examined the role of decreasing articles and terms with each increasing level of classification and found that they had little impact on the results.

To test our second research question on the effectiveness of the AZFinText against Professional Traders and quantitative funds, we arbitrarily selected a group of experts and

funds for comparison. The experts we chose were Jim Cramer, Jim Jubak and DayTraders.com. Selection criteria mainly focused on the availability and ease of gathering recommendations in an automated fashion, however, we were conscious to spread out experts across a variety of media sources. We then invest \$1,000 into each buy/short recommendation, buying at the opening price and then selling it at the close of trading. We further assume that there is a zero transaction cost, consistent with Lavrenko (Lavrenko, Schmill et al., 2000b). Table 7.2 shows the recommendation breakdown for each of these Trading Professionals during our period of study.

	Cramer	Jubak	DayTraders.com
Number of Buy Recommendations	322	13	10
Number of Sell Recommendations	164	0	1
Number of Unique Companies Mentioned	291	13	5

Table 7.2. Recommendation breakdown of each Trading Professional

While Jubak and DayTraders.com had few trading recommendations during our period of study, we felt that their contribution would help add depth to the overall trading professionals' advice. Similarly, since the gathered recommendations did not provide any predicted price information, we can only compare them to AZFinText using the Directional Accuracy metric.

As for comparisons against quantitative funds, we selected the top ten performing quantitative mutual funds of 2005 (Burke, 2006). Since Quant trading strategies and predictions are closed-source, we gathered the observable fund prices at the beginning and end of our research period to make performance comparisons against our system using the Simulated Trading metric.

7.5 Experimental Results and Discussion

7.5.1 Sector-based training has the best performance:

To answer our first research question “*What effect does GICS partitioning of articles have on the prediction of stock price?*,” we trained our system on the different GICS classification levels and evaluated them with our three metrics. Results are presented in Table 7.3.

	Closeness		Directional Accuracy		Simulated Trading	
	Average	Std Dev.	Average	Std Dev.	Average	Std Dev.
Universal	0.0443	0.1081	58.17%	0.4933	2.86%	0.1085
Sector	0.1954	3.7102	71.18%	0.4530	8.50%	0.2192
Group	3.3129	23.1133	66.12%	0.4734	4.57%	0.1789
Industry	16.1087	71.2319	62.37%	0.4845	2.02%	0.1572
Sub-Industry	26.1330	102.6304	57.50%	0.4944	1.09%	0.1501
Stock Specific	1.0443	2.7615	56.92%	0.4954	1.01%	0.1295

Table 7.3. Three metric evaluation of the different GICS levels

The first item to notice, is that Universal training has the lowest average Closeness score of 0.0443. Sector-based training had the highest Directional Accuracy and Simulated Trading scores of 71.18% and 8.50% respectively (p-values < 0.05). Comparing our results to the Universal versus Stock Specific research conducted by Lavrenko, our Stock Specific model should have a lower Closeness score than Universal training. However, we observed the opposite result where Universal keywords had a lower Closeness score of 0.0443 as compared to Stock Specific’s 1.0443. Returning to Lavrenko’s work, we should also expect to see lower variances with Universal training. Confirming this, Universal training had a standard deviation of 0.1081 as compared to 2.7615 for Stock Specific. We believe that the observed uniformity is a result of

Universal's homogenous keywords that behave similarly across all stocks. However, when we expand this work to include in-between GICS categories, Closeness scores gradually increase with each successive level and suddenly drop at Stock Specific training. Directional Accuracy and Simulated Trading appear to spike at the Sector level and then steadily decrease. We believe that this behavior is the result of small inconsistencies within the GICS categories. We further investigated whether the reduction of either the number of articles or terms were a part of this phenomenon. However, when looking at the results of Stock Specific training, which used the least number of articles and terms, its numbers were somewhat similar to Universal, leaving the GICS categories with spiking behavior. Although both Bhojraj and Boni found the GICS classification system to be superior to similar systems (Bhojraj, Lee et al., 2003; Boni and Womack, 2004), its classifications are not perfect.

Analyzing model performance across all levels including those of the GICS, we found that Sector-based training performed better than similar classifications. While Sector had the best Directional Accuracy score of 71.18% and Simulated Trading of 8.50% return, Sector also had the second-lowest Closeness score of 0.1954 to Universal's 0.0443 (p-values < 0.05). This would seem to indicate that Sector-based training was better able to forecast stock price direction and capitalize on forecasted trades, yet was not as precise as Universal-training in obtaining a future price.

To further investigate the reasons behind Sector's predictive success, Figure 7.4 illustrates a histogram of trade returns for the Sector model.

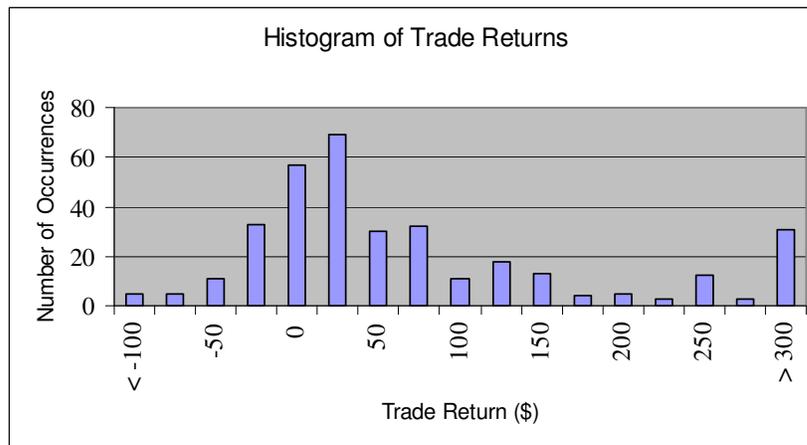


Figure 7.4. Histogram of Trade Returns

From this figure, approximately 69% of trades had returns between $-\$100$ and $\$100$, however, the average is skewed towards the positive. It was also interesting to see that 29 trades resulted in excess of $\$300$ returns while only 5 trades lost $\$100$ or more. This phenomenon was found to be a function of article length where longer articles contained more Proper Nouns and hence were more predictable. These returns arose from an outlay of $\$342,000$ (i.e.; 342 trades) and an excess return of $\$29,059$.

Digging further into the components of each Sector, we present Table 7.4 which breaks down each metric and compares them to a composite index return over the five week period of study.

Sector	Sector Name	AZFinText			Index
		Closeness	Dir. Acc.	Sim. Trade	% Return
10	Energy	0.1951	68.79%	17.50%	1.60%
15	Materials	3.8269	63.00%	5.50%	8.76%
20	Industrials	0.0206	72.64%	2.10%	5.96%
25	Consumer Discretionary	0.0616	70.04%	19.40%	6.05%
30	Consumer Staples	0.0218	64.81%	2.60%	2.07%
35	Health Care	0.0239	70.15%	-1.00%	4.28%
40	Financials	0.0189	76.02%	6.60%	8.25%
45	Information Technology	0.0220	72.59%	12.70%	6.47%
50	Telecommunication Services	0.4684	72.36%	0.50%	7.66%
55	Utilities	0.0475	68.24%	-4.30%	-7.64%

Table 7.4. Sector breakdown and evaluation

From this table, the Financials sector, Sector 40, had the lowest Closeness score of 0.0189, the highest Directional Accuracy, 76.02% and a Simulated Trading return of 6.60% (p-values < 0.05). However, this trading return did not outperform the market's composite Financials return of 8.25%. In a similar vein, the Materials sector, Sector 15, performed the worst with a Closeness score of 3.8269, Directional Accuracy of 63.00% and Simulated Trading return of 5.50% compared to the market's Materials composite of 8.76%. Sector 25, Consumer Discretionary had the best Simulated Trading return of 19.40% while Sector 55, Utilities had the worst return of -4.30% but did manage to lessen the loss of the composite average at -7.64%.

To answer the follow-up question of why the Consumer Discretionary sector performed better than Utilities; we further investigated the Simulated Trading results of the constituent companies within each sector. Table 7.5 illustrates the component companies for Consumer Discretionary while Table 7.6 shows the same for Utilities. Not shown, Consumer Discretionary had 11 trades for 9 companies with \$0 gain/loss and Utilities had 2 trades for 2 companies with \$0 gain/loss.

Consumer Discretionary Sector			
Simulated Trading Engine			
Company	# Trades	Gain/Loss	% Return
BBY	2	\$2	0.20%
BLI	4	-\$6	-0.60%
CC	1	\$1	0.10%
CCL	1	\$1	0.10%
CTB	3	-\$1	-0.10%
CTX	1	\$10	1.00%
DCN	1	\$2	0.20%
DDS	1	\$6	0.60%
DHI	2	\$4	0.40%
DJ	1	\$6	0.60%
EBAY	3	\$8	0.80%
EK	2	\$9	0.90%
F	13	-\$5	-0.50%
FD	1	\$3	0.30%
FO	2	\$12	1.20%
GM	4	\$24	2.40%
GPS	3	\$8	0.80%
GT	1	\$11	1.10%
HAS	1	\$2	0.20%
HOT	1	\$5	0.50%
HRB	1	\$2	0.20%
IGT	1	\$6	0.60%
JCP	1	\$9	0.90%
KSS	1	-\$1	-0.10%
NYT	1	\$4	0.40%
OMX	1	\$3	0.30%
TGT	2	\$4	0.40%
TJX	4	\$37	3.70%
TWX	3	-\$5	-0.50%
VC	2	\$22	2.20%
YUM	1	\$1	0.10%

Table 7.5. Company components of the Consumer Discretionary Sector

Utilities Sector			
	Simulated Trading Engine		
Company	# Trades	Gain/Loss	% Return
AES	1	\$2	0.20%
CIN	2	\$8	0.80%
CPN	6	-\$50	-5.00%
DTE	1	-\$1	-0.10%
DUK	1	\$2	0.20%
DYN	4	-\$17	-1.70%
KSE	1	\$7	0.70%
NI	1	\$2	0.20%
PPL	1	\$1	0.10%
TE	1	\$1	0.10%

Table 7.6. Company components of the Utilities sector

From these tables, Simulated Trading made trades in 40 of the 89 Consumer Discretionary companies and 12 of the 33 Utility companies. Table 7.5 shows that company TJX, The TJX Companies Inc., posted the largest gains with a 3.70% return on investment compared to a 10.15% increase over the five week period. For the Utilities sector of Table 7.6, there were quite a few low return transactions and the CPN, Calpine Corporation, trade with a -5.00% return further hurt the results of this sector. It is also notable that a good portion of trades in both Sectors netted returns of \$5 or less.

7.5.2 Sector-based training outperforms Professional Traders & Quants:

To answer our second research question “*How effective is a discrete prediction model versus the market and human traders?*,” we measure the Directional Accuracy of our Sector-based approach versus the Trading Professionals, as shown by Table 7.7.

	Dir. Acc.
Sector	71.18%
Cramer	57.00%
Jubak	69.23%
DayTraders	81.82%

Table 7.7. Comparison of Directional Accuracy results

From this table, our Sector-based training performed better in Directional Accuracy (71.18%) than Jim Cramer at 57.00% and Jim Jubak at 69.23%, however, our system did not perform as well against DayTraders.com at 81.82% (p-values < 0.05). Even with statistical significance we must be mindful of the sparsity of stock recommendations from both Jubak and DayTraders. This sparsity of recommendations, as previously shown within Table 7.2, would suggest that Jim Jubak and DayTraders.com may be more conservative in their selection approaches.

Comparing our system against both Trading Professionals and the top 10 quant funds according to their trailing one-year returns (Burke, 2006), results in the Simulated Trading results of Table 7.8. Since Quant trading strategies and predictions are closed-source, we could only gather observable price movements over our trading period and make comparisons using our Simulated Trading metric.

	Simulated Trading
AZFinText	8.50%
S&P 500	5.62%
Trading Professionals	
Cramer	0.15%
Jubak	-0.14%
DayTraders.com	0.46%
Quantitative Funds	
ProFunds Ultra Japan Inv (UJPIX)	24.73%
ProFunds Ultra Japan Svc (UJPSX)	24.59%
American Century Global Gold Adv (ACGGX)	12.96%
American Century Global Gold Inv (BGEIX)	12.93%
Quantitative Advisors Emerging Markets Instl (QEMAX)	8.16%
Quantitative Advisors Emerging Markets Shs (QFFOX)	8.15%
Lord Abbett Small-Cap Value Y (LRSYX)	5.22%
Lord Abbett Small-Cap Value A (LRSCX)	5.19%
Quantitative Advisors Foreign Value Instl (QFVIX)	4.99%
Quantitative Advisors Foreign Value Shs (QFVOX)	4.95%

Table 7.8. Simulated Trading results of Professionals and Quants

As evidenced by this table, AZFinText with its 8.50% return outperformed the overall market, 5.62% and the trading professionals (p-values < 0.05). Comparing AZFinText against the top 10 quants shows AZFinText performed well, outperforming 6 of the top 10 quant funds. It is interesting to note that the four better performing quants were trading in the Nikkei and gold markets where AZFinText was constrained to the companies in the S&P 500. In making a more direct performance comparison, Table 7.9 shows the trade returns of AZFinText versus several quant funds that are also operating within the S&P 500.

	Return
AZFinText	8.50%
Vanguard Growth & Income (VQNPX)	6.44%
BlackRock Investment Trust Portfolio Inv A (CEIAX)	5.48%
RiverSource Disciplined Equity Fund (ALEIX)	4.69%

Table 7.9. Simulated Trading results of S&P 500 quants

As shown in this table, AZFinText performed better than its peer quant funds. It is worthwhile to point out that AZFinText's success came mostly from making predictions from financial news articles and stock quotes, whereas quants used sophisticated mathematical models on a large set of financial variables. We believe that our research helps identify a promising research direction in financial text mining. However, more research is critically needed.

7.6 Conclusions

Using a triangulation of evaluation methods that have been used mostly independently of each other, we determined that Sector-based training had the better performance of the models tested. Sector had the best Directional Accuracy at 71.18% and Simulated Trading of 8.50% return on investment. Sector also had the second lowest Closeness score, 0.1954 as compared to Universal, 0.0443. This would seem to indicate that Sector-based training was better able to forecast stock price direction and capitalize on forecasted trades, yet was not as precise as Universal training in obtaining a future price. In an analysis of the individual Sectors we found that the Financials Sector was unusually predictive with a Closeness score of 0.0189 and Directional Accuracy at 76.02%. This would mean that this sector was more sensitive to our representations than others.

When comparing the Sector-based approach to trading professionals, we found that AZFinText had a Directional Accuracy of 71.18%, which was second-best to DayTraders.com 81.82%. However, in Simulated Trading, Sector-based training

performed the best with an 8.50% return and even outperformed six of the top ten quantitative funds. We believe that our system was better able to capitalize on the Proper Nouns presented at the Sector-level. This ability translated to consistently better predictions than comparable human experts while reacting to market changes faster than the history-dependent quantitative funds.

CHAPTER 8: QUANTITATIVE STRATEGIES IN PREDICTION

This chapter studies the coupling of basic quantitative portfolio selection strategies with a financial news article prediction system, AZFinText. By varying the degrees of portfolio formation time, we found that a hybrid system using both quantitative strategy and a full set of financial news articles, performed the best. With a 1-week portfolio formation period we achieved a 20.79% trading return using a Momentum strategy and a 4.54% return using a Contrarian strategy over a five-week holding period. It was also found that trader overreaction to these events led AZFinText to capitalize on these short-term surges in price.

8.1 Background

Predicting activity in the stock market has always had a certain appeal to researchers. While numerous attempts have been made, the difficulty has always centered on the behaviors of human traders within this socially constructed system. With behavioral parameters not fully defined and constantly changing, making accurate predictions in this environment has been difficult. To further create confusion, there are two diametrically opposed philosophies of stock market research; fundamental and technical analysis techniques (Technical Analysis, 2005). While fundamental analysis leverages the security's relative data, ratios and earnings, technical analysis utilizes charts and modeling techniques based on historical trading volume and pricing. The main issue between them becomes *Can the market be timed or not?*

With the advent of cheaper processing and knowledge acquisition techniques, the roles of computers in stock prediction has increased dramatically, where they have become mostly automated versions of existing fundamental and/or technical strategies. Their goal is to achieve better returns than their human counterparts by removing the elements of emotions and biases from trading (Jelveh, 2006). The downside of these systems is that they lack intuition and context where they continue buying battered stocks even after unfavorable news events, such as losing a costly court battle. These systems instead rely on news events being translated to numeric data before appropriate decisions can be made. This problem introduces serious lag-time into decisions and in some cases, trades must be overridden by human analysts.

The motivation of this chapter is the following. We plan to build and test a hybrid quantitative system that incorporates both traditional quantitative trading strategies and a financial news article prediction piece. We plan to test various quantitative strategies and couple them with a financial news article prediction engine to find an optimal trading system.

This chapter is arranged as follows. Section 8.2 provides an overview of literature concerning Stock Market prediction, textual representations and quantitative portfolio building techniques. Section 8.3 describes our proposed approaches and the AZFinText system. Section 8.4 provides an overview of our experimental design. Section 8.5 details our experimental findings and discusses their impact on stock market prediction. Section 8.6 delivers our conclusions and a brief discourse on future research directions.

8.2 Related Works

Within stock market research, there have been two theories that have had a significant impact on predicting security prices; Efficient Market Hypothesis (EMH) and Random Walk Theory. In Fama's EMH, the price of a security is a reflection of complete market information and whenever a change in financial outlook occurs, the market instantly adjusts the price of the security to reflect this new information (Fama, 1964). Within EMH the amount of information can be varied to encompass three distinct levels; the weak form, the semi-strong and the strong form. In weak EMH, only historical data is embedded within the current price. The semi-strong form goes a bit farther by incorporating historical and current public information into its prices. The strong form includes historical, current public information and private information, such as Insider Trading. From this theory, it was believed that markets behaved efficiently and that instantaneous price corrections would make prediction models useless.

Random Walk Theory is similar to the Semi-Strong EMH model where all information is contained within the current price and is worthless for future prediction. This theory is slightly different in its approach by insisting that short-term price movements are indistinguishable from random activities (Malkiel, 1973). This random activity produces unpredictable prices and makes it impossible to consistently outperform the market.

Studying the decisions of traders and the micro-effects of trading behavior on the scale of a market exchange is extremely difficult. However, to obviate this difficulty and test the impact of fundamental and technical trading strategies, LeBaron created an

artificial stock market with simulated traders which can be dissected to inspect individual trading decisions (LeBaron, Arthur et al., 1999). He introduced new pieces of information into the market and adjusted the amount of time between when an individual trader would receive information and act upon it. It was found that traders with longer waiting times would form fundamental strategies (latching onto company performance data) while those that waited less time developed technical strategies (such as timing trades). This study had a more important contribution because it discovered a lag existed between the time that information was introduced to when the market would correct itself to equilibrium. This apparent delay in market behavior helped to dispel the instantaneous correction notions of EMH and lent support to the idea that the stock market could be forecast in short durations following the introduction of new information. Subsequent research into how long this short duration of time is, led to the discovery of a twenty minute window of opportunity before and after a financial news article is released (Gidofalvi, 2001). Within this window, weak prediction of the direction of a stock price was found to be possible.

8.2.1 Financial News Articles

The introduction of new information into the market happens all the time. While rumors, eavesdropping and scandals can all move security prices, financial news articles are a more stable and generally a more trustworthy source. However, the exact relationship between financial news articles and stock price movement is complex. While the information contained in financial news articles can have a visible impact on a

security's price (Wuthrich, Cho et al., 1998; Lavrenko, Schmill et al., 2000a; Gidofalvi, 2001; Mittermayer, 2004), sudden price movements can still occur from large unexpected trades (Camerer and Weigelt, 1991).

The first challenge of textual financial prediction is to process the large amounts of textual information that exist for securities. This material includes required reports such as periodic SEC filings, press releases and financial news articles reporting both unexpected events and routine news. These documents can be automatically parsed using Natural Language Processing (NLP) techniques and can identify the specific article terms most likely to cause dramatic share price changes. This method can take advantage of arbitrage opportunities faster than human counterparts by repeatedly forecasting price fluctuations and executing immediate trades.

Obtaining timely financial documents from Web sources is a critical step. Luckily there are a variety of financial news aggregation sites that provide such services. One of which is Comtex which offers real-time financial news in a subscription format. Another source is PRNewsWire which offers free real-time and subscription-based services. By contrast, Yahoo Finance is a compilation of 45 different news sources including the Associated Press, Financial Times and PRNewsWire among others. This source provides a variety of perspectives and timely news stories regarding financial markets.

8.2.2 Textual Representation

Once we have gathered the financial news articles, we must ultimately represent their important features in some machine-friendly form. One technique is to use a Bag of

Words approach which has been extensively used in textual financial research (Lavrenko, Schmill et al., 2000a; Gidofalvi, 2001). This process involves removing the meaningless stopwords such as conjunctions and declaratives from the text and using what remains as the textual representation. While this method has been popular, it suffers from noise issues associated with seldom-used terms as well as problems of scalability where immense computational power is required for large datasets. An improved representational system which addresses a majority of these shortcomings is Noun Phrases. This representation retains only the nouns and noun phrases within a document and has been found to adequately represent the important article concepts (Tolle and Chen, 2000). As a consequence, this technique uses fewer terms and can handle article scaling better than Bag of Words. A third representational technique is Named Entities, which is an extension of Noun Phrases. It functions by selecting the proper nouns of an article that fall within well-defined categories. This process uses a semantic lexical hierarchy (Sekine and Nobata, 2004) as well as a syntactic/semantic tagging process (McDonald, Chen et al., 2005) to assign candidate terms to categories. Selected categorical definitions are prescribed by the Message Understanding Conference (MUC-7) Information Retrieval task and they encompass the entities of date, location, money, organization, percentage, person and time. This method allows for better generalization of previously unseen terms. It also does not possess the scalability problems associated with a semantics-only approach. A fourth representational technique is Proper Nouns which functions as an intermediary between Noun Phrases and Named Entities. This representation is a subset of Noun Phrases which selects specific nouns and is also a

superset of Named Entities, but without the constraint of pre-defined categories. This representation removes the ambiguity associated with those particular proper nouns that could be represented by more than one named entity category or fall outside one of the seven defined Named Entity categories. In a comparison study using these four representational techniques, it was found that Proper Noun representation was more effective in symbolizing financial news articles (Schumaker and Chen, 2006).

Simply assigning one representational mechanism is not sufficient to address the scalability issues associated with large datasets. A common solution to this problem is to introduce a term frequency threshold (Joachims, 1998). This method uses a term frequency cut-off to represent article terms that appear more frequently. This technique has the dual effect of eliminating noise from lesser used terms and reducing the number of features that need to be represented. Following this line of research, machine learning algorithms are unable to process raw article terms and require an additional layer of representation. A popular method is to represent the terms in binary where the term is either present or not in a given article (Joachims, 1998). This solution leads to sparse matrices where the number of represented terms throughout the dataset will greatly outnumber the terms used in an individual article.

Once financial news articles are represented, computers can then begin the task of identifying patterns of predictable behavior. One accepted method, Support Vector Regression (SVR), is a regression equivalent of Support Vector Machines (SVM) but without the aspect of classification (Vapnik, 1995). Like SVM, SVR attempts to minimize its fitting error while maximizing its goal function by fitting a regression

estimate through a multi-dimensional hyperplane. This method is also well-suited to handling textual input as binary representations and has been used in similar financial news studies (Tay and Cao, 2001; Schumaker and Chen, 2006).

8.2.3 Quantitative Portfolio Building

There are a variety of considerations and techniques involved in building a quantitative portfolio. In Figure 8.1, we illustrate a taxonomy of the major portfolio-building considerations partially based on the works of Torre and Rudd (Torre and Rudd, 2004).

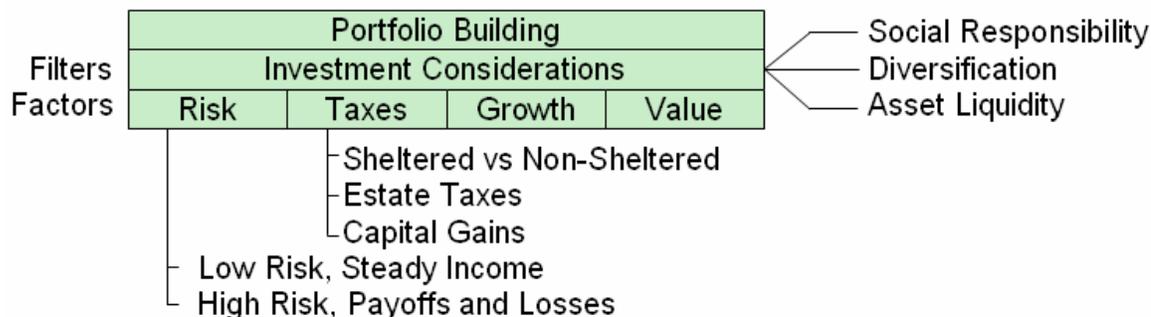


Figure 8.1. A Taxonomy of Quantitative Portfolio Building

From this taxonomy, there are several investment considerations that can be used to filter the number of stocks to choose from. These considerations could involve Social Responsibility, such as the energy consumption or environmental practices of companies, diversification issues where balancing portfolios in a mix of sectors may be a goal or asset liquidity where a high degree of liquidity in investments is needed to manage cash flow or seizing upon opportunities. Once the filters have been applied, there are some

factors to consider; investment risk, the effect of taxes, and growth and value stock selection approaches.

Risk is the dimension of investment that focuses on the risk tolerance of the Investor. This is usually a factor of age and/or investment goals, where low risk can provide a steady income and high risk can lead to high payoffs or losses. Taxes involve the tax consequences of investments and can sometimes play a role in asset selection. The choices between sheltered and non-sheltered investments and the effects of estate and capital gains taxes can all be a factor in stock selection.

While Risk level and Tax consequences are important factors in portfolio building, Growth and Value form two major criterion for stock selection (Fama and French, 1998). Growth looks for above-average returns and a reasonable profit. Stocks under the Growth factor will generally have Earning Per Share (EPS) of greater than 20%. On the other hand, Value looks for out of favor stocks that are currently being ignored by the market. These stocks have low Price to Earnings (P/E), low price to book and sometimes high dividends to attract buyers. Most literature seeks to define distinct differences between Growth and Value, however, stock guru Warren Buffet believes that no such distinction exists (Buffet, 1992).

Both Growth and Value have differing stock selection strategies. One of the more common approaches is the Securities Market Line (SML) approach which seeks to balance portfolio risk and return through a mathematical process of stock selection. A variety of techniques have used the SML approach and differ mainly in the exact mathematics used or external factors evaluated. These strategies include Modern

Portfolio Theory (MPT), Capital Asset Pricing Model (CAPM) and Arbitrage Pricing Theory (APT).

In MPT, assets are priced by balancing the potential risk and return of adding one more stock to an existing portfolio (Markowitz, 1952). CAPM prices assets by discounting future cash flows (Sharpe, 1964) and APT prices assets by a function of risk and macro-economic factors (Ross, 1976). APT further requires perfect competition which may not be an ideal choice for real-world application.

There are other selection strategies that do not use the SML approach. These include Growth at a Reasonable Price (GARP), Intrinsic Value and Momentum/Contrarian strategies. GARP was pioneered by Peter Lynch and balances Growth and Value investment goals. In this strategy, securities are selected based on good growth (e.g., $EPS > 20\%$) and are not overpriced (e.g., $P/E < EPS$ growth rates). Intrinsic Value, which was pioneered by Warren Buffet, is a fundamental analysis of stocks. This strategy uses the fundamentals of a stock to derive an 'intrinsic value.' Stock selection then encompasses buying stocks that have an intrinsic value lower than their current market price. In Momentum/Contrarian strategies, stocks are selected based upon recent returns, with the expectation that past winning stocks will continue to win and that past losers will continue to lose. This strategy is easy to implement, has been widely studied and is susceptible to movement from financial news articles.

In the Momentum/Contrarian strategies, there is a portfolio formation period of f where returns are analyzed and stocks are selected, and a holding period h where the stocks are generating their returns. The first step involves analyzing the stock returns

within period f . Step two is a rank ordering of these returns. Then finally selecting the top fraction of stocks for period h is a Momentum strategy while selecting the bottom fraction of stocks is a Contrarian approach. From prior literature, it was found that past winning stocks will continue to outperform in intermediate-term horizons of 3-12 months (Jegadeesh and Titman, 1993), while past losing stocks will turn-around and outperform in short-term horizons of weeks or months (Jegadeesh, 1990; Lehmann, 1990) and longer-term horizons of 3-5 years (De Bondt and Thaler, 1985; De Bondt and Thaler, 1987).

In order to implement these strategies, a determination of the length of the formation period and the exact threshold of stocks to use, must be determined. In selecting the period of f , there have been several implementations. One of which is to assign $f = h - 1$ (Gervais, Kaniel et al., 2001), or make period f and h equivalent (Conrad and Kaul, 1998), or even assign f to be a series of values less than or equal to h (Kang, Liu et al., 2002), where the latter implementation provides a more robust model of strategies.

As for determining the top/bottom fractional cutoff for portfolio selection, there have also been several different instantiations. One is to assign the cutoff at 10% of stocks (Chan, Jegadeesh et al., 1996), or the more typically used 20% threshold (Lo and MacKinlay, 1990; Jegadeesh and Titman, 1993; Kang, Liu et al., 2002), and even 33% of stocks (Chan, 2003).

While Momentum and Contrarian strategies are generally linked to price, there are several approaches other than price. The first of which is earnings, where the momentum of earnings surprises can have an effect on current estimates (Chan, Jegadeesh et al.,

1996). Second, in using a dividend contrarian strategy, Asness found a strong relation between dividend yield and the contrarian returns (Asness, 1997). Rouwenhorst studied the effects of market capitalization and found that larger companies were more prone to Momentum because of their market exposure (Rouwenhorst, 1998). In terms of Volume, Gervais et. al., found that stocks with unusually high or low trading volumes outperform those with normal trading volumes (Gervais, Kaniel et al., 2001). However, using price returns is the most common usage of these strategies.

The reason behind the success of Momentum and Contrarian strategies is mainly because of investor's under or overreaction to news events. Investors tend to overweight recent information (overreaction) and underweight past information (underreaction) (De Bondt and Thaler, 1985). This overreaction was found to be a result of price shocks which cause excess trading volume and volatility (Chan and Franklin, 2003). Underreaction is when investors do not react quickly enough to a news event (Forner and Marhuenda, 2003). However, the reaction times are generally measured in terms of weeks, months and years following a news event in which price drift is evident and not in terms of minutes following a news release.

From this study of prior research, we identified several gaps which we plan to investigate. While Momentum and Contrarian strategies are well-understood in horizons of weeks or longer, short-term linkage to news event releases has not been studied. Will investors overreact to news events within the minutes following the release of a news article? We can indirectly measure this by applying basic quantitative strategies to an

existing financial news prediction system and predict where the stock should be according to the news article terms.

8.3 Proposed Approaches

From these gaps, we have formulated several research questions. The first of which is:

- What is the effect of combining Quantitative strategies with a financial news article prediction system?

While we know from the literature that both Quantitative and financial news article systems provide superior price predictions, combining the two would provide additional insight into AZFinText's predictive mechanism.

As a follow-up to this question, we also ask:

- Should a combined system use all financial news articles or only the articles of companies selected by the Quantitative strategy?

In order to properly evaluate our research questions, we designed the AZFinText system. Figure 8.2 illustrates the AZFinText system design.

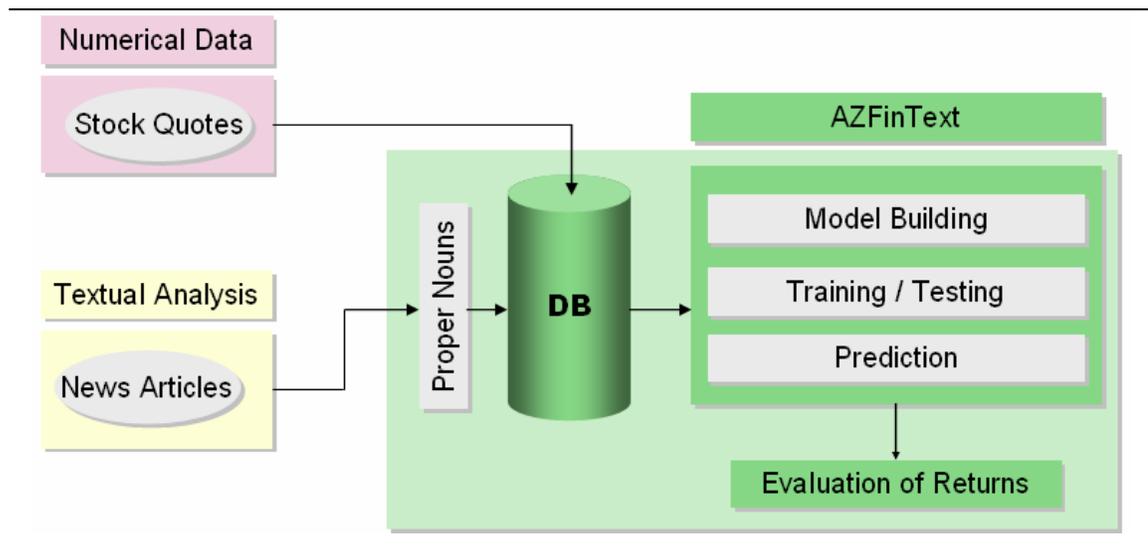


Figure 8.2. AZFinText system design

From the AZFinText system design in Figure 8.2, there are several major components to describe in detail. The first component is Numerical Data which gathers stock price data in one minute increments from a commercially available stock price database. The second component is Textual Analysis. This component gathers financial news articles from Yahoo Finance and represents them by their proper nouns. This module further limits extracted features to three or more occurrences in any document, which cuts down the noise from rarely used terms (Joachims, 1998).

Once this data has been gathered, AZFinText makes its predictions on each financial news article. From prior empirical testing, we found that including the proper noun representations and the stock price at the time the news article was released, provided AZFinText with superior predictive performance compared to other textual representations and different pieces of price information (Schumaker and Chen, 2006).

Within the Model Building stage of AZFinText, we partitioned the data gathered in order to best answer our research questions. In the cases of the Quantitative strategies, this involved buying or shorting those stocks within the quantitative portfolio; Momentum and Contrarian.

For the machine learning algorithm we chose to implement the SVR Sequential Minimal Optimization (Platt, 1999) function through Weka (Witten and Eibe, 2005). This function allows for discrete numeric prediction instead of classification. We selected a linear kernel and ten-fold cross validation. A similar prediction method was employed in the forecasting of futures contracts (Tay and Cao, 2001).

AZFinText is then trained on the data and issues price predictions for each financial news article encountered. Evaluations are then made regarding the effect of stock returns in terms of the quantitative models generated.

8.4 Experimental Design

For our experiment, we selected a consecutive period of time to serve as our experimental baseline. We selected a five-week research period of Oct. 26, 2005 to Nov. 28, 2005, which incorporates twenty-three trading days. The five-week period of study was selected because it gathered a comparable number of articles in comparison to prior studies: 6,602 for Mittermayer (Mittermayer, 2004) and 5,500 for Gidofalvi (Gidofalvi, 2001). We also observe that the five-week period chosen did not have unusual market conditions and would be a good testbed for our evaluation. In order to identify companies with more financial news, we further limited the scope of activity to focus on

companies listed in the S&P 500 as of Oct. 3, 2005. Articles gathered during this period were restricted to occur between the hours of 10:30am and 3:40pm. While trading starts at 9:30am, we felt it important to reduce the impact of overnight news on stock prices and selected a period of one-hour to allow prices to adjust. The 3:40pm cut-off for news articles was selected to disallow any +20 minute stock predictions to occur after market hours. A further constraint was introduced to reduce the effects of confounding variables, where two articles on the same company cannot exist within twenty minutes of each other or both will be discarded. The above processes had filtered the 9,211 candidate news articles gathered during this period to 2,809, where the majority of discarded articles occurred outside of market hours. Similarly, 10,259,042 per-minute stock quotations, 302 analyst recommendations and 3,937 insider transactions were gathered during this period. This large testbed of time-tagged articles and fine-grain stock quotations allow us to perform our evaluation systematically.

For our quantitative portfolio, we followed the work of Kang et. al., and used our study period of Oct. 26 to Nov. 28 as our holding period h (Kang, Liu et al., 2002). Immediately preceding this period is the portfolio formation period f , where stock price returns are rank ordered and the top quintile of stocks becomes the Momentum portfolio while the bottom quintile of stocks becomes the Contrarian portfolio. The length of f is varied in 1 week increments up to 5 weeks in order to increase the model's robustness.

To answer our first research question on finding the most profitable strategy in terms of quantitative portfolios and financial news prediction, we created four models to test, as shown in Figure 8.3.

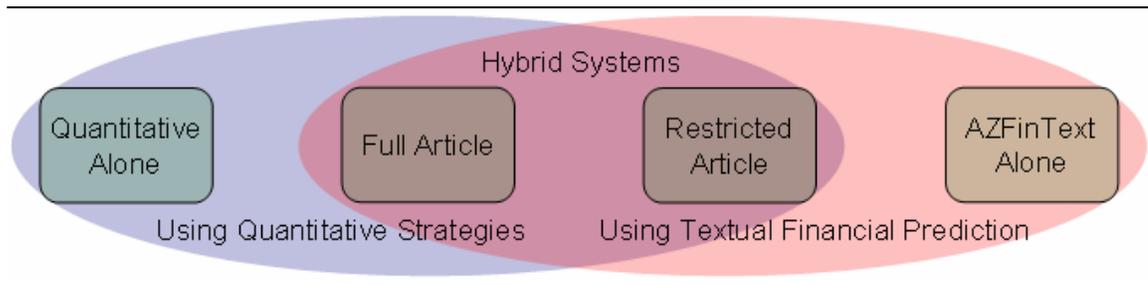


Figure 8.3. Four models to analyze

From this figure, the first model utilizes quantitative strategies by themselves, by measuring the returns of Momentum and Contrarian portfolios during the holding period under differing formation periods. In this model, trading returns are simply the percentage difference between the stock at the beginning and end of period h .

Our second and third models are hybrid systems incorporating both quantitative strategy and financial news prediction. In these models, AZFinText is limited to performing trades on only those companies within the Momentum or Contrarian portfolios. These models differ from one another on the scope of financial news articles available for system training. Model two, or ‘Full Article,’ uses all financial news articles regardless of whether they are in the quantitative portfolio. Model three, or ‘Restricted Article,’ limits AZFinText to only those financial news articles within the portfolio. This differentiation can help determine the value of using financial news articles from peer organizations as a prediction tool.

The fourth model is the financial news prediction system, AZFinText, by itself. This model is free to trade on all of the S&P500 stocks and uses all financial news articles at its disposal.

Prior research has also shown that grouping companies based on their industry sectors leads to better results. One such grouping classification system is Global Industrial Classification System (GICS) which was developed by Morgan Stanley. This classification system was found to be superior to other systems (Bhojraj, Lee et al., 2003) and is also best able to describe analysts' areas of expertise (Boni and Womack, 2004). This method will be repeated on our models using AZFinText, where articles are trained within each GICS Sector and then aggregated together to determine trading returns.

While computing the trading returns of a quantitative-only strategy is straightforward, for the models incorporating AZFinText we utilize a modified version of Lavrenko's Trading Engine (Lavrenko, Schmill et al., 2000a) that examines the percentage return of the stock. When a stock demonstrates an expected movement exceeding 1%, then \$1,000 worth of that stock is then either bought or shorted and then disposed of after twenty minutes. This modified version differs from Lavrenko's original design in regards to the dollar amount of stock bought. We further assume zero transaction costs, consistent with Lavrenko.

An example of our system's operation is shown in Figure 8.4 as well as appendices A and B.

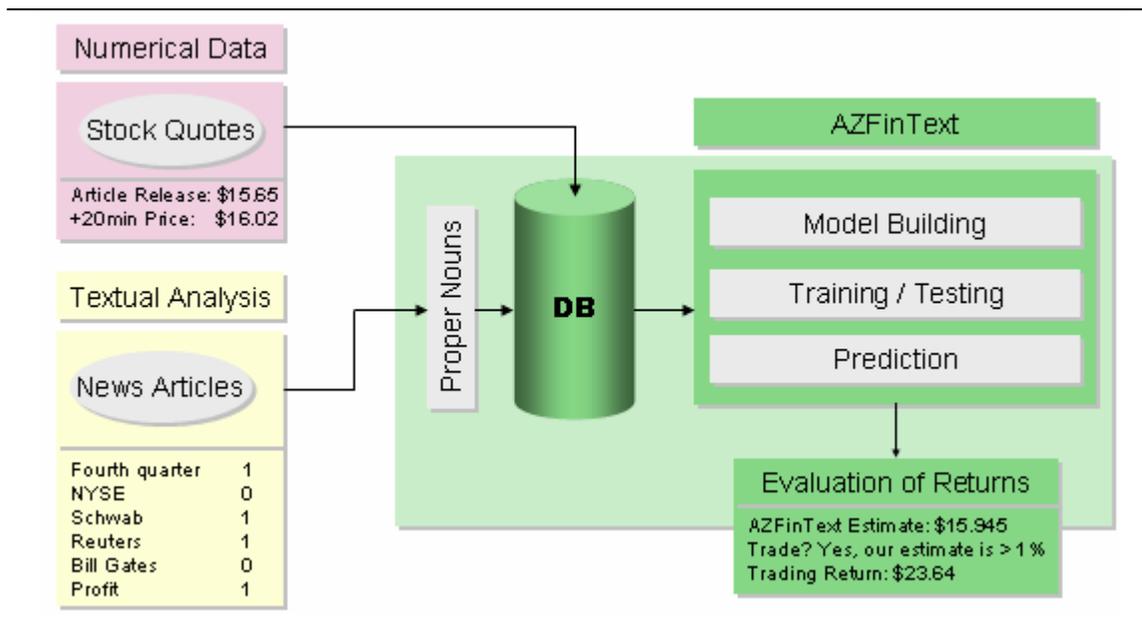


Figure 8.4. Example of AZFinText Processing

The first part of operation is to extract financial news articles. The entire corpus of news articles are represented by its Proper Nouns in binary as to whether or not that particular Proper Noun feature is present in the article and then stored within the database. Concurrently, stock quotations gathered on a per minute basis and are similarly stored. Analyst Recommendations and Insider Transactions are also captured and stored. To build a model, we first pair together the representational Proper Nouns and stock quotation at the time the article was released, for each financial news article. Then, depending upon the particular model that is tested, data is aggregated and passed to our machine learning component for training and testing. Stock price predictions are then made for each financial news article and passed along to the evaluation instruments.

In the example above, AZFinText derived a prediction price of \$15.945 which is greater than 1% of the stock price at the time the article was released, \$15.65. Our

trading engine makes a trade and disposes of it in twenty minutes time, for a trade return of \$23.64 or 2.36%.

8.5 Experimental Findings and Discussion

To answer our first research question of *What is the effect of combining Quantitative strategies with a financial news article prediction system*, we tested four models of varying levels of quantitative and financial news prediction as well as different portfolio formation periods. The results of the Momentum strategies are presented in Table 8.1 and Contrarian is in Table 8.2.

	Trading Returns							
	Strategy Alone		AZFinText Alone		Full Article Training AZFinText		Restricted Article AZFinText	
	n	Return	n	Return	n	Return	n	Return
Momentum Strategies								
1 week formation	92	-5.54%	1998	8.50%	513	20.79%	513	0.33%
2 week formation	92	-3.93%			534	14.53%	534	-2.36%
3 week formation	92	-3.19%			648	12.00%	648	-0.23%
4 week formation	92	-2.69%			541	7.40%	541	1.08%
5 week formation	92	-2.49%			588	6.80%	588	-0.19%

Table 8.1. Trading Returns for Momentum Strategies

In Table 8.1, we are measuring the trading returns of each strategy. As a note to readers, the n's for each strategy reflect the number of trading returns, which in the case of 'Strategy Alone' is also the number of companies in the portfolio. For the remaining three models, n-values are still a measure of the number of trading returns performed, but on a news article-basis. Since we are ultimately evaluating the number of trading returns, whether they are company or article-based, we feel that these values are comparable and can be evaluated.

The first thing to notice is that the 'Full Article' AZFinText hybrid outperformed the Momentum strategy alone in all five portfolio formation periods (20.79% to -5.54%, etc.), with p-values < 0.05 . While the Momentum strategies by themselves are successful in the 3-12 month horizon, they are not successful by themselves in the short-term, five week period of our study, as evidenced by the negative trading returns for each of the five portfolio formation periods.

The second item of interest, is that 'Full Article' AZFinText outperformed 'Restricted Article' AZFinText for all five portfolio formation periods (20.79% to 0.33%, etc.), p-values < 0.05 . It would appear that AZFinText needs a breadth of articles from peer companies not in its portfolio, to make better predictions.

Third, 'Full Article' AZFinText steadily loses returns with additional portfolio formation time (20.79% to 14.53%, etc.), p-values < 0.05 . This result most likely capitalizes on short-term investor overreaction to news articles while the company is still in the headlines and consequently in our portfolio. These results were not totally unexpected as Schiereck et. al. found similar decreasing returns with increasing portfolio formation time (Schiereck, DeBondt et al., 1999).

Fourth, in comparing 'Full Article' AZFinText to 'AZFinText Alone,' the hybrid system had better returns for portfolio formation periods of three weeks or less (20.79% to 8.50%, etc), p-values < 0.05 . It seems that the incorporation of a Momentum strategy into AZFinText helped it to achieve higher returns than would be ordinarily possible.

We present the results of the Contrarian strategy in Table 8.2.

	Trading Returns							
	Strategy Alone		AZFinText alone		Full Article Training AZFinText		Restricted Article AZFinText	
	n	Return	n	Return	n	Return	n	Return
Contrarian Strategies								
1 week formation	92	3.36%	1998	8.50%	505	4.54%	505	0.37%
2 week formation	92	1.86%			541	13.18%	541	0.60%
3 week formation	92	1.40%			432	4.87%	432	1.44%
4 week formation	92	0.87%			497	11.65%	497	1.91%
5 week formation	92	0.29%			490	11.94%	490	-0.08%

Table 8.2. Trading Returns for Contrarian Strategies

From this table, the first thing to note is that Contrarian ‘Strategy Alone’ outperformed Momentum ‘Strategy Alone’ for all five portfolio formation periods. Prior studies noted that Contrarian strategies by themselves are successful in the short-term (weekly, monthly) and long-term (3-5 year) horizons. However, Contrarian’s best success of a 3.36% return was overshadowed by an overall market return of 5.62%.

Second, ‘Full Article’ AZFinText outperformed the Contrarian ‘Strategy Alone’ for all five formation periods (4.54% to 3.36%, etc.), p-values < 0.05. This result was similarly observed in Momentum strategies and would imply that the addition of financial news article prediction has helped the system achieve better returns.

Third, ‘Full Article’ again performed better than ‘Restricted Article’ for all five portfolio formation periods (4.54% to 0.37%, etc.), p-values < 0.05. Again it would appear that AZFinText needs a breadth of articles from companies not in its portfolio to make better predictions.

The Contrarian ‘Full Article’ AZFinText did not perform as well against ‘AZFinText Alone’ (4.54% to 8.50%), p-value < 0.05. While the 2, 4 and 5 week portfolio formation returns of 13.18%, 11.65% and 11.94% performed better than the 8.50% of ‘AZFinText

Alone,’ there was no consistency in trading returns as a function of portfolio formation periods, as was observed in Momentum.

In both Momentum and Contrarian, ‘Full Article’ AZFinText performed the best. However, in comparing the two against one another, Momentum appeared stronger with sizably larger returns. This may be from capitalizing on investor overreaction to news on companies that have had recent success. Investors bid the stock price higher than it should be and AZFinText makes a profit on it.

8.6 Conclusions

From our investigations we found several valuable results. The first of which is that using a Momentum strategy coupled with ‘Full Article’ training of AZFinText led to the best trading returns. While the ‘Contrarian Alone’ strategy outperformed the ‘Momentum Alone’ strategy in all five portfolio formation periods, coupling the financial news prediction system of AZFinText to Momentum led to better overall performance on our five-week dataset. Furthermore, the one-week portfolio formation time led to the best trading returns of 20.79% and most likely capitalizes on the short-term overreaction to news articles while the company is still in the headlines.

We would also suggest several future directions for this stream of research. The first of which is to investigate the linguistic weighting of trained article terms. Within the machine learning algorithm, certain article terms will be weighted higher than others, indicating their prominence in predicting future prices. Investigating the impact of the most important terms will provide insights into future tuning activities.

Second, an investigation into the type of financial news article being used would be essential. Perhaps certain classes of news articles such as acquisitions, changes in directorships, etc. may lead to trader sensitivity and further overreaction. Some article types may not provide additional predictive capacity and may be omitted from prediction activity.

Third, the roles of market exchanges and market capitalization on the predictive abilities of a financial news system is not known. Perhaps traders are more likely to pursue large-cap stocks on particular exchanges and open profit-taking opportunities on the lesser traded stocks. Analyzing these features may lead to better predictability and a better understanding of how a financial news article system beats the market.

CHAPTER 9: CONCLUSIONS AND FUTURE DIRECTIONS

The work presented in this dissertation highlights several methods of acquiring, delivering and making predictions from textual Web documents within diverse domains. Our experiments have shown that textual Web knowledge can be leveraged as a viable source for competitive advantage. Furthermore, this knowledge can also be accurately represented, returned to the environment and built upon, all within the confines of the Data-Information-Knowledge-Wisdom (DIKW) framework. Overall, this dissertation has outlined some of the domain-specific requirements needed for an effective and efficient acquisition, delivery and prediction mechanism. This work can be expanded over the next several years by looking further into the application of other techniques to the acquisition and delivery needs of other knowledge domains as well as tweaking existing methods in terms of knowledge prediction.

9.1 Contributions

Throughout this dissertation, we have made several contributions which can be found useful to future researchers and knowledge organizations. Each of them will be discussed separately in terms of theoretical, technical and empirical contribution.

9.1.1 Theoretical Contributions

This dissertation has made several theoretical contributions. The first of which is our contribution of acquisition, delivery and prediction methods to overlay the existing DIKW framework. This appending of methods can provide a theoretical basis for developing additional tools and techniques.

The second theoretical contribution is through the building of new taxonomical areas within Quantitative stock selection. Prior Quantitative theory was hinged solely on mathematical and algorithmic formalisms. Our creation of a system that uses linguistic techniques opens new taxonomical areas and will provide a foundation for future investigations into the linguistic aspects of Quantitative analysis.

9.1.2 Technical Contributions

This dissertation has made several technical contributions as well. First is the creation of AZ-ALICE which is a modified version of the freely available ALICE chatterbot program. AZ-ALICE's modifications included the addition of per-user logging capabilities which allow easier access to chat dialogs, and the inclusion of an evaluation component where users can reward/penalize system responses or create/modify existing ones.

Our second major technical contribution is the development of AZFinText. This advancement opens up new avenues of linguistic-financial stock market research by recognizing and overcoming one of the major difficulties within Quantitative systems, namely the inability to incorporate information found within breaking financial news articles into discrete predictions.

9.1.3 Empirical Contributions

We believe that the Natural Language Dialog System (NLDS) chapters of this dissertation have made hard empirical contributions to the Question/Answering community by discovering and systematically evaluating the types of information that

Web user's desire and are willing to give back. From these four studies, the Web user's tendency to seek out definitional types of information across disparate domains can be of benefit to future knowledge builders by allowing them to focus more attention on those areas of knowledge that Web users find more important and less on other areas. In terms of gathering knowledge from Web users, it was found that a majority of this collected knowledge was for increasing the breadth of system responses. This finding can allow knowledge builders to harness the power of Web users to expand existing knowledge bases and add variety to current dialog system responses.

Within the Stock Prediction chapters we have made similar empirical contributions to the linguistic-financial community by showing that discrete predictions can be obtained from financial news articles. Our system was able to capture article terms most likely to contribute to price movement and when coupled with Quantitative stock selection schemes, our system was able to outperform existing stock market techniques.

9.2 Relevance to MIS Research

The decision-making process of incorporating diverse repositories of knowledge and managing it effectively is of paramount interest to decision-makers. While this knowledge may come from scattered knowledge domains, each with unique representational needs; it is necessary to integrate it within a unified framework that is flexible enough to the particular needs of the domain. Thus our findings of efficient and effective methods for leveraging such knowledge-based activities becomes of critical interest.

This research has also been positioned within the Design Science framework (Hevner, March et al., 2004) as a balance between the behavioral and technical aspects of MIS research. The knowledge acquisition and delivery components incorporate human-computer-interaction elements and computer mediated communication; both of which fit more on the behavioral side of this paradigm. On the other extreme, knowledge prediction is a wholly technical exercise. However, both behavioral and technical research combines to propose IT artifacts to extend the existing boundaries of knowledge management and enhance its effectiveness beyond what is already accepted. By following the Design Science guidelines, we have sought to design purposeful and relevant IT Artifacts that have been rigorously evaluated and our contributions made clear. Lastly, we favored to communicate our work within this dissertation body and discipline-specific publications in such a way as to be relevant and useful to business organizations and IT researchers alike.

9.3 Future Directions

There are several research directions in which we envision our work to follow. The knowledge acquisition and delivery techniques utilized by AZ-ALICE can be further improved in terms of efficiency and effectiveness. As noted in several of the chapters, NLDS systems are still lacking in an acceptable breadth of general knowledge and further investigations into the quality-control aspects of automated knowledge gathering would be of significant benefit. Furthermore, extensions into the algorithmic retrieval and

response generation features would also hold promise in creating a more effective Question/Answering system.

Second, the predictive techniques within AZFinText could also be further improved through the careful review of other types of machine learning methods and other financial sub-areas that hold high predictive value. Further work would also include the obvious evolution of such a system to perform real-time analysis and execute immediate trades. The subject of what components an evolved system would need to achieve this goal, remains an open question.

APPENDIX A: AZFINTEXT STATE DIAGRAM

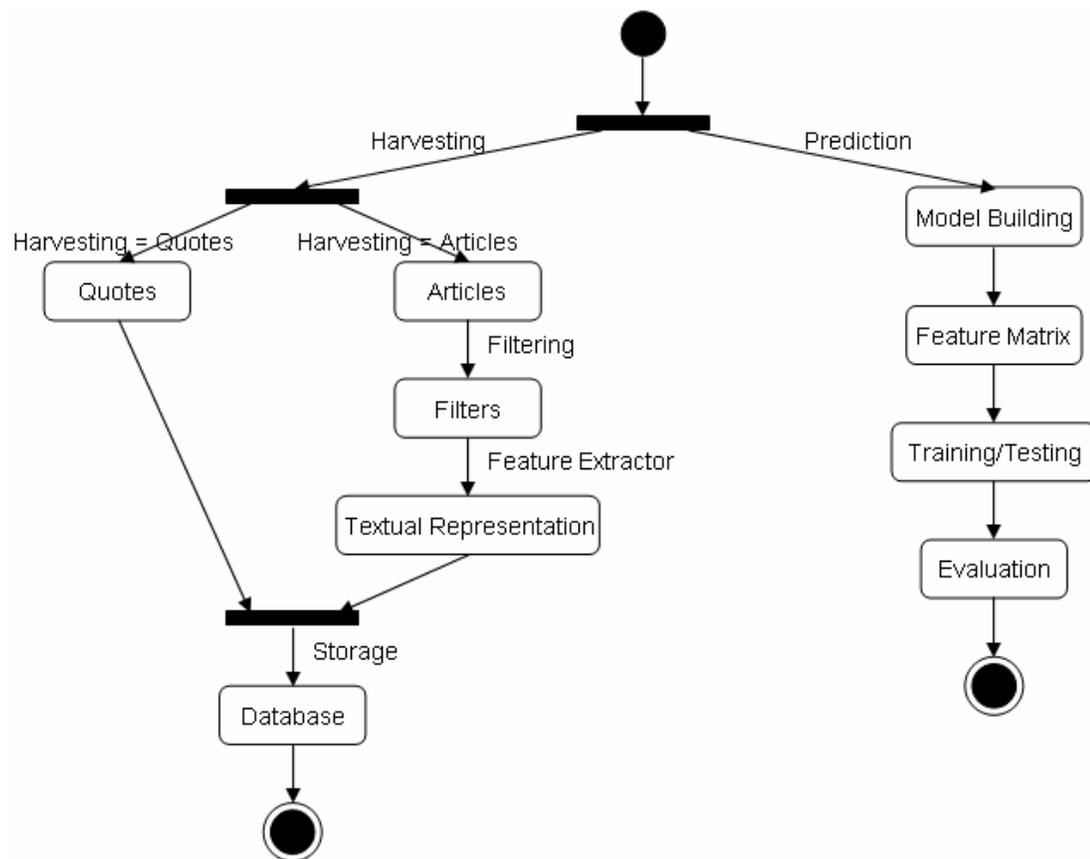


Figure A.1. State Diagram of AZFinText

APPENDIX B: AZFINTEXT CLASS DIAGRAM

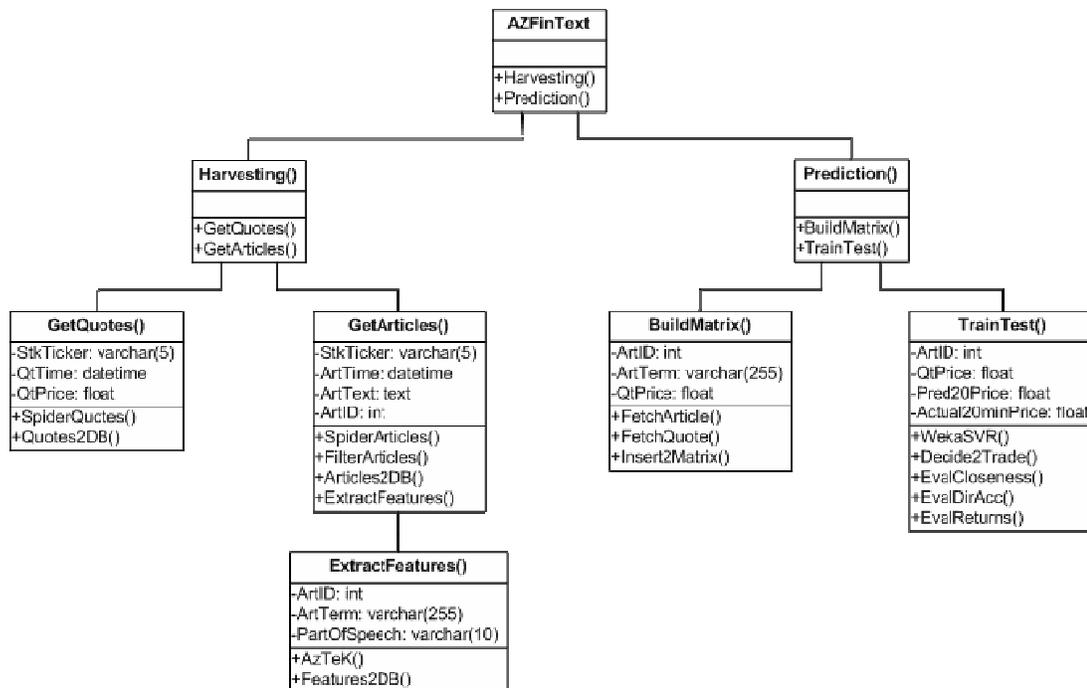


Figure B.1. Simplified Class Diagram of AZFinText

REFERENCES

- Abbink, K. and B. Rockenbach, (2006). Option Pricing by Students and Professional Traders: A Behavioural Investigation. *Managerial and Decision Economics*. 27(6): 497-510.
- Ackerman, M., (1998). Augmenting Organizational Memory: A Field Study of Answer Garden. *ACM Transactions on Information Systems*. 16(3): 203-224.
- Ackerman, M. and T. Malone, (1990). Answer Garden: A Tool for Growing Organizational Memory. Paper presented at the *ACM Conference on Office Information Systems*. Cambridge, MA.
- Ackoff, R., (1989). From Data to Wisdom. *Journal of Applied Systems Analysis*. 16: 3-9.
- Alavi, M. and D. E. Leidner, (2001). Review: Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues. *MIS Quarterly*. 25(1): 107-136.
- Alexander, D. and S. Klein, (2006). The Challenge of Preparation for a Chemical, Biological, Radiological or Nuclear Terrorist Attack. *Journal of Postgraduate Medicine*. 52(2): 126-131.
- Andernach, T., (1996). A Machine Learning Approach to the Classification of Dialogue Utterances. Paper presented at the *Second International Conference on New Methods in Language Processing*. Bilkent, Turkey.
- Anonymous, (Dec. 1, 2003). Home Page of The Loebner Prize in Artificial Intelligence. Accessed Apr. 15, 2004 from <http://www.loebner.net/Prizef/loebner-prize.html>
- Ashton, A. H. and R. H. Ashton, (1985). Aggregating Subjective Forecasts: Some Empirical Results. *Management Science*. 31(12): 1499-1508.

- Asness, C., (1997). The Interaction of Value and Momentum Strategies. *Financial Analysts Journal*. 53(2): 29-37.
- Baesens, B., R. Setiono, C. Mues and J. Vanthienen, (2003). Using Neural Network Rule Extraction and Decision Tables for Credit-Risk Evaluation. *Management Science*. 49(3): 312-329.
- Barlas, I., A. Ginart and J. L. Dorrity, (2005). Self-Evolution in Knowledgebases. Paper presented at the *IEEE AutoTestCon*. Orlando, FL.
- Barnett, J., K. Knight, I. Mani and E. Rich, (1990). Knowledge and Natural Language Processing. *Communications of the ACM*. 33(8): 50-71.
- Batchelor, R. and P. Dua, (1995). Forecaster Diversity and the Benefits of Combining Forecasts. *Management Science*. 41(1): 68-75.
- Beer, M. D., R. Hill and A. SixSmith, (2003). Deploying an Agent-based Architecture for the Management of Community Care. Paper presented at the *Second International Joint Conference on Autonomous Agents and Multiagent Systems*. Melbourne, Australia.
- Bhojraj, S., C. M. C. Lee and D. K. Oler, (2003). What's My Line? A Comparison of Industry Classification Schemes for Capital Market Research. *Journal of Accounting Research*. 41(5): 745-774.
- Bierly, P. E., E. H. Kessler and E. W. Christensen, (2000). Organizational Learning, Knowledge and Wisdom. *Journal of Organizational Change Management*. 13(6): 595-618.
- Bishop, C. M. and M. E. Tipping, (2003). Bayesian Regression and Classification. Amsterdam, IOS Press.
- Blattberg, R. and S. Hoch, (1990). Database Models and Managerial Intuition: 50% Model + 50% Manager. *Management Science*. 36(8): 887-899.

- Boisot, M. and A. Canals, (2004). Data, Information and Knowledge: Have We Got it Right? *Journal of Evolutionary Economics*. 14(1): 43-67.
- Boni, L. and K. L. Womack, (2004). Industries, Analysts, and Price Momentum. *Tuck School of Business Working Paper No. 03-12*, Dartmouth College.
- Breese, J. S. and D. Heckerman, (1996). Decision-theoretic Case-based Reasoning. *IEEE Transactions on Systems, Man and Cybernetics, Part A*. 26(6): 838-842.
- Buffet, W., (1992). Letter to Shareholders. Berkshire Hathaway Inc.
- Burke, K., (Sept. 1, 2006). Not the Man, But the Machine. Accessed Sept. 1, 2006 from http://www.registeredrep.com/moneymanagers/finance_not_man_machine/index.html
- Burns, D. and K. Wutkowski, (Nov. 15, 2005). Schwab to Miss Forecast, Fined by NYSE. Accessed Nov. 15, 2005 from http://biz.yahoo.com/rb/051115/financial_schwab.html?.v=3
- Cai, G. and P. Wurman, (2005). Monte Carlo Approximation in Incomplete Information, Sequential Auction Games. *Decision Support Systems*. 39(2): 153-168.
- Camerer, C. and K. Weigelt, (1991). Information Mirages in Experimental Asset Markets. *Journal of Business*. 64(4): 463-493.
- Carlisle, J. P., (2006). Escaping the Veil of Maya - Wisdom and the Organization. Paper presented at the *39th Hawaii International Conference on System Sciences*. Koloa Kauai, HI.
- Cassimatis, N., E. Mueller and P. Winston, (2006). Achieving Human-Level Intelligence through Integrated Systems and Research. *AI Magazine*. 27(2): 12-14.

- Chan, L., N. Jegadeesh and J. Lankonishok, (1996). Momentum Strategies. *The Journal of Finance*. 51(5): 1681-1713.
- Chan, S. W. K. and J. Franklin, (2003). Dynamic Context Generation for Natural Language Understanding: A Multifaceted Knowledge Approach. *IEEE Transactions on Systems, Man and Cybernetics, Part A*. 33(1): 23-41.
- Chan, W., (2003). Stock Price Reaction to News and No-News: Drift and Reversal After Headlines. *Journal of Financial Economics*. 70(2): 223-260.
- Chavez, M., (2006). Artificial Intelligence and Dynamic Design: Adaptive Real Time 3D Characters. Paper presented at the *Information Visualization*. Baltimore, MD.
- Chen, H., (2001). Knowledge Management Systems - A Text Mining Perspective. Tucson, The University of Arizona - Dept of Management Information Systems.
- Chen, K.-Y. and C. Plott, (2002). Information Aggregation Mechanisms: Concept, Design and Implementation for a Sales Forecasting Problem. *Social Science Working Paper No. 1131*, California Institute of Technology.
- Chen, Y., (2005). Information Valuation for Information Lifecycle Management. Paper presented at the *2nd International Conference on Autonomic Computing*. Seattle, WA.
- Cho, V., (1999). Knowledge Discovery from Distributed and Textual Data. Thesis from the *Computer Science Department*, The Hong Kong University of Science and Technology.
- Cho, V., B. Wuthrich and J. Zhang, (1999). Text Processing for Classification. *Journal of Computational Intelligence in Finance*. 7(2).

- Choi, I., J. Jung and M. Sung, (2004). A Framework for the Integration of Knowledge Management and Business Process Management. *International Journal of Innovation and Learning*. 1(4): 399-408.
- Choo, C. W., (1996). The Knowing Organization: How Organizations Use Information to Construct Meaning, Create Knowledge, and Make Decisions. *International Journal of Information Management*. 16(5): 329-340.
- Cleveland, H., (1982). Information as a Resource. *The Futurist*. 16(6): 34-39.
- Colby, K. M., S. Weber and F. D. Hilf, (1971). Artificial Paranoia. *Artificial Intelligence*. 2(1): 1-26.
- Conrad, J. and G. Kaul, (1998). An Anatomy of Trading Strategies. *The Review of Financial Studies*. 11(3): 489-519.
- Conrad, J. G. and J. R. S. Claussen, (2003). Early User - System Interaction for Database Selection in Massive Domain-specific Online Environments. *ACM Transactions on Information Systems*. 21(1): 94-131.
- Cowgill, B., (Sept. 21, 2005). Putting Crowd Wisdom to Work. Accessed Nov. 21, 2006 from <http://googleblog.blogspot.com/2005/09/putting-crowd-wisdom-to-work.html>
- Davenport, T. H. and L. Prusak, (1998). Working Knowledge: How Organizations Manage What They Know. Boston, MA, Harvard Business School Press.
- De Angeli, A., G. I. Johnson and L. Coventry, (2001). The Unfriendly User: Exploring Social Reactions to Chatterbots. Paper presented at the *International Conference on Affective Human Factors Design*. London, England.
- De Bondt, W. and R. H. Thaler, (1985). Does the Stock Market Overreact? *The Journal of Finance*. 40(3): 793-805.

- De Bondt, W. and R. H. Thaler, (1987). Further Evidence on Investor Overreaction and Stock Market Seasonality. *The Journal of Finance*. 42(3): 557-581.
- Drucker, P., (1969). The Age of Discontinuity; Guidelines to Our Changing Society. New York, Harper and Row.
- Durodié, B. and S. Wessely, (2002). Resilience or Panic? The Public and Terrorist Attack. *The Lancet*. 360(9349): 1901-1902.
- Easley, R. F. and R. Tenorio, (2004). Jump Bidding Strategies in Internet Auctions. *Management Science*. 50(10): 1407-1419.
- Fama, E., (1964). The Behavior of Stock Market Prices. *The Journal of Business*. 38(1): 34-106.
- Fama, E. and K. French, (1998). Value versus Growth: The International Evidence. *The Journal of Finance*. 53(6): 1975-1999.
- Feigenbaum, E. A. and H. A. Simon, (1962). Simulation of Human Verbal Learning Behavior. *Communications of the ACM*. 5(4): 223.
- Ferguson, G. and J. F. Allen, (1998). TRIPS: An Integrated Intelligent Problem-Solving Assistant. Paper presented at the *American Association for Artificial Intelligence Conference*. Stanford, CA.
- Ferguson, G., J. F. Allen, B. W. Miller and E. K. Ringger, (1996). The Design and Implementation of the TRAINS-96 System: A Prototype Mixed-Initiative Planning Assistant. Paper presented at the *Third Conference on Artificial Intelligence Planning Systems*. Edinburgh, Scotland.
- Ferret, O., B. Grau, M. Hurault-Plantet, G. Illouz and C. Jacquemin, (2001). Terminological Variants for Document Selection and Question/Answer Matching.

Paper presented at the *Association for Computational Linguistics*. Toulouse, France.

Flycht-Eriksson, A. and A. Jönsson, (2000). Dialogue and Domain Knowledge Management in Dialogue Systems. Paper presented at the *First SIGdial Workshop on Discourse and Dialogue*. Hong Kong.

Forner, C. and J. Marhuenda, (2003). Contrarian and Momentum Strategies in the Spanish Stock Market. *European Financial Management*. 9(1): 67-88.

Fu, L., (1996). Incremental Knowledge Acquisition in Supervised Learning Networks. *IEEE Transactions on Systems, Man and Cybernetics, Part A*. 26(6): 801-809.

Fung, G. P. C., J. X. Yu, X. Yu and W. Lam, (2002). News Sensitive Stock Trend Prediction. Paper presented at the *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Taipei, Taiwan.

Furedi, F., (2004). Heroes of the Hour. *NewScientist*(2446).

Gao, J. B., S. R. Gunn, C. J. Harris and M. Brown, (2002). A Probabilistic Framework for SVM Regression and Error Bar Estimation. *Machine Learning*. 46(1-3): 71-89.

Gervais, S., R. Kaniel and D. Mingelgrin, (2001). The High-Volume Return Premium. *The Journal of Finance*. 56(3): 877-919.

Gidofalvi, G., (2001). Using News Articles to Predict Stock Price Movements. *Department of Computer Science and Engineering*, University of California, San Diego.

Gilboa, I. and D. Schmeidler, (2000). Case-based Knowledge and Induction. *IEEE Transactions on Systems, Man and Cybernetics, Part A*. 30(2): 85-95.

- Ginsburg, M., (2002). The Catacomb Project: Building a User-Centered Portal the Conversational Way. Paper presented at the *Fourth International Workshop on Web Information and Data Management*. McLean, VA.
- Ginsburg, M. and A. Kambil, (1999). Annotate: A Knowledge Management Support System. Paper presented at the *Hawaii International Conference on System Sciences*. Wailea, HI.
- Graesser, A. C., N. K. Person and D. Harter, (2001). Teaching Tactics and Dialog in AutoTutor. *International Journal of Artificial Intelligence in Education*. 12: 257-279.
- Hahn, R. W. and P. C. Tetlock, (2006). A New Approach for Regulating Information Markets. *Journal of Regulatory Economics*. 29(3): 265-281.
- Hammerton, J., M. Osbourne, S. Armstrong and W. Daelemans, (2002). Introduction to Special Issue on Machine Learning Approaches to Shallow Parsing. *Journal of Machine Learning Research*. 2(3): 551-558.
- Han, J. and K. Chang, (2002). Data Mining for Web Intelligence. *Computer*. 35(11): 64-70.
- Han, S. and Y. Kim, (2001). Intelligent Dialogue System for Plane Euclidean Geometry Learning. Paper presented at the *International Conference on Computers in Education*. Seoul, Korea.
- Hassan, M. R. and B. Nath, (2005). Stock Market Forecasting Using Hidden Markov Model: A New Approach. Paper presented at the *Intelligent Systems Design and Applications*. Wroclaw, Poland.
- Hastie, T., R. Tibshirani and J. Friedman, (2001). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. New York, Springer-Verlag.

- Hearst, M., (1999). Untangling Text Data Mining. Paper presented at the *37th Annual Meeting of the Association for Computational Linguistics*. College Park, MD.
- Herhold, S., (2003, Mar. 20). Exchange Taking Bets on Saddam's Ouster. *The Mercury News*.
- Hevner, A. R., S. T. March, J. Park and S. Ram, (2004). Design Science in Information Systems Research. *MIS Quarterly*. 28(1): 75-105.
- Hildreth, P. M. and C. Kimble, (2002). The Duality of Knowledge. *Information Research*. 8(1).
- Hutchens, J. L. and M. D. Alder, (1998). Introducing MegaHAL. Paper presented at the *Conference on Computational Language Learning*. Sydney, Australia.
- Isbell, C. L., M. Kearns, D. Kormann, S. Singh and P. Stone, (2000). Cobot in LambdaMOO: A Social Statistics Agent. Paper presented at the *American Association for Artificial Intelligence*. Austin, TX.
- Jegadeesh, N., (1990). Evidence of Predictable Behavior of Security Returns. *The Journal of Finance*. 45(3): 881-898.
- Jegadeesh, N. and S. Titman, (1993). Returns to Buying Winners and Selling Losers: Implications for Stock Market Efficiency. *The Journal of Finance*. 48(1): 65-91.
- Jelveh, Z., (2006, July 9). How a Computer Knows What Many Managers Don't. *The New York Times*.
- Jia, J., (2002). The Study of the Application of a Keywords-based Chatbot System on the Teaching of Foreign Languages. *Institute for Interdisciplinary Informatics*, University of Augsburg.

- Joachims, T., (1998). Text Categorization with Support Vector Machines: Learning with Many Relevant Features. Paper presented at the *European Conference on Machine Learning*. Chemnitz, Germany.
- Kang, J., M.-H. Liu and S. Ni, (2002). Contrarian and Momentum Strategies in the China Stock Market: 1993-2000. *Pacific-Basin Finance Journal*. 10(3): 243-265.
- Kloptchenko, A., T. Eklund, J. Karlsson, B. Back, H. Vanharanta and A. Visa, (2004). Combining Data and Text Mining Techniques for Analysing Financial Reports. *Intelligent Systems in Accounting, Finance & Management*. 12(1): 29-41.
- Kupiec, J., (1993). MURAX: A Robust Linguistic Approach for Question Answering Using an On-Line Encyclopedia. Paper presented at the *Conference on Research and Development in Information Retrieval*. Pittsburgh, PA.
- Lavrac, N. and I. Mozetic, Eds. (1992). Second Generation Knowledge Acquisition Methods and their Application to Medicine. Deep Models for Medical Knowledge Engineering. Elsevier, New York.
- Lavrenko, V., M. Schmill, D. Lawrie and P. Ogilvie, (2000b). Mining of Concurrent Text and Time Series. Paper presented at the *International Knowledge Discovery and Data Mining*. Boston, MA.
- Lavrenko, V., M. Schmill, D. Lawrie, P. Ogilvie, D. Jensen and J. Allan, (2000a). Language Models for Financial News Recommendation. Paper presented at the *International Conference on Information and Knowledge Management*. Washington, DC.
- LeBaron, B., W. B. Arthur and R. Palmer, (1999). Time Series Properties of an Artificial Stock Market. *Journal of Economic Dynamics and Control*. 23(9-10): 1487-1516.
- Lehmann, B., (1990). Fads, Martingales, and Market Efficiency. *Quarterly Journal of Economics*. 105(1): 1-28.

- Lenat, D., G. Miller and T. Yokoi, (1995). CYC, WordNet, and EDR: Critiques and Responses. *Communications of the ACM*. 38(11): 45-48.
- Lo, A. and C. MacKinlay, (1990). When are Contrarian Profits Due to Stock Market Overreaction? *The Review of Financial Studies*. 3(2): 175-205.
- Louwerse, M., A. Graesser and A. Olney, (2002). Good Computational Manners: Mixed-Initiative Dialog in Conversational Agents. Paper presented at the *American Association for Artificial Intelligence Conference, Fall Symposia*. Stanford, CA.
- Lucchetti, A. and J. Lahart, (2006, Sept. 30). Your Portfolio on AutoPilot; Brokerages Roll Out Software to Automate Trading Strategies; Risks of Becoming a 'Quant'. *Wall Street Journal*. pp. B1.
- Malkiel, B. G., (1973). A Random Walk Down Wall Street. New York, W.W. Norton & Company Ltd.
- Markowitz, H., (1952). Portfolio Selection. *The Journal of Finance*. 7(1): 77-91.
- Mazzotta, G., (1993). Dante's Vision and the Circle of Knowledge. Princeton, NJ, Princeton University Press.
- McDonald, D. M., H. Chen and R. P. Schumaker, (2005). Transforming Open-Source Documents to Terror Networks: The Arizona TerrorNet. Paper presented at the *American Association for Artificial Intelligence Conference Spring Symposia*. Stanford, CA.
- McKevitt, P., D. Partridge and Y. Wilks, (1999). Why Machines Should Analyse Intention in Natural Language Dialogue. *International Journal of Human-Computer Studies*. 51(5): 947-989.
- Mehrotra, S., J. Wickramasuriya, R. Eguchi, C. Huyck, C. Butts, D. V. Kalashnikov, N. Venkatasubramanian, K. Altintas, R. Hariharan, H. Lee, Y. Ma and A. Myers, (2004). CAMAS: A Citizen Awareness System for Crisis Mitigation. Paper

presented at the *ACM SIGMOD International Conference on Management of Data*. Paris, France.

- Mittermayer, M., (2004). Forecasting Intraday Stock Price Trends with Text Mining Techniques. Paper presented at the *Hawaii International Conference on System Sciences*. Kailua-Kona, HI.
- Moldovan, D., M. Pasca, S. Harabagiu and M. Surdeanu, (2003). Performance Issues and Error Analysis in an Open-domain Question Answering system. *ACM Transactions on Information Systems*. 21(2): 133-154.
- Moore, R. and G. Gibbs, (2002). Emile: Using a Chatbot Conversation to Enhance the Learning of Social Theory. *Centre for Learning & Teaching - Sociology, Anthropology and Politics*, University of Huddersfield.
- National Commission on Terrorist Attacks upon the United States, (2004). The 9-11 Commission Report. National Archives and Records Administration.
- National Research Council, (2002). Making the Nation Safer. Washington D.C., The National Academies Press.
- Newell, A., (1973). You Can't Play 20 Questions with Nature and Win: Projective Comments on the Papers of this Symposium, W.G. Chase, Academic Press.
- Pai, P. F. and C. S. Lin, (2005). A Hybrid ARIMA and Support Vector Machines Model in Stock Price Forecasting. *Omega*. 33(6): 497-505.
- Pasca, M. A. and S. M. Harabagiu, (2001). High Performance Question/Answering. Paper presented at the *Annual ACM Conference on Research and Development in Information Retrieval*. New Orleans, LA.
- Pejtersen, A. M., (1998). Semantic Information Retrieval. *Communications of the ACM*. 41(4): 90-92.

- Platt, J. C., (1999). *Fast Training of Support Vector Machines using Sequential Minimal Optimization* in B. Scholkopf, C. Burges and A. Smola, Advances in Kernel Methods: Support Vector Learning. Cambridge, MA, MIT Press. 185-208.
- Polanyi, M., (1997). *The Tacit Dimension* in L. Prusak, Knowledge in Organizations. Newton, MA, Butterworth-Heinemann. 135-146.
- Potter, S., A Survey of Knowledge Acquisition from Natural Language. Accessed Nov. 1, 2002 from <http://www.aiia.ed.ac.uk/project/akt/work/stephenp/TMA%20of%20KAfromNL.pdf>
- Raberto, M., S. Cincotti, S. M. Focardi and M. Marchesi, (2001). Agent-based Simulation of a Financial Market. *Physica A: Statistical Mechanics and its Applications*. 299(1-2): 319-327.
- Raberto, M., S. Cincotti, S. M. Focardi and M. Marchesi, (2003). Traders' Long-run Wealth in an Artificial Financial Market. *Computational Economics*. 22(2): 255-272.
- Radv, D., W. Fan, H. Qi, H. Wu and A. Grewal, (2005). Probabilistic Question Answering on the Web. *Journal of the American Society for Information Science and Technology*. 56(6): 571-583.
- Reeves, B. and C. Nass, (1996). The Media Equation: How People Treat Computers, Televisions and New Media Like Real People and Places, Cambridge University Press.
- Ross, S., (1976). The Arbitrage Theory of Capital Asset Pricing. *Journal of Economic Theory*. 13(3): 341-360.
- Rouwenhorst, K., (1998). International Momentum Strategies. *The Journal of Finance*. 53(1): 267-284.

- Russell, R. S., (2002). Language Use, Personality and True Conversational Interfaces. *Artificial Intelligence and Computer Science Project Report*, Univ of Edinburgh.
- Schiereck, D., W. DeBondt and M. Weber, (1999). Contrarian and Momentum Strategies in Germany. *Financial Analysts Journal*. 55(6): 104-116.
- Schumaker, R. P. and H. Chen, (2006). Textual Analysis of Stock Market Prediction Using Financial News Articles. Paper presented at the *Americas Conference on Information Systems*. Acapulco, Mexico.
- Schumaker, R. P. and H. Chen, (2007). Leveraging Question Answer Technology to Address Terrorism Inquiry. *Decision Support Systems*. Forthcoming.
- Schumaker, R. P., M. Ginsburg, H. Chen and Y. Liu, (2007). An Evaluation of the Chat and Knowledge Delivery Components of a Low-Level Dialog System: The AZ-ALICE Experiment. *Decision Support Systems*. Forthcoming.
- Schumaker, R. P., Y. Liu, M. Ginsburg and H. Chen, (2006). Evaluating Mass Knowledge Acquisition Using the ALICE Chatterbot: The AZ-ALICE Dialog System. *International Journal of Human-Computer Studies*. 64(11): 1132-1140.
- Sekine, S. and C. Nobata, (2004). Definition, Dictionaries and Tagger for Extended Named Entity Hierarchy. Paper presented at the *Language Resources and Evaluation Conference*. Lisbon, Portugal.
- Seo, Y.-W., J. Giampapa and K. Sycara, (2002). Text Classification for Intelligent Portfolio Management. *Technical Report CMU-RI-TR-02-14*, Carnegie Mellon University.
- Sharpe, W., (1964). Capital Asset Prices: A Theory of Market Equilibrium Under Conditions of Risk. *The Journal of Finance*. 19(3): 425-442.

- Silver, M. S., (1990). Decision Support Systems: Directed and Nondirected Change. *Information Systems Research*. 1(1): 47-70.
- Simmons, R. F., (1970). Natural Language Question Answering Systems: 1969. *Communications of the ACM*. 13(1): 15-30.
- Smith, R. W., (1998). An Evaluation of Strategies for Selectively Verifying Utterance Meanings in Spoken Natural Language Dialog. *International Journal of Human-Computer Studies*. 48(5): 627-647.
- Spann, M. and B. Skiera, (2003). Internet-Based Virtual Stock Markets for Business Forecasting. *Management Science*. 49(10): 1310-1326.
- Stauffer, D., (2000). EconoPhysics - A New Area for Computational Statistical Physics? *International Journal of Modern Physics C*. 11(6): 1081-1087.
- Stenmark, D., (2002). Information vs. Knowledge: The Role of Intranets in Knowledge Management. Paper presented at the *35th Hawaii International Conference on System Sciences*. Waikoloa, HI.
- Tam, K. Y. and M. Y. Kiang, (1992). Managerial Applications of Neural Networks: The Case of Bank Failure Predictions. *Management Science*. 38(7): 926-947.
- Tay, F. and L. Cao, (2001). Application of Support Vector Machines in Financial Time Series Forecasting. *Omega*. 29: 309-317.
- Technical Analysis, The Trader's Glossary of Technical Terms and Topics. Accessed Mar. 15, 2005 from <http://www.traders.com>
- Thomas, J. D. and K. Sycara, (2002). Integrating Genetic Algorithms and Text Learning for Financial Prediction. Paper presented at the *Genetic and Evolutionary Computation Conference*. Las Vegas, NV.

- Tolle, K. M. and H. Chen, (2000). Comparing Noun Phrasing Techniques for Use with Medical Digital Library Tools. *JASIS*. 51(4): 352-370.
- Torngren, G. and H. Montgomery, (2004). Worse than Chance? Performance and Confidence Among Professionals and Laypeople in the Stock Market. *The Journal of Behavioral Finance*. 5(3): 148-153.
- Torre, N. and A. Rudd, (2004). The Portfolio Management Problem of Individual Investors: A Quantitative Perspective. *The Journal of Wealth Management*. 7(1): 56-64.
- Tuomi, I., (1999). Data is More than Knowledge: Implications of the Reversed Knowledge Hierarchy for Knowledge Management and Organizational Memory. *Journal of Management Information Systems*. 16(3): 107-121.
- Uzuner, O., R. Davis and B. Katz, (2004). Using Empirical Methods for Evaluating Expression and Content Similarity. Paper presented at the *Hawaii International Conference on System Sciences*. Waikoloa, HI.
- Vanschoenwinkel, B., (2003). A Discrete Kernel Approach to Support Vector Machine Learning in Language Independent Named Entity Recognition. *Computational Modeling Lab*, Vrije Universiteit, Brussel.
- Vapnik, V., (1995). The Nature of Statistical Learning Theory. New York, Springer.
- Voorhees, E. M., (2001). Overview of the TREC 2001 Question Answering Track. Paper presented at the *Text REtrieval Conference*. Gaithersburg, MD.
- Voorhees, E. M., (2003). Overview of the TREC 2003 Question Answering Track. Paper presented at the *Text REtrieval Conference*. Gaithersburg, MD.
- Voorhees, E. M. and D. M. Tice, (2000). Building a Question Answering Test Collection. Paper presented at the *Research and Development in Information Retrieval*. Athens, Greece.

- Vrajitoru, D., (2003). Evolutionary Sentence Building for Chatterbots. Paper presented at the *Genetic and Evolutionary Computation Conference*. Chicago, IL.
- Wagner, C., (2004). WIKI: A Technology for Conversational Knowledge Management and Group Collaboration. *Communications of the AIS*. 13(3): 265-289.
- Wallace, R. S., The Elements of AIML Style. Accessed March 28, 2003 from <http://www.alicebot.org>
- Wallace, R. S., The Anatomy of A.L.I.C.E. Accessed Feb 4, 2004 from <http://www.alicebot.org/anatomy.html>
- Weizenbaum, J., (1966). Eliza - A Computer Program for the Study of Natural Language Communication Between Man and Machine. *Communications of the ACM*. 9(1): 36-45.
- West, D., S. Dellana and J. Qian, (2005). Neural Network Ensemble Strategies for Financial Decision Applications. *Computers and Operations Research*. 32(10): 2543-2559.
- Wikipedia, (Jan. 15, 2007). Information Retrieval. Accessed Jan. 17, 2007 from http://en.wikipedia.org/wiki/Information_Retrieval
- Winograd, T., (1977). *Five Lectures on Artificial Intelligence* in A. Zampolli, Fundamental Studies in Computer Science. North Holland. 399-520.
- Witten, I. H. and F. Eibe, (2005). Data Mining: Practical Machine Learning Tools and Techniques. San Francisco, Morgan Kaufmann.
- Wolfers, J. and E. Zitzewitz, (2004). Prediction Markets. *Journal of Economic Perspectives*. 18(2): 107-126.

- Woods, W. A., (1977). *Lunar Rocks in Natural English: Explorations in Natural Language Question Answering* in A. Zampolli, Fundamental Studies in Computer Science. North Holland. 521-569.
- Wright, F., (1829). "Religion." *Course of Popular Lectures*. London, Watson. 53-68.
- Wuthrich, B., V. Cho, S. Leung, D. Permunetilleke, K. Sankaran, J. Zhang and W. Lam, (1998). Daily Stock Market Forecast from Textual Web Data. Paper presented at the *IEEE International Conference on Systems, Man, and Cybernetics*. San Diego, CA.
- Yoon, Y. and G. Swales, (1991). Predicting Stock Price Performance: A Neural Network Approach. Paper presented at the *24th Hawaii International Conference on System Sciences*. Waikoloa, HI.
- Zacharski, R., (2004). A Discourse System for Conversational Characters. Paper presented at the *Intelligent Text Processing and Computational Linguistics*. Mexico City, Mexico.
- Zeleny, M., (1987). Management Support Systems: Towards Integrated Knowledge Management. *Human Systems Management*. 7(1): 59-70.
- Zhong, N., J. Liu and Y. Yao, (2002). In Search of the Wisdom Web. *Computer Speech & Language*. 35(11): 27-31.
- Zuckerman, I. and E. Horvitz, (2001). Using Machine Learning Techniques to Interpret WH-questions. Paper presented at the *Association for Computational Linguistics*. Toulouse, France.