3-22-2012

Acquisition Program Problem Detection Using Text Mining Methods

Trevor P. Miller

ACQUISITION PROGRAM PROBLEM DETECTION USING TEXT MINING METHODS

THESIS

Trevor P. Miller, Second Lieutenant, USAF

AFIT/GCA/ENC/12-02

DEPARTMENT OF THE AIR FORCE

AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GCA/ENC/12-02

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the United States Government. This material is declared a work of the United States Government and is not subject to copyright protection in the United States.

AFIT/GCA/ENC/12-02

ACQUISITION PROGRAM PROBLEM DETECTION USING TEXT MINING METHODS

THESIS

Presented to the Faculty

Department of Mathematics and Statistics

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Cost Analysis

Trevor P. Miller

Second Lieutenant, USAF

March, 2012

AFIT/GCA/ENC/12-02

ACQUISITION PROGRAM PROBLEM DETECTION USING TEXT MINING METHODS

Trevor P. Miller

Second Lieutenant, USAF

Approved:

_________________________	____________
Dr. Edward D. White (Chairman)	Date
________________________	____________
Maj. Jonathan Ritschel (Member)	Date
_________________________	____________
Mr. James S. Okolica (Member)	Date

AFIT/GCA/ENC/12-02

Abstract

This research provides program analysts and Department of Defense leadership with an approach to identify problems in real-time for acquisition contracts. Specifically, we test the abilities of statistical algorithms using text mining techniques to detect unusual changes in acquisition programs' cost estimates at the completion of the programs. Currently, the government purchases monthly written reports, an informational tool on status of an acquisition program, but has not been integrated into problem prediction analysis. We center our research on the following two questions: First, can we quantify the qualitative written reports? Second, can we use these quantifications of the texts to predict cost growths in acquisition programs? Through using text mining techniques, we validate the worth of the written reports by creating algorithms that identify 80% percent of problems in acquisition programs, while increasing the probability of a problem existing given our algorithm detects by 56% from the current methods. These positive results for this analysis provide program offices with a method to detect potential problems in acquisition contracts; furthermore, this research helps the government more efficiently manage their resources as well as reduce cost and schedule overruns.

AFIT/GCA/ENC/12-02

I dedicate these pages to my beautiful wife and to my Lord Jesus Christ. Without their support, I would never have completed this endeavor.

Acknowledgments

Thank you to my Lord Jesus Christ for all of the blessing I have been given and for caring for my wife and me. Furthermore, without the loving support of my wife, who continually encouraged me, I would not have been able to accomplish what I have done. I would like to thank her for everything she has provided me and dedicate this thesis to her.

I express sincere appreciation to my thesis advisor, Dr. Edward White, who spent a substantial portion of his time in assisting me through the thesis process. Moreover, Thank you to my committee members Maj. Jonathan Ritschel and Mr. James Okolica, who invested their time and efforts in helping me accomplish my academic and thesis efforts. I thank Mr. Gilbert Peterson and Mr. Eric Unger for their help with the process.

I thank my classmates for helping me through my first assignment. Their continual support and friendship helped me through this process. I have spent a fun and interesting 18 months with them and will never forget it.

Finally, I thank TEAM 17 for providing me sanity through my classes and the AFIT experience.

Trevor Miller

Table of Contents

Page

Page

List of Figures

Page

List of Tables

ACQUISITION PROGRAM PROBLEM DETECTION USING TEXT MINING METHODS

I: Introduction

With the United States government attempting to decrease the amount of spending in the yearly budget, currently consisting of $3.65 trillion, the Department of Defense (DoD) has been asked to decrease its annual budget by $450 billion over the next ten years in the Budget Control Act of 2011 (Khan, 2011). Because the DoD comprises about 19% of the annual budget, the DoD holds a large share of the responsibility in reducing its budget (The White House, 2011). This substantial decrease in the annual budget causes a closer inspection of every dollar spent in the military. With acquisition projects accounting for nearly 35% of the DoD budget, government and military leaders increasingly care how effectively the money is spent on these projects (The Secretary of Defense, 2009).

Acquisition programs for the military can easily cost hundreds of millions if not billions of dollars. Historical analysis shows the average DoD acquisition program overruns the original estimate by 46% and programs like the Space Based Infrared System (SBIRS) go 160% over budget, early detection of potential problems in the program becomes very important to detect; moreover, with an ability to predict these problems, program management and the military as a whole can decrease the overruns with acquisition programs (Younossi & et al., 2007). By detecting and predicting that a problem might be occurring in an acquisition program, rather than reacting, program management can concentrate its resources to adequately attack the issue and fix the problem before it escalates. Furthermore, Christensen (1993) states that acquisition programs rarely recover from overruns that occur beyond the 10% completion point of the project. Therefore, developing a model to identify problems in acquisition programs early may keep a program from becoming un-recoverable and save billions of taxpayer dollars in unnecessary cost overruns.

Every acquisition program of an Acquisition Category (ACAT) 1D produces Contract Performance Reports (CPR) on a monthly basis as a status update for the program managers. These CPRs include monthly data on the dollar amounts spent compared to what has been allocated to be spent. Furthermore, the CPRs include a written portion (Format 5) that documents the occurrences of the past month. Traditional Earned Value Management (EVM) analysis uses the data stating the dollars spent compared to the dollars allocated to spend (Format 1 data) for problem detection. For example, if the amount of dollars spent deviates from the dollars allocated to be spent for a one or more time periods, then an analyst may suspect an issue occurring in the program. Keaton successfully explores the possibility of detecting problems in programs through an algorithm using Format 1 data; however, the results could be improved through improved algorithms or other methods. He uses EVM methods using cumulative Cost Performance Index (CPI) and Schedule Performance Index (SPI) changes in a control chart with bounds of one standard deviation to detect whether a problem will occur in the future of the program (Keaton et al., 2011).

While analysts predominately use Format 1 data, the written portion of CPRs are most often only used to reference something in case of the numbers detect a problem. The government pays defense contractors to submit the written portions of the CPRs; therefore, the written portions must contain information that the government values. While variables derived from the data such as CPI and SPI can provide predictive ability into whether a problem will occur in an acquisition program, the predictive ability of the written portion of the CPR for problem detection is uncharted territory. However, in addition to his EVM Format 1 analysis, Keaton (2011) provides cursory text mining analysis using frequency counts of words associated with certain topics. These frequency counts present interesting trends, but none of Keaton's models to predict increases in the Estimate at Complete (EAC) prove statistically significant (Keaton, 2011b). Therefore, text mining through these written reports may provide useful results in predicting problems in acquisition programs. In order to determine whether text mining processes are of use in the acquisition management field, we seek to answer the following questions:

1. Can we accurately quantify qualitative textual data through text mining methods?
2. Can we determine a relationship between the text mining results and Earned Value data?
3. Can we use the results from text mining methods to predict changes in a contractor's EAC?
4. If we can predict changes, how accurate are these predictions?

In order to answer these questions, we use text mining, an approach that counts words and phrases and uses those counts to determine predictability, to determine whether or not the acquisition program may experience a major problem (5% change in expected costs) in the future. We use these results to provide program analysts and DoD decision makers the ability to utilize a tool that detects a potential problem in the program and take action accordingly. The text mining method adds an additional technique currently not used when analyzing acquisition programs. If we successfully use qualitative data such as the Format 5 to predict EAC changes, we accomplish something never before done in the DoD cost field. Moreover, by employing text mining to detect problems as early as possible, millions of dollars can be saved and proper equipment can reach the warfighter at an earlier time. These changes improve the acquisition process of the DoD and improves the DoD's ability to provide products to the war front.

This research describes previous efforts and techniques that relate to this research in Chapter 2 with the literature review. We also describe the methodology and our processes we use for our research in Chapter 3. Furthermore, we then discuss the results from our methodology and processes in Chapter 4. Finally, in chapter 5, we conclude with a discussion of the implications of the research and ideas for further research.

II: Literature Review

To create a model to predict deviations in the EAC from the written portion of the CPRs for acquisition programs we require text mining techniques. These techniques extract the information from the written portion and are subsequently applied in a robust statistical model. This research bonds these two areas (text mining and EVM) together to aid the decision makers of military acquisition projects. Researchers use text mining and statistical models to help determine solutions to problems; although, researchers have used statistical models much more extensively than text mining. This literature review contains two portions of concentration for analysis on CPRs: text mining and statistical modeling.

Contract Performance Reports

Contractors for DoD programs publish CPRs as a status report of the program. The CPRs include five portions: Format 1-5. Format 1 “provides data to measure cost and schedule performance by summary level Work Breakdown Structure (WBS) elements (Wallender, 1994).” Format 2 provides data that is organized by functional or organizational cost categories (Wallender, 1994). Both Format 1 and 2 data include Budgeted Cost of Work Performed (BCWP), Budgeted Cost of Work Scheduled (BCWS), Actual Cost of Work Performed (ACWP), and EAC. The BCWP, BCWS, ACWP, and EAC provide metrics to determine if the project is lagging in schedule or going over cost. For example, we create CPI and SPI variables via the following equations $\mathrm{CPI} = \frac{\mathrm{ACWP}}{\mathrm{BCWP}}$ and $\mathrm{SPI} = \frac{\mathrm{BCWP}}{\mathrm{BCWS}}$ (Wallender, 1994). If a CPI >1, then a project has spent more than what was budgeted for the amount of work accomplished, while a SPI >1 represents a project that has performed more work than what has been scheduled to be done in a given time frame. Moreover, these metrics provide a quick tool to determine cost and schedule performance of a project.

The DoD Guide to Analysis states that the Format 3 “provides the time-phased performance measurement baseline changes to the contract for the current month and a forecast of the BCWS for future periods. Format 4 contains manpower loading data and includes manpower applicability with different WBS’s EACs. Finally, Format 5 includes “a narrative report used to explain significant cost and schedule variances and other identified contract problems”; moreover, the Format 5 provides the reasoning and descriptions for why the program is performing as it is throughout the different WBS elements (Wallender, 1994).

The DoD currently uses the contractor cost data to compare the actual cost of different WBS elements to the predicted cost of the element; furthermore, the government uses these comparisons to grade the contractor on their performance for future contracts (Wallender, 1994). Moreover, analysts inspect the contractor cost data for indications of major cost overruns or issues that the numbers make apparent from the report. Analysts also use the data to calculate the contractor’s learning curves for their production and to develop cost models for programs cost in “what if” situations by changing the contract quantities (Wallender, 1994). When analyzing Format 5’s, analysts need to read through up to 200 pages of writing to determine whether a problem exists.

The immense amount of reading either does not get done due to time constraints or hampers a reader's ability to understand trends or find issues. This study uses the data from the Format 1 and 5 portions of the contractor cost data provided to the government to expand upon the utility of contractor cost data for the DoD. Some previous research using these Format 1 and 5's provides insight into how to best improve the utility of these documents.

Keaton looks for signals in the Format 1 data to detect future problems in acquisition programs; moreover, he analyzes this data using a control chart for change detection. Using change detection algorithms, Keaton determines if the ACWP strays too far from the planned BCWP and BCWS; for example, Keaton detects if the monthly CPI or SPI fell outside one standard deviation from the mean (Keaton et al., 2011). When Keaton's algorithm detects a deviation from the previous CPI and SPI budgeted metrics, the algorithm signals that the acquisition program contains a potential problem in the future (Keaton, 2011a).

Keaton also did some initial analysis on text mining and found preliminary results that could predict future EAC. Keaton's text mining results from his methods provide a visual trend of topic occurrences throughout the duration of different acquisition programs (Keaton, 2011b). However, when Keaton tests the differences in the frequency counts to predict problems in acquisition programs, the differences show no statistical significance (Keaton, 2011b). While Keaton could not prove predictive ability from the textual analysis, he laid a foundation for the concept. Keaton's analysis and tools provide a stepping stone toward giving decision makers the ability to detect problems with acquisition programs and take action.

Text Mining

Text mining forms a subset of data mining, where instead of searching through numbers for information, software searches through words and text to detect patterns from these text documents. While text mining has been understood in theory since data mining techniques were developed, text mining has been limited by processing power of computers (Carrol, 2005). Because language and words are more complex than numbers, text mining requires more extensive scrubbing and standardizing of data, before the data can be used than with data mining (Feldman, 2007). With the immense growth of technology in the past couple of decades, text mining has become a reality for research.

Text mining analyzes a corpus of text documents and searches for meaningful information or patterns within the texts. The corpus can be either static or dynamic; static collections consist of a consistent set of documents, while dynamic collections update the collection of texts over time (Feldman, 2007). The corpus size varies from as little as hundreds to as large as millions (Feldman 2007). Based upon the amount of structure included in the text, different amount of standardization of the texts needs to be applied to improve the results from the analysis. For instance, comparing a formal report to internet blog posts requires more standardization for analysis than consistent analysis of many different documents of the same formal report type.

Text mining concentrates on four different things in the text: characters, words, terms, and concepts (Feldman, 2007). Characters describe letters, number and symbols within the text, while words describe a combination of characters with a space separating them from the other words. The text mining techniques for characters and words prove to be easier to conduct, but deliver less information and predictive ability. Terms comprise a combination of any two or more words based upon the occurrences within the text and offer more information to the analyst; however, these terms can be limiting in occurrences depending upon the amount of data. Finally, analysts use hybrid categorization methodologies and cross-referencing phrases and words to determine concepts contained in the text, even if specific words or phrases are not included (Feldman, 2007).

For all items in the text, text mining most basically uses counts of these key words or phrases throughout a text document. Detection of a response can be made based upon the changes in the number of times key words or phrases are used throughout the documents. For example, if the amount of times “minor delay” written in a document increases forty percent over that of the previous five months, then this change in the count for that phrase may detect a problem that might occur in the future. While a basic count of occurrences helps provide information on the collection, a process to eliminate fluff words such as “the”, “and”, and “is” allows the program to concentrate the analysis on words that have more meaning.

A newer process called rapid automatic keyword extraction (RAKE) uses these “fluff” or “stop-words” that break up content bearing words and phrases to identify important keywords (word or compilation of words). RAKE takes the separate keywords and uses co-occurrences of the words or phrases within the keywords to analyze the importance and significance of the words or phrases (Berry & Kogan, 2010). The RAKE process identifies and counts the occurrences of the significant words or phrases. While RAKE shows some advantages, the process has not been fully accepted by the text mining community yet and other processes occur much more often, because the process has only recently been published and others have not yet included this method into their practices (Berry & Kogan, 2010).

Latent Semantic Analysis (LSA), Probabilistic Latent Semantic Analysis (PLSA), Latent Dirichlet Allocation (LDA), and Correlated Topic Model (CTM) each have different processes of extracting predictive and useful information from text. LSA, also known as Latent Semantic Indexing, uses a series of three matrices (document eigenvector, eigenvalue, and term eigenvector) to approximate the original document matrix; moreover, the original document matrix can be obtained by multiplying these three matrices together (Lee & et al., 2010). The document eigenvector matrix size is the number of documents (n) by the unique dimensions of the corpus (r). The eigenvalue matrix simply defines the unique dimensions (r x r) and the term eigenvector matrix defines is defined as the number of unique terms by the number of unique dimensions (m x r).

These matrices contain the allocation of terms within the separate topics; however, one limitation of LSA is that the words in one topic have little relation to other topics (Lee & et al., 2010). Moreover, words in one topic cannot occur in other topics;

therefore, if a word has multiple meanings and can be classified under two different topics, LSA can only allocate that word to one topic (Lee & et al., 2010). LSA works best when used on documents with similar writing styles, but has two limitations: the reduction of the document matrix does not use robust probability theory and an adequate number of topics cannot be determined statistically (Lee & et al., 2010).

PLSA offers an expansion on the LSA methodology in order to improve results. PLSA assumes a probability of a document within the corpus to be P(d), a topic to be $P(z|d)$, and a word $P(w|z)$, where d, z, and w stand for document, topic, and word respectively. $P(z|d)$ describes the probability of a topic given a document, while $P(w|z)$ describes the probability of a word given a topic. PLSA expresses the results in terms of probabilities of these three occurrences, which allows PLSA to find general themes or trends in documents easier than LSA (Lee & et al., 2010). PLSA also allows words to occur in different topics, unlike LSA, but does not "fully reflect the generative process at a document level", which LDA incorporates (Lee & et al., 2010).

Blei et al. developed LDA to elaborate upon the applications of LSA and PLSA. LDA determines the number of words in a document by sampling with a Poisson distribution, creates a distribution for the topics using the Dirichlet distribution, and generates topics and words for the topics based upon a documents distribution (Lee & et al., 2010). LDA performs well when analyzing lengthy documents that have multiple topics. Also, LDA includes spatial statistics to provide a relational factor to words that occur near each other often. Similar to LDA, CTM uses a similar process to LDA, but attempts to improve upon the ability to describe the relationship between topics by using a logistic normal distribution rather than the Dirichlet distribution, which is described in greater detail in the methodology chapter (Lee & et al., 2010). We provide a more extensive description of the software used for text mining in this research in the methodology chapter.

Analysts use LSA and LDA to sift through the keyword counts and extract the most predictive ones. LSA searches documents for words used in the same context and identifies trends between the different words and terms in the document (Zelikovitz & Kogan, 2006). Both methods assume that the text is a vector and each term within the text defines a specific location along that vector (Zelikovitz & Kogan, 2006). LSA weights a term based upon how often the term occurs within the document as well as throughout the total number of documents (Zelikovitz & Kogan, 2006). Similarly, LDA ensures that the number of occurrences of one word or phrase does not overshadow the power or significance of other words that do not occur as often in the document (Wilson & Chew, 2010). In other words, LDA adjusts the analysis to allow less common words or phrases to have more influence than without the adjustment. By using a hierarchical Bayesian model, LDA can determine topics and words in the texts that refer to these topics, where words that exist universally to all topics, such as “the” will be filtered out as insignificant (Blei & Lafferty, 2003). Researchers use methods such as LSA and LDA to improve the results depending upon the subject and the text being analyzed.

We use LDA for this research, because LDA offers immense word separation abilities not achieved through LSA or other text mining methods. More specifically, LDA creates topics that have a probability distribution of how often words occur within them. Every time the algorithm goes through the documents, it sorts words based upon the probability distributions of the words given the topics and then re-creates the distributions for the topics. After completing this process many times, depending upon the application, the words given the topics gain a specific distribution that describes the topic; therefore, the topics become well defined and separate themselves from the other topics throughout the texts. This separation occurs by allocating the words into different topics based upon the conditional probability distributions of words given a topic created through the previous sorting iteration; furthermore, by separating the words enough times based upon the previous distribution of how they were sorted, the words eventually separate into their own topics.

For example, a simple description of the LDA process could be described in terms of separating coins. Given millions of coins (pennies, quarters, dimes, nickels, half dollars, and dollar coins), the LDA process initially separates the coins into groups randomly. After the initial separation, LDA creates a distribution of each grouping of coins. One group may contain thirty pennies, five dimes, twelve quarters, two nickels, one half dollar, and no dollar coins, while another group may have a high density of nickels. The next time through the Expectation Maximization (sorting) process, LDA uses the distribution of the coins within that group after the previous iteration to determine which group the coin should go; therefore, LDA will allocate more pennies to the group described and more nickels to another group. After repeating this process a few hundred times, groups dominated by one coin or another will emerge and the groups will clearly represent a certain coin. For a more in depth description of LDA reference Appendix A or read Blei's paper on LDA.

This research uses LDA-SOM (Latent Dirichlet Allocation Self-Organizing Map) software to accomplish the text mining. The Air Force Institute of Technology (AFIT) Center for Cyberspace Research developed this software. The government developed and owns the LDA-SOM software for all uses of it. The Center for Cyberspace Research originally created the software to track written information about new technologies and weapons to determine the threats of these new technologies (Millar & et al., 2009). The Center for Cyberspace Research developed the LDA-SOM software to be able to assess the impacts of these technologies without having personnel read through all of the texts.

The government validated the software by applying the software to two different data sets to evaluate its effectiveness. The LDA-SOM software correctly sorted the texts into the categories already known in these data sets, which provides evidence that the software can sift through text and correctly analyze them (Millar & et al., 2009). The LDA-SOM "method transforms the word histogram representations of documents into topic distributions, reducing dimensionality in the process (Millar & et al., 2009)." We do not use the self organizing map portion of this software for this analysis.

Examples of Text Mining Research

One of the first fields that text mining has been used to aid in research is the medical field. The National Centre for Text Mining, a British organization that concentrates on the advancement and application of text mining, has published a myriad of articles pertaining to the medical field. In one article, Ananiadou (2011) attempts to improve the organization of bacteria classes, because many bacteria have multiple names and descriptions depending upon the location. Ananiadou uses text mining to take text with descriptions of bacteria and their names and categorize the bacteria (Ananiadou & et al., 2011). Riza uses text describing biological activity and the target (gene, protein, cell, or microorganism) of the biological activity to predict and understand the effects of natural substances (Riza, 2011). Similarly, Kolluru uses text mining to automatically extract the microorganisms and habitats (Kolluru & et al., 2011).

Also, Shi Yu (2010) uses text mining in his research of gene prioritization. Text mining proves to be extremely useful in the medical field and with biologists, because it allows researchers to search through vast amounts of data in the form of literature text and extract important information (Shi Yu & et al., 2010). Shi Yu searches through 284,569 biological literature documents using LSA techniques and determines gene prioritization for different diseases (Shi Yu & et al., 2010). Shi Yu finds that including LSA techniques improves his results in his statistical analysis by reducing the number of terms and sorting them on importance given the terms eigenvalue (Shi Yu & et al., 2010).

Text mining has not been limited to the medical field. The marketing and knowledge management fields use text mining to improve their practices as well. Shaw describes in his paper how large amounts of data can be tapped by using text mining that will provide marketers the ability to better reach customers and encourage them to purchase their product (Shaw & et al., 2001). Similar to marketing, text mining has also been used to improve customer relations by sifting through text and determining what customers want most (Sirmakessis, 2004). Social media can also be used as shown by Corley's research in which updates to social media sites such as MySpace® were used to track influenza in different regions of the nation and predict where outbreaks may occur (Corley & et al., 2010).

The convenience store 7-Eleven used text mining to decide where and how to implement an iced coffee product (Bush, 2009). The marketing team text mined social media sites to determine how people thought about their product, what flavors customers wanted and where to implement the different iced beverages (Bush, 2009). Social media sites have been mined to determine customer sentiment toward companies and then applied to trends in the stock market as well; furthermore, it has been suggested that using text mining could allow the government to detect economic trends quicker (Giles, 2011).

RenewData® provides another example of the private sector using these text mining methods (2012). RenewData® provides customers with data backup as well as content analytics (Blayney, 2012). Their content analytics specializes in working with law firms with cases that require a large amount of documents. RenewData uses their own patented and patent pending versions of text mining techniques to determine which of the legal documents will most likely contain pertinent information to the case. By using their text mining analytics, they reduce the amount of lawyer fees by presorting the non-pertinent documents and not having to employ lawyers to read through every single document (Blayney, 2012).

Text mining has proven to help the medical and marketing fields and as suggested can start to help solve problems for the DoD. Since September 11^{th} 2001, the DoD and other government agencies uses text mining as a premier tool to find possible threats in the nation's defense (Metz, 2003). The DoD uses mining techniques for counter-terrorism measures by creating models to predict and detect terrorist activities (Thuraisingham, 2002). Furthermore, the Defense Threat Reduction Agency has used text mining to find viruses and trace them (DTRA, 2012). Also, text mining has been used to track the circumstances of war and the amount of interest in Weapons of Mass Destruction (WMD); therefore, the DoD can better predict the ability and interest of organizations in WMD's (Sallach, 2009).

The research in this paper indicates a different application of text mining for the DoD. The application would be to use text mining on the written portions of the CPRs of major acquisition programs and be able to use the mined information to detect deviations in the progress. Keaton presents this application of text mining for the first time at the Department of Defense Cost Analysis Symposium (DODCAS) in 2011. Keaton's research used QDA Miner® to map the changes of topic occurrences through a document as an indication of text mining ability to predict EAC (QDA Miner®, 2005). Keaton's initial results provide reason to believe that text mining can be successful, but does not include any concrete statistics. This research follows the material Keaton presents and expands upon it.

Statistical Models

When the text mining portion of the analysis completes, the output consists of data containing topic frequencies for each document. These frequencies for the different topics become the variables for the statistical analysis to predict potential problems in the programs. Statistical analysis allows researchers to determine the strength of relationships between variables and the response. Simply, a change in the cost of an acquisition program can be predicted by the frequencies of these topics. This prediction of the change in cost of an acquisition program falls into a subset of statistical modeling called forecasting.

Forecasting allows for users to predict future outcomes with some level of certainty in order to improve their decision making ability. Forecasting uses time series analysis, “a set of measurements, ordered over time, on a particular quantity of interest”, to use past data to predict future outcomes (Newbold & et al., 2010). Past data of the response variable usually predicts the future values of the response variable better than other independent variables. For example, the price of gasoline over the last month predicts what the price of gasoline will be tomorrow better than the price of corn over the last month. However, other variables can aid in predicting future values of a variable as well.

Society uses past data to forecast the future in many different fields and areas. Insurance companies use data to predict the risk of a person based on past data and hedge fund managers use past data to predict what a stock will be in the future. Manufacturing businesses use forecasts to determine how much of an item to produce; furthermore, the

DoD uses forecasts for determining the manning levels across the different branches of the military (Stewart, 2003).

Keaton's (2011) previous research using EVM methods to predict changes in EAC provides another example of DoD cost analysis applications of this method. Keaton tracks the CPI and SPI as well as a combination of the two and whenever these data points fall outside of one standard deviation Keaton's algorithm detects that a deviation of the EAC will occur. Keaton also attempted different bounds other than one standard deviation as well as predicts one to twelve months from the current period. All of these examples demonstrate the importance and power of using forecasts.

However, sometimes system program offices in the government do not use predictive algorithms to determine problems in acquisition programs, but instead rely predominately on program manager's insight and communication between the contractor and the program offices along with traditional EVM analysis. This technique requires less manpower and development of analytic techniques early in the acquisition program, but can cause increase in need of manpower when problems occur. Furthermore, by not using predictive analytics, the programs tend to be reacting to problems that have already occurred rather than predicting them and preventing them from occurring.

This research concentrates in determining whether the written portions of the CPR can be used to aide in predicting the future costs of acquisition programs. Furthermore, this research explores to what extent and with what confidence can the data from the written portion of the CPRs help program managers and decision makers manage their acquisition programs. Having an extra tool to use and monitor the programs may result in cost savings and improved program management for the DoD.

III. Methodology

In order to detect problem occurrences in acquisition programs, this research concentrates on using text mining techniques on CPR Format 5's to generate data that can be used to create a statistical model to predict if problems will occur within the next six months. This research defines a problem as a five percent or more absolute change in the EAC. This chapter describes the analysis of this research to obtain the end goal of aiding the decision maker. The chapter discusses the data source, data limitations, text mining process, and the statistical model. By providing analysis of the written portion of the monthly performance reports to predict an increase in the EAC, this research provides a decision maker a greater ability to manage the program.

Data Sources

The data for this research comes from the Defense Cost and Resource Center (DCARC). DCARC publishes EV data for DoD acquisition contracts, which includes the monthly CPRs used in this research. We use monthly data for ACAT 1D programs in the DCARC database, because these programs have the most oversight and money associated with them; therefore, a useful prediction for these programs provides greater cost savings to the government.

The CPR data on DCARC contained many different formats including: Extensible Markup Language (XML), Hyper Text Markup Language (HTML), Portable Document File (PDF), Text files, and Microsoft Word documents. We converted all of these documents into text files using created visual basic code and PDF converters to achieve text mining software compatibility. The data spans from as early as September, 2007 to as late as August, 2011. The data collected contained consecutive CPR data for 38 programs, including 1,304 total monthly CPRs. A table describing the process of selecting programs for use in this analysis can be seen in Table 1, while a complete list of the programs and the number of monthly reports associated with the corresponding program can be seen in Appendix B. The average Format 5 length in our data set is 58 pages and the longest Format 5 is 215 pages.

Once we collect the data, we reserve 20% of the data for a validation set. We choose the validation data based upon a stratified random sample of the programs. We divide the programs into three categories: large, medium, and small, based upon the number of consecutive months of data the program contains. Small, medium, and large sized programs consist of less than thirty, thirty to forty, and greater than forty consecutive months of data respectively. According to these bins we created, there are 11 small, 10 medium, and 16 large programs as seen in Table 1, which also displays how many programs of each size we select for the validation set. Because the validation set should contain around 20% of your data, we selected approximately 20% of programs from each category. We select eight programs from the stratified random sample to be in the validation set totaling 268 months of data. The validation programs include AMF, CH-53, DDG 1000, EFV, GPS Navstar, MH-60, MPS SEICR, and V-22. While we collect 1,304 data points, not all of the data collected from DCARC could be used for the analysis.

Table 1: Number of Programs and Validation Programs Classified as Small, Medium, and Large

	Small Programs	Medium Programs	Large Programs
Number of Total Programs	11	10	16
Number of Validation Programs	2	3	3

Data Limitations

Not all of the data could be used and some of the data contained complications. Data could not be used, because of unusable data formats and gaps in the data. When we encounter XML files, many of them did not contain any portion of the Format 5. If the XML files did include a portion of the Format 5, the Format 1 data associated with the file could not be recovered for analysis, because we do not have the file that would un-encrypt the XML file. With these cases of data, we do not use the data for the analysis.

Moreover, if a program contains a gap of two months in the data, we determined we could not use the program's data; however, if a one month gap occurred, we decided to use a linear approximation for that month's data. A linear approximation would average the values for the variables the month before and the month after the hole to create a value for that month. For example, if we have the ACWP for January and March, but not for February, we add the ACWP from January and March and divide it by two to calculate the linear approximation for February. Moreover, we also would linearly approximate frequency of topic (bin) occurrence from the January to March document. We linearly approximate 0.77% of the data; therefore, we mitigate the affect of the linearly approximated data.

Some of the programs in the DCARC database did not include Format 5 data or published unusable data throughout the existence of the program. In these cases, we did not use these programs. As seen in Table 2, DCARC contains 118 programs with CPR data, while only 64 of those 118 programs are ACAT 1D programs. Furthermore, due to gaps in the data and un-useable data formats, we eliminate 27 ACAT 1D programs and contain a remaining final data set of 37 ACAT 1D programs.

Table 2: Breakout of Programs for Data Collection

Category	**Number of Programs**
All Programs	118
1D	64
Useable	37

The different combinations of characters throughout the texts prove to be a complication of the data. Text mining software groups characters as words to analyze. The Format 5 includes many numbers or combination of numbers and letters that initially the software analyzes as words. In order for the software to concentrate on meaningful words, we excluded these strings of characters in the software. To exclude these combinations of numbers and letters so that the strings of characters do not get sorted into different topics and diminish the meaning of the topics, we included a list of numbers and letters into an exception list. This exception list excludes strings of characters from being sorted into different topics (bins). For example, we exclude combinations of numbers and characters in the form of dates in the text that would appear like “12JUN2008”; moreover, we exclude other forms of non-meaningful character and letters as well. Other examples of combinations of characters we exclude include numbers from zero to one billion and nonsense combinations as “23lk453jio2”. The text mining software algorithm allocates more weight on the words that occur more frequently, but LDA methods adjust for an abundance of certain words, when they occur often enough to diminish the impact of other words. Furthermore, we use the percentage of time a topic occurs within a document as a predictive variable instead of the frequency, in order to adjust for different lengths of documents between programs.

Text Mining Process

After the data has been collected and standardized through converting the texts to a consistent format, we use text mining software to find trends in the Format 5’s. The text mining software uses LDA to find meaning in the words of the documents. LDA examines every word of every document and creates topic probability distributions from the documents. Essentially, LDA determines the number of words in a document, uses a Dirichlet distribution for the topics within a text, creates the topics, and allocates words within each topic (Lee & et al., 2010). The Dirichlet distribution is a multidimensional Beta distribution, where a Beta distribution can define the maximum and minimum points and take the form of almost any distribution (Ng & Tian, 2011). The software sorts words into a user defined number of topics for allocation.   LDA repeats through all the documents a few hundred times (depending upon the redundancy of the analysis) in order to better define the distribution of words within each topic. We increase the times we repeat the process from the default 200 to 1,000 times in our analysis to ensure the topics become properly defined.

To elaborate on the process, we will expand upon the coin example mentioned in Chapter 2. Imagine sampling the change people have from all over the world and then sorting them into different categories based upon the same process in the earlier example. Because the process includes relationship between words when sorting between bins, British coins would be found in many of the same bins that the European Union coins would be found, because of the high volumes of travel and trade between these nations. Therefore, after sorting and creating distributions and sorting and creating distributions to sort from again, the distributions end up looking not just like a single spike, but more of a conglomeration of spikes of different magnitudes that signify the different words occurring in that bin. The conglomerations of these spikes define the bin and summarize the different coins that exist within the bin as well as their relationships between each other.

For instance, after sorting coins, you may result with a bin predominately consisting in British coinage, with high amounts of Bulgarian and Egyptian coins, and a smattering of United States, Chinese, and Indian coinage as well. This bin of coins could represent many different circumstances such as a British citizen who has been given U.S., Chinese, and Indian coins as gifts from people and has recently taken advantage of the cheap airfare to enjoy the beaches along the Black and Mediterranean Seas. However, the bin could also describe a Bulgarian business owner who uses a more stable currency of the British often and trades with the Egypt, United States, and India. The bins can describe many different circumstances of the relationships between the contents within it; therefore, labeling these bins loses information and would be incorrectly define the bins instead of the bin being defined as the whole of its contents.

After the topic or bins have been defined through the Dirichlet distribution sorting process and the documents contents have been separating into different bins, the software outputs the frequency a topic occurs in each document. By dividing the number of times the topic occurs by the total times any topic occurs in a document, we create a variable that accounts for the percent of time a topic occurs throughout a document (i.e. the probability of a topic given a document). We then create new variables from this percent variable to create other variables to analyze for the most predictive model. For example, we create a variable that is the standard deviation of the last three periods of the percent variable; therefore, if a change in the percent variable occurs, the standard deviation variable will detect it. Moreover, this standard deviation variable takes the standard deviation of the percent of time a topic occurs in January, February, and March when the current time period is March. We create the difference in the percent occurrence from one document to the one previous, the difference in the percent occurrence from a document to the one two previous, and the standard deviation of the percent occurrence during last two and current periods. Therefore, these variables would take the percent of time a topic occurs in March minus the percent of time a topic occurs in January or February minus January for the one month apart variable.

Statistical Model

Using the results for the LDA analysis, we create a statistical model to predict changes in the EAC. We use standard Ordinary Least Squares (OLS) to create the models for this research. We create three models: a model to predict the EAC each four, five, and six months out. We create variables from the frequencies of each topic occurring throughout the documents to predict the changes in EAC. We created different independent variables and analyzed them in JMP® to determine if the variables show prediction tendencies (JMP, 2011). We look at the following variables for significance: percent a topic occurs in a document, change in the percent a topic occurs over one period span, change in the percent a topic occurs over a two period span, and the standard deviation of the last three periods percent of a topic occurring in a document. We decided to use percentages a topic occurs within a document in order to standardize for the different lengths of documents; for example, if document A is four times as long as document B, then document A might have four times as much of a topic than document B, but should still be considered as equal in terms of percentage of occurrence throughout the document.

We attempt different combinations of the number of topics for the text mining software to use. We find that after attempting to separate words into as little as 25 topics to as many as 250 topics that having the text mining software separate the contents into 250 topics provides the most predictive results. Increasing the number of topics past 250 results in too many variables for JMP® to be able to run a single stepwise regression and at 300 topics it would be too much for two separate stepwise regressions to handle. The R^2 values of the different stepwise regressions for the different number of topics can be seen in Figure 1. As seen by the fitted curve in Figure 1, increasing the number of topics only provides marginal returns and improves the results marginally; furthermore, given the computational increase with an increase in the number of topics, we can assume that 250 topics serves as a appropriate number of topics for an accurate predication algorithm.

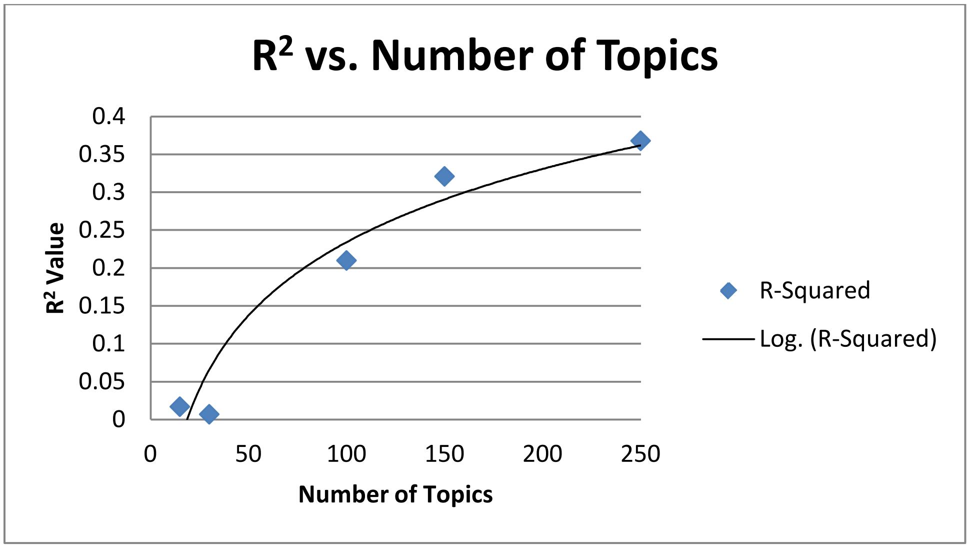


Figure 1: R^2 Values from Stepwise Regression of Variables for Different Topics

The percent a topic occurs in a document contains some of the most predictive variables, but these variables often had issues. Most of these variables only scored highly in one, two or three different programs, which does not allow for that topic to be able to be applied well outside the data set. The change in the percentage a topic occurs across one or two time periods did not have the issue of only including a couple programs; however, these variables do not tend to have as much predictive ability as the standard deviation of the percent a topic occurs variables do. The standard deviation variables dominate all three of the models to predict EAC, because of their predictive ability.

Once we determine the variables for each model, we use Microsoft Excel$^{®}$'s Solver command to change the estimated parameter's coefficients to minimize the Mean Absolute Percent Error (MAPE) of the model. We calculate the MAPE with the following equation $\sum_{\mathrm{i=1}}^{\mathrm{n}} \frac{|\mathrm{Y_{Actual}} - \mathrm{Y_{Predicted}}|}{Y_{\mathrm{Actual}}}$, where n is the number of predictions. By minimizing for MAPE as opposed to the Sum of Squared Error (SSE), the average prediction value will be closer to the average actual result. We determine that having a more accurate point estimate provides the decision makers better information.

First we use OLS to determine our significant variables and then we optimize the variables beta coefficients to minimize MAPE. When we select these variables from the initial minimization of SSE as done in OLS, we ensure that the variables selected provide significant effects by selecting them only if their p-value is less than .005, which is 10 times more stringent than the normal accepted cutoff value. We want to be more stringent, because we know that by minimizing for MAPE may slightly change the significance of the variables as well as we anticipate the distribution of the variance structure to not mirror a perfect normal curve. Therefore, in order to resolve those issues and ensure we select the correct variables, we decide to make our selection criteria more stringent. Because we only use OLS to select our variables and minimize MAPE when determining our model, we do not need to satisfy the assumptions of OLS. However, because we select the variables stringently, we know that the model consists of the correct variables for prediction.

We use OLS to choose the x variables in order to reduce the amount of time to determine the model, because we use mixed stepwise regression to limit 1,000 variables to about ten. We use a mixed stepwise regression that enters all variables into the model to begin; therefore, the mixed stepwise regression eliminates a variable based on the criteria and runs a new model, then excludes another variable for a new model and then re-includes the originally excluded variable to see if the originally excluded variable gains significance from removing the second variable. This process continues so that every variable that is excluded gets re-included to check for significance every time a new variable is excluded. Because of computational power limitations, we limit the first 500 variables to about fifty and then take the most predictive 50 and enter them into a second mixed stepwise regression with 550 variables. We then limit this list down to about ten variables. Moreover, we create $(500! - 50!) + (550! - 10!)$ models to determine which variables we select for the final model. Because of the amount of models created from the mixed stepwise regression, we use OLS in order to complete the computations.

We then do a forward stepwise regression to determine the variables as a validation of the variables selected by this process. A forward stepwise regression starts with no variables in the model and adds one variable at a time depending upon its significance. After running the forward stepwise regression, we compare the variables chosen from the forward stepwise regression to that of the mixed stepwise regression. If the variables remain relatively constant, we know that the model is relatively stable.

While we do not need to satisfy the assumptions of OLS, we still need to check for influential data points. In order to determine whether a model contains overly influential data points, we calculate the Cook's distance (a measure of data point influence) for all of the data points. If a data point contains a Cook's distance greater than 0.5, we need to look at why the data point contains this influence and assess whether we can use the model.

In order to validate the final model, we compare the MAPE of the model to the validation set and use the difference of means t-test. We run a two sided t-test with a .1 level of significance; therefore, because we want to fail to reject the test, a .1 level of significance is more stringent than a .05 level of significance. If the model passes the test, then the model does not over-fit the data and can be applied to outside data in the future. If the model does not pass the test, then the model works well within the given database, but cannot be applied to any other data sets and the results have no meaning outside of the given database.

Control Chart

After minimizing for the MAPE and determining our final model, we create a control chart to predict when a problem may occur in an acquisition program. By using the predicted EAC values of the model and comparing them to the actual values of whether or not a problem occurs, we determine the optimal point in which to draw the control chart bound for the most accurate results. We determine the most accurate results by comparing the percent of problems detected and the percent of time a problem exists when a detection occurs, while trying to keep the amount of times the model detects below 25% of the time. We do not want to detect more than 25% of the time, because if the model detects too often, the model becomes a less accurate and unreliable source; moreover, if the model only detects one of four times, users will execute more diligence than a model that has an alarm one third or half the time. We do not minimize for false detections, because we cannot optimize for two outputs; therefore, we maximize the probability of a problem given we detect while keeping the other factors within the bounds we set. Keaton's work that preceded this work used one standard deviation as a bound which equals roughly 33% and we wanted to be slightly more stringent than his work.

We find the control chart bounds by running simulation to determine which bounds output the best results according to the three criteria mentioned in the last paragraph. By setting the bounds to change within a uniformly random distribution and the three criteria as forecast cells, we run simulations using Oracle's Crystal Ball® software (Crystal Ball®, 2011). In other words, we conduct a grid search of all possible control chart bounds through simulation and choose the bounds that provide the most accurate prediction of a five percent absolute change in EAC. We ensure that we draw bounds between every different prediction and test every different bound by doing a million trials, analyzing the results, resetting the random distribution bounds, and re-running a million trials until we hit every combination of bounds. The results of the 3,000,000 simulations output every combination of the variables and by sorting the criteria from the best values to the worst; we determine which bounds provide the most predictive results.

When sorting the results, we sort for the bounds with the maximum number of problems we detect, while keeping the total detections below 25% and the percent of time a problem exists given we detect a problem to be around 40% or more. While we can have a problem exist when we detect over 50% of the time, the total amount of problems we detect drastically decreases. By looking at all the different simulation results, we find the cutoff point that provides the most detections, while maintaining high reliability. Once the cutoff points for the control chart bounds have been determined, we validate the control chart by doing a difference of proportions z-test between the percent of time correct for the analysis data and the validation data. The z-test uses the normal curve to calculate the probabilities, while a t-test uses an approximated normal curve to calculate probabilities. We use a difference of proportions test, because we compare results that are ones or zeros and are not continuous. Therefore, we test the difference of the proportion of ones in one group compared to the proportion of ones in the other group. The difference of proportions z-test uses the following equation for the z-statistic: $Z = \frac{(\widehat{P}_1 - \widehat{P}_2)}{\sqrt{\widehat{P}(1-\widehat{P})(\frac{1}{n_1}+\frac{1}{n_2})}}$ where $\widehat{P}_1$ and $\widehat{P}_2$ define the probability of the two samples, $\widehat{P}$ defines the weighted average of the two probabilities based upon the number of data points in each, and n_1 and n_2 represent the number of data points in each sample. If the data fails to reject the null hypothesis that the data come from the same distribution, then one can assume that the control chart bounds pass a check for reliability.

IV: Results

Text Mining

When running the different combinations of topics through the text mining software, the topics (bins) and the combination bins that comprise of a model change in significance. After testing the different number of topics for the best and most predictive results, we found that 250 topics provide the most predictive results based upon the adjusted R^2. The adjusted R^2 increased from about .21 for 100 topics to about .368 for 250 topics. Increasing the number of topics beyond 250 proves problematic, because JMP® cannot do a stepwise regression, described in the methodology chapter, with all of the variables associated with each topic when increasing beyond 250 topics. Because 250 topics provide the best results, in terms of predictability, given the computational limitations and the knowledge of the diminishing improvement from increasing the number of topics, we use that for creating our model. After running the text mining software with 250 bins, we receive an output of every bin and how often the bin or topic occurs in every document. We use this output to create the percent of time a topic occurs in the document variable as well as all other variations of that variable discussed in the Methodology chapter.

We found that the four, five, and six month model produce nine, nine, and eight significant variables to predict a contractor's EAC, which we reference in the Regression section of this chapter. When using the variables selected from the 250 topic results of text mining, we create three different models to predict the contractor's EAC for programs. The three models, predicting four, five, and six months from the current time period, provide estimates with low error, then we optimize the model to reduce the MAPE using the same variables to create predictions. After creating predictions for the contractor's EAC, we use the predictions to create control charts for each model and then validate both the predictions and the control chart to ensure the applicability to outside data.

Regression

We use a mixed stepwise regression, with an entrance and exit criteria of .0001 and .0001 respectively, and find that the standard deviations of the percent of time a topic (bin) occurs provide the most predictive results. We run a forward stepwise regression with the same criteria for every model and find that one to two variables differ, but the rest of them stay consistent to that of the mixed stepwise regression; therefore, we know the model is stable. Furthermore, the standard deviation variables do not contain high values for only certain programs. For instance, the percent of time a topic occurs in a document includes high percentages only in a couple of programs, which could cause problems when applying these variables to programs not in the analysis set. However, the standard deviations of a topic over the last three periods does not only contain high values for certain programs and contains a much more even spread of values throughout all programs.

The mixed stepwise regression creates a model of eight standard deviation variables for the six month out prediction and nine standard deviation variables for the five and four month out predictions. The largest spread of values for the standard deviation variables is from 0 to 0.34. We attempted to keep the number of variable in the models low so that the models did not become overly complex, which is why the models contain eight to nine variables. When we minimized the models for MAPE opposed to SSE, we reduce the four, five, and six months MAPE's from 3.98%, 4.37%, and 4.80% to 3.32%, 4.13%, and 4.40% respectively. These MAPE scores indicate a measure of accuracy of the predictions to the actual data. The four, five, and six month models' variables and coefficients for the variables can be seen in Table 3, 4, and 5 respectively.

The three tables also include the p-values for the variables as well as the percent impact the variables has on the model. These models create the predicted estimates four, five, and six months from the current time period. The model outputs a ratio of the future predicted EAC to the current EAC most likely; therefore, by multiplying the output of the model by the current EAC most likely, the model produces the predicted future EAC most likely. For example, by taking the four month model's intercept of 1.0079 and multiplying that by a current EAC of $50,000,000, with no input from other variables, our model predicts the EAC four months from the current period to be $50,395,000. Because all of the models apply and contain applicable variables, we can use any of the models for predictions of EAC; however, if we use the six month model, we can see further into the future, but if we use the four month model we will have the least error in our prediction.

Table 3: Four Month Model's Variable Coefficients, p-Values, and % Impacts on the Model

4 Month Model	MAPE Coefficients	OLS P-Values	OLS % impact
Intercept	1.0079	<.0001	0
Stdev(bin 9)	1.38388	<.0001	0.149079311
Stdev(bin 54)	7.27326	<.0001	0.316053208
Stdev(bin 125)	0.05752	<.0001	0.046360071
Stdev(bin 132)	0.06029	<.0001	0.106447832
stdev(bin 171)	1.31883	<.0001	0.043028958
stdev(bin 189)	1.40236	<.0001	0.031219538
stdev(bin 207)	4.14308	<.0001	0.248270577
Stdev(bin 232)	0.84601	<.0001	0.041299535
stdev(bin 238)	0.73161	0.008	0.018240972

Table 4: Five Month Model's Variable Coefficients, p-Values, and % Impacts on the Model

5 Month Model	MAPE Coefficients	OLS P-Values	OLS % impact
Intercept	1.01068	<.0001	0
Stdev(bin20)	0.0261	<.0001	0.036408634
Stdev(bin 73)	0.28377	<.0001	0.361033343
Stdev(bin 84)	0.77975	<.0001	0.067419036
Stdev(bin 125)	0.21233	<.0001	0.070973831
Stdev(bin 171)	0.06477	<.0001	0.055747
Stdev(bin 189)	0.13635	<.0001	0.037253081
Stdev(bin 207)	0.05204	<.0001	0.283899112
Stdev(bin 232)	0.3802	<.0001	0.05941547
Stdev(bin 238)	0.34238	0.0002	0.027850494

Table 5: Six Month Model's Variable Coefficients, p-Values, and % Impacts on the Model

6 Month Model	MAPE Coefficients	OLS P-Values	OLS % impact
Intercept	1.0258	<.0001	0
Stdev(bin20)	2.23325	<.0001	0.042074203
Stdev(bin 73)	9.0752	<.0001	0.374140253
Stdev(bin 84)	2.45713	<.0001	0.081433653
Stdev(bin 125)	0.94413	<.0001	0.080731819
Stdev(bin 168)	0.30253	0.0003	0.031773669
Stdev(bin 171)	1.24956	<.0001	0.05519402
Stdev(bin 204)	2.4461	<.0001	0.037875145
Stdev(bin 207)	4.51444	<.0001	0.296777238

Influential Data Points

The four month model did not contain any overly influential data points, because none of the model's Cook's distances are greater than 0.5. As seen in Figure 2 containing the overlay plot of the Cook's distances, the highest Cook's distance is less than 0.4. Similarly, the five month model also did not contain overly influential data points and contains a highest Cook's distance of about 0.4, which can also be seen in Figure 2. The six-month model did have a slight issue with an overly influential data point. As displayed in Figure 3, one data point contains Cook's distance of about 0.7, which is above the normally accepted threshold of 0.5; however, when excluding that data point and re-running the model with the same variables and calculating the Cook's distances again, the highest Cook's distance becomes around 0.35.

While the p-values change by excluding the data point, all of the data points contains enough significance that the p-values in JMP$^{®}$ are essentially identical. When observing this data point more closely, we notice that the data point's influence stems from predicting about a 75% increase in the EAC compared to the actual 60% increase in EAC; moreover, this point stood out, because most other predictions underestimate the prediction change compared to the actual change in EAC. Because no other data points became overly influential after removing that point, the results indicate that the high Cook's distance stems from leverage more than the error of the data point, which we confirm by calculating both. Moreover, the beta coefficients of the model remain practically constant with and without excluding the data point, which further indicates that the data point does not overly affect the model. Because of these reason, we believe that the model does not have influential data issues and we leave the data point in the analysis.

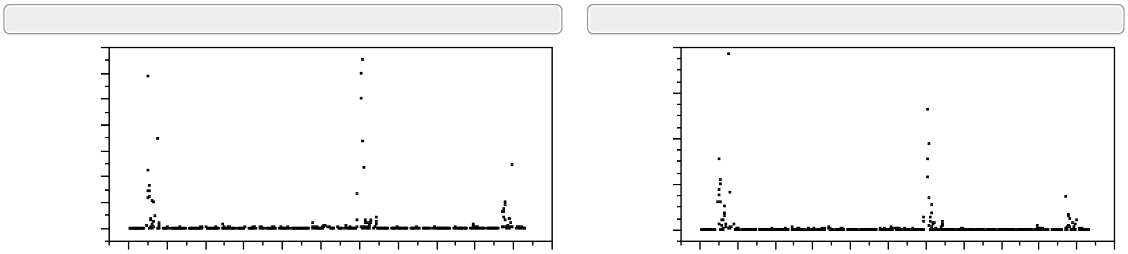


Figure 2: Display of Cook's Distances for 4 and 5 Month Models

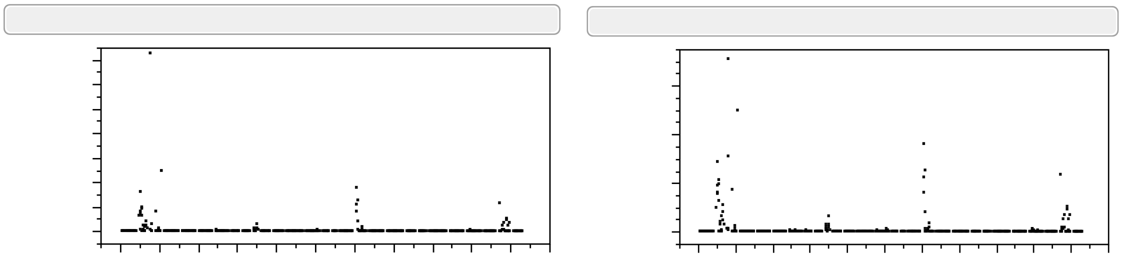


Figure 3: Display of Cook's Distances for 6 Month Model With and Without Excluding Data Point 76

Control Chart

We create control charts for each model and find that the six month model performs best as a control chart. Both the four and five month control charts provide similar to worse results of whether a problem exists when the control chart states there will be one and they only detect around 50% of the problems, while the six month control chart detects 79% of the problems, a drastic increase. The six month model that creates predictions for the control chart provides the decision maker with the most problems being detected, while maintaining reliability. In the control chart displayed in Figure 4, the six month change in EAC predictions have been plotted and the optimized bounds drawn for the control chart. Figure 4 only contains a portion of the data points in order to better visualize how the control chart works; moreover, we have included data points 351-550 to provide what 200 of the data points look like on a control chart.

The circles on the chart display the times the control chart was correct, while the x's display the times in which the control chart was incorrect. Moreover, the circles outside the bounds depict times when the control chart predicts a problem and one does exist, while the x's outside the bounds depict instances in which the control chart predicts a problem and one does not exist. Furthermore, every prediction over 1.035 correctly identifies a 5% change in EAC with no false detections. Therefore, we know that the larger the prediction our model produces, the greater confidence we can have that a problem will occur.

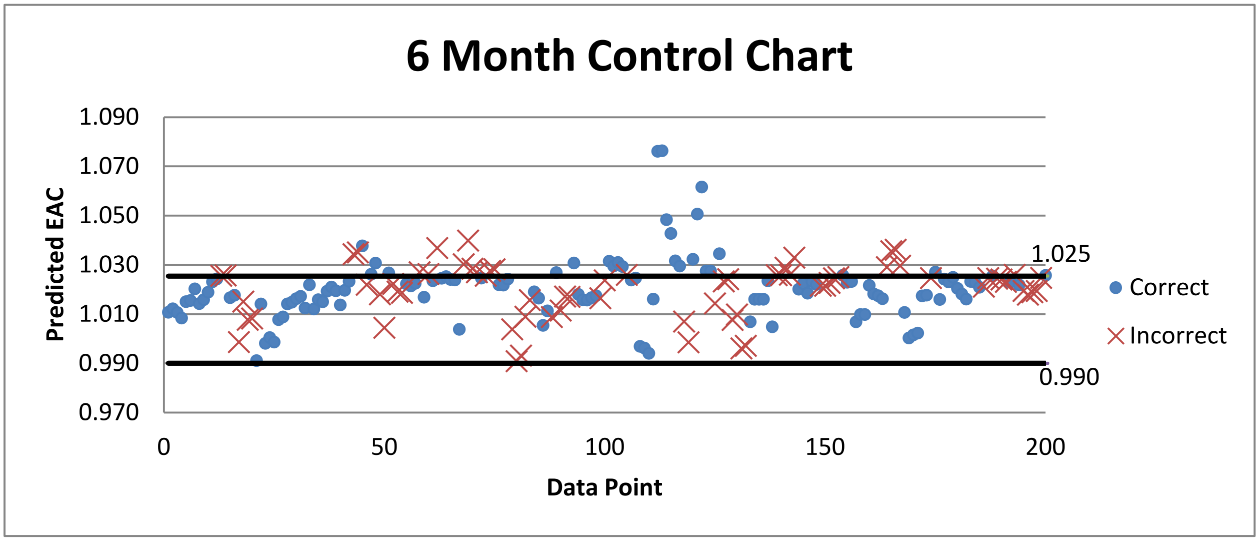


Figure 4: 6 Month Control Chart

Table 6: 6-Month Control Chart Results

	Within 1 Month of Occurrence	Within 2 Month of Occurrence	Within 3 Month of Occurrence	Within 4 Month of Occurrence	Within 5 Month of Occurrence	Within 6 Month of Occurrence
Percent of Total Problems Detected	37.50%	50.00%	66.67%	77.08%	77.08%	79.17%
Probability of Problem Given a Detection	9.14%	16.24%	24.37%	31.47%	36.04%	39.09%
Probability of Problem with No Information	5.81%	11.15%	15.91%	19.74%	22.88%	25.00%
Probability of a Problem Given No Detection	4.95%	9.74%	13.37%	16.34%	19.31%	22.11%

A comprehensive breakdown of the four, five, and six month control chart can be seen in Table 6. Table 6 provides the percent of problems detected, probability of a problem given a detection, probability of a problem with no information, and probability of no problem given no detection for one to six months out. The six month model detects only 24% of the time, which keeps the model reliable, detects 79.2% of the problems, is correct 39.1% of the time when it says there is a problem, and is correct 78% of the time when it says there is not a problem. Furthermore, Table 7 displays the conditional probabilities associated with a detection by our algorithm. For example, given our algorithm detects, there is a 39.8% chance of there being a 5% absolute change in EAC, while the current detection method only provides a 22.7% probability.

Table 7: Text Mining and the Gold Standard Conditional Probabilities

Text Mining Method

	Detect	**No Detect**
Problem	0.398239	0.2018101
No Problem	0.600416	0.7986272

Gold Standard (CPI Detection)

	Detect	**No Detect**
Problem	0.226886	0.2799724
No Problem	0.773131	0.7200276

Table 8: Control Chart Data

	4-Month Predictions	5-Month Predictions	6-Month Predictions
Upper Control Chart Bound	1.01812069	1.0119984	1.02547711
Lower Control Chart Bound	0.99951871	1.00696351	0.99001984
% of Time Detection Occurs (analysis)	0.09175377	0.16466346	0.24533001
% of Time Detection Occurs (validation)	0.25114155	0.19811321	0.25980392
% of Total Problems Detected (analysis)	0.48387755	0.51020408	0.79166667
% of Total Problems Detected (validation)	0.6	0.5	0.5

The results comparing all of the control charts of each model can be seen in Table 8. Table 8 includes the upper and lower bounds of the control charts, the percent of time each model detects a problem, and percent of total problems detected for the validation and analysis set of data. This table provides a comprehensive view comparing the different control charts characteristics. As seen in Table 8, the six month control chart detects far more problems than the four or five month control charts and has a slightly higher upper control chart bound than the other control charts, which means that the six month predictions contain a greater spread than the other two models.

Validating the Models

We validate that the models created by the analysis set of data did not over fit the data by comparing the results on a validation set (20% of the original data). We conduct a difference of means test to determine if the results in the analysis set of data differ from that of the validation set of data. When comparing the APE’s of the analysis and validation set of data, we find that the results suggest no statistical difference between them, because the p-values are all greater than 0.1. Furthermore, the models that have p-values close to 0.1 actually improved upon their MAPE’s, which means that the model could not have over fit the data. The results comparing the four, five and six month prediction analysis set and validation set can be viewed in Table 9.

Table 9: Comparison of Data for Validation of Predictions

	4 Month	5 Month	6 Month
MAPE (analysis set)	3.316127	4.128455	4.395834
MAPE (validation set)	3.255639	3.493956	3.924226
T-test p-value	0.86926	0.120048	0.260169
n (analysis set)	861	832	803
N (validation set)	220	212	204

In order to validate the control charts, we compare the percent of time the control chart is correct with the analysis set and the validation set. Specifically, we compare the combination of the percent of time the control chart indicates a problem and there is a problem as well as the percent of time we don’t say there is a problem and there is not one. The four and six month models control charts show no statistical difference between the analysis set and the validation set; however, the five month does show a statistical difference. Nevertheless, the five month model's statistical difference stems from an improved percent of time the model is correct in the validation set and the model passes validation and can be applied to outside data. The Z-test p-values for the difference of proportions test can be viewed in Table 10.

Table 10: Comparison of Data for Validation of Control Chart

	4 Month	5 Month	6 Month
% of Time Correct (analysis)	0.741587	0.688702	0.683686
% of Time Correct (validation)	0.731132	0.787736	0.622642
Z-test p-value	0.7604	0.0046	0.119
n (analysis set)	861	832	803
N (validation set)	220	212	204

V. Conclusions

Discussion of Results

The results of this research suggest important findings. We can accurately quantify qualitative textual data, determine a relationship between text mining results and EVM data, use the text mining results to predict EAC, and predict EAC more accurately than current abilities. By analyzing the usage of words in a document, we can predict the contractor's EAC up to six months from a current time period with about 4% error on average. Furthermore, we predict whether a program will incur a 5% change in its EAC from one month to the next, six months before the occurrence. These results provide evidence that text portions of a CPR can predict the EAC through text mining. This research expands the uses of text mining by using the techniques in a field in which it has never been applied. Moreover, this research validates the importance of the government paying for the Format 5's to be filled out by the defense contracting companies.

The six month model outperforms the four and five month models. Different reasons can cause the six month predictions improve over the four and five month models. The different variables chosen for the six month model could include more predictability than the variables for the other variables. More likely though, contractors write down information in the Format 5 that does not show up in the EAC numbers until six months later.

This research not only determines that text mining could be used in the future, but also applies the research by creating two tools in which to use the results. The model that creates predicted contractor's EAC contains a MAPE of approximately 3%. Being able to predict the EAC four months from the current time period with approximately 3% error allows the user to better understand the progress and status of their programs. The control chart offers even more applicable results. The control chart can predict 80% of problems in programs and when the control chart predicts a problem, a problem exists over 39% of the time. The percent of time that a problem exists within six months of any time period is about 25%; therefore, the created control chart increases the probability by 56%. The control chart only detects less than 25% of the time to improve reliability; moreover, the control chart improves the decision maker's ability to know if a problem will occur. Also, the methods we propose in this research provide an increase in knowledge for the decision maker over the currently used methods. When comparing both the prediction model and the control chart with the results from the validation set, both tools pass. Therefore, the tools created provide reliable results that can be applied to outside data in the future.

Implications of Research

This research substantiates that text and written forms have the ability to predict costs and dollar amounts of programs. Not only does the research substantiate the ability, it provides a decision maker two tools as mentioned earlier. The text mining methods add to the abilities of EVM techniques to provide decision makers with more information. By using the tools to better understand the status of programs, a greater concentration of efforts can take place and corrective actions can occur up to six months earlier than otherwise. Furthermore, these tools allow for trend and problem detection in far less time and effort than reading through hundreds of pages every month. With the reduction in manning that the Air Force and the DoD as a whole have been dealing with, a greater understanding of where the manning and effort is most needed can help decision makers more optimally allocate resources. Moreover, these tools provide the decision makers with an objective metric to grade the status of a program that might be able to contrast the opinions of program managers. By implementing this research into a program management toolset the government could possibly save millions of dollars as well as deliver the warfighter the necessary high performance tools in a timely fashion.

Follow on Research

This research contains areas for further exploration. The data set we use for this research only includes ACAT 1D programs; moreover, I believe the tools created from the data set can only be applied to ACAT 1D data. By collecting data from non-ACAT 1D programs, the tools and models we created can be validated for applicability. Further research could be done by updating the database used for this research and testing how the model continues to perform. Moreover, by updating the text mining software, new programs and future data can be applied to the same model and further validate the results. Also, new models could be created and tested for improved results. Additionally, the research can be applied to non-DoD data.

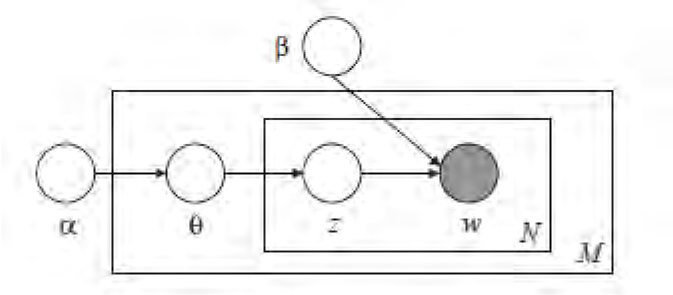
By applying other predictive variables of EAC a control chart with variable bounds can be created. By using two models to create the varying bounds control chart, the percent of time a problem exists when detecting as well as total number of problems we can detect can increase. This application of two models will allow for greater control of the results and be able to mold to decision makers desires more easily. Moreover, a model could be created to predict beyond six months and see if text mining techniques can still have accurate results beyond six months.

This research uses LDA for the text mining techniques. By using other text mining techniques the results could improve or get worse, but the methodology has never been attempted before. Also, the current code used for this research could be improved upon by creating a better list of stop words and words that should be excluded from the text mining sorting process. Also, by separating the Format 5 portions into separate Work Breakdown Structure (WBS) elements and running a text mining analysis, text mining might be able to be used to predict where exactly a problem will occur within a program.

All of the proposed follow on research expands the boundary of what has been done in the cost analysis community. The results of this research and future research findings can save the Air Force and the DoD millions of dollars through their application. Furthermore, they will allow for the warfighter to receive their needs in the field quicker with less schedule delays. Moreover, these results improve the acquisition process of the DoD by producing goods with less cost and schedule overruns.

Appendix A: Technical Description of LDA

"LDA is a generative probabilistic model of a corpus. The basic idea is that documents are represented as random mixtures over latent topics, where each topic is characterized by a distribution over words (Blei, 2003)." The LDA process does this representation through a series of mathematical expressions. First, LDA chooses N based upon the Poisson distribution and a topic mixture based upon the Dirichlet distribution. Furthermore, for each of the words the process selects a topic with a Dirichlet distribution and a word with a multinomial probability conditioned on the topic chosen (Blei, 2003). Blei uses the following image to describe the process of LDA.



This image structure resembles structures seen in parametric empirical Bayes models (Blei, 2003). The M in the image represents the number of documents, while the N represents the number of words throughout the documents. The *w* represents a word, while *z* represents a topic. β represents a matrix of word probabilities that are parameterized by the number of topics and unique words. θ represents the distribution of a topic mixture in the documents and is dimensioned by the number of topics (Blei, 2003). Finally, α represents a vector of the topics. This process learns where words should adequately be placed by finding words that closely relate to a word and determining whether consistent trends of relationships exist.

Appendix B: List of Programs and the Number of Monthly Data Associated with Each Acquisition Program

B2-EHF	14
AMF JTRS SDD (BBX)	20
MM III GRP FRP '07	20
Non Line of Sight - Launch System (FCS Navy)	20
C130J BUIC Del Order 0003	22
LCS - CLIN 0008 AUSTAL	24
E-2D Advanced Hawkeye (AHE)	27
EFV SDD-2	27
B-2 RMP	28
FORCE XXI BATTLE COMMAND BRIGADE AND BELOW (FBCB2)	28
NPOESS	28
NMT EDM	30
C-130 Block 6.5.1 HCMC	31
E871209B (MH-60)	31
CH-53	32
V-22	33
WINT_INC2-M	33
ISPAN	34
MPS – FPM	37
UH-60M	37
WGS BLOCK II	37
MP-RTIP Phase 2	41
Blue Grass Chemical Agent Destruction Pilot Plant	42
DDG 1000	42
F-35 JSF System Development & Demonstration	42
Chem. Demil Stockp (Chem Demil CMA)	43
GPS MUE CLIN 002 (Navstar)	43
C130 Avionics Moderinzation Program	44
SBIRS	44
AEHF	45
C-5 Reliability Enhancement & Reengining Program SDD	45
MPEC JMPS-E (mps-exp ops)	45
SM6	45
MPS SEICR1	48
MOBILE USER OBJECTIVE SYSTEM (MUOS)	50
JLENS	52
P-8	52

Appendix C: List of Acronyms

ACWP:	Actual Cost of Work Preformed
AFIT:	Air Force Institute of Technology
BCWP:	Budgeted Cost of Work Preformed
BCWS:	Budgeted Cost of Work Scheduled
CPI:	Cost Performance Index
CPR:	Contract Performance Reports
CTM:	Correlated Topic Model
DoD:	Department of Defense
DoDCAS:	Department of Defense Cost Analysis Symposium
EAC:	Estimate at Complete
EVM:	Earned Value Management
HTML:	Hyper Text Markup Language
LDA:	Latent Dirichlet Allocation
LSA:	Latent Semantic Analysis
MAPE:	Mean Absolute Percent Error
OLS:	Ordinary Least Squares
PDF:	Portable Document File
PLSA:	Latent Semantic Analysis
RAKE:	rapid automatic keyword extraction
SBIRS:	Space Based Infrared System
SPI:	Schedule Performance Index
SSE:	Sum of Squared Error
WBS:	Work Breakdown Structure
WMD:	Weapons of Mass Destruction
XML:	Extensible Markup Language

Bibliography

Ananiadou, S., Sullivan, D., & Black, W. (2011). Named Entity Recognition for Bacterial Type IV Secretion Systems. *PLoS ONE*, *6*(3). Retrieved September 15, 2011, from http://www.plosone.org/article/info%3Adoi%2F10.1371%2Fjournal.pone.0014780

Berry, M. W., & Kogan, J. (2010). *Text Mining Applications and Theory*. Hoboken NJ: John Wiley & Sons.

Blayney, J. (2012, Feb 10). CEO RenewData. (T. Miller, Interviewer)

Blei, D., & Lafferty, J. (2009). Topic Models. *Text mining: classification, clustering, and applications* (pp. 71-83). Boca Raton FL: CRC Press.

Bush, M. (2009, June 27). Text Mining Provides Marketers With the 'Why' Behind Demand. *Ad Age Digital*. Retrieved October 22, 2011, from adage.com/article/digital/marketing-text-mining-demand/138110/

Carroll, John, Evans, Roger and Klein, Ewan (2005) *Supporting text mining for e-Science: the challenges for Grid-enabled natural language processing.* In: Workshop on Text Mining, e-Research And Grid-enabled Language Technology at the Fourth UK e-Science Programme All Hands Meeting (AHM2005), 19-22 Sep 2005, Nottingham UK.

Christensen, D. (1993). An analysis of cost overruns on defense acquisition contracts. *Project Management Journal*, *3*, 43-48.

Corley, C., Cook, D., Mikler, A., & Singh, K. (2010). Text and Structural Data Mining of Influenza Mentions in Web and Social Media. *Environmental Research and Public Health*, *7*. Retrieved January 8, 2012, from www.mdpi.com/journal/ijerph

Crystal Ball®, Fusion Edition. 2011. Redwood Shores CA: Oracle, Inc.

DTRA. (n.d.). Home. *Defense Threat Reduction Agency*. Retrieved January 20, 2012, from http://www.dtra.mil/Home.aspx

Feldman, R., & Sanger, J. (2007). *The text mining handbook: advanced approaches in analyzing unstructured data*. Cambridge: Cambridge University Press.

Giles, J. (2011, May 21). 2020 vision: The crystal ball internet. *NewScientist*. Retrieved November 25, 2011, from www.newscientist.com/article/mg21028121.900-2020-vision-the-crystal-ball-internet.html

JMP®, 9th Edition. 2011. Cary NC: SAS Institute, Inc.

Keaton, C., White, E., & Unger, E. (2011). Using earned value data to detect potential problems in acquisition contracts. *Journal of Cost Analysis and Parametrics*, *4*(2), 148-159.

Keaton, Charles. *Using Earned Value Data to Detect Potential Problems in Acquisition Contracts*, MS thesis, AFIT/GFA/ENV/11-M02, School of School of Engineering and Management, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, March 2011a.

Keaton, C. (Director) (2011b, February 15). Problem Detection Using Text Mining Techniques. *DoDCAS*. Lecture conducted from Office of the Secretary of Defense, Williamsburg.

Khan, H. (2011, October 13). *GOP Lawmakers, Secretary Leon Panetta Ring Alarm Bells on Defense Budget Cuts*. Retrieved from http://abcnews.go.com/blogs/politics/2011/10/gop-lawmakers-secretary-leon-panetta-ring-alarm-bells-on-defense-budget-cuts/

Kolluru, B., Hawizy, L., & Murray-Rust, P. (2011). Using Workflows to Explore and Optimise Named Entity Recognition for Chemistry. *PLoS ONE*, *6*(5). Retrieved September 17, 2011, from http://www.plosone.org/article/info%3Adoi%2F10.1371%2Fjournal.pone.0020181

Lee, S., Song, J., & Kim, Y. (2010). An Empirical Comparison of Four Text Mining Methods. *Journal of Computer Information Systems*, *3*, 1-10.

Metz, C. (n.d.). Software: Text Mining - The Future of Technology | PCMag.com. *Technology Product Reviews, News, Prices & Downloads | PCMag.com | PC Magazine*. Retrieved February 19, 2012, from http://www.pcmag.com/article2/0,2817,1130911,00.asp

Millar, J., Peterson, G., & Mendenhall, M. (2009). Document Clustering and Visualization with Latent Dirichlet Allocation and Self-Organizing Maps. *FLAIRS Conference*, *21*, 69-74.

Newbold, P., Carlson, W. L., & Thorne, B. (2010). *Statistics for Business and Economics* (7th ed.). Boston: Prentice Hall.

Ng, K. W., & Tian, G. (2011). *Dirichlet and Related Distributions: Theory, Methods and Applications*. Hoboken N.J.: Wiley.

QDA MINER®, 4th Edition. 2005. Montreal QC: Provalis Research.

Riza, T. (Director) (2011, January 6). Discovering Potential Drugs by Extracting Biological Activities of Natural Products. *Pacific Symposium on Biocomputing*. Lecture conducted from Stanford, Big Island.

Sallach, D. (Director) (2009, November 5). Data Theory, Discourse Mining and Thresholds. *AAAI Fall Symposium*. Lecture conducted from AAAI, Arlington.

Shaw, M., Subramaniam, C., Tan, G., & Welge, M. (2001). Knowledge Management and Data Mining for Marketing. *Decision Support Systems*, *31*(1), 127-137.

Sirmakessis, S. (2004). *Text mining and its applications: results of the NEMIS Launch Conference*. Berlin: Springer-Verlag.

Stewart, D. (2003). DoD Personnel: Documentation of the Army's Civilian Workforce-Planning Model Needed to Enhance Credibility. *GAO*, *03-1046*, 1-13.

The Secretary of Defense. (2009). *U.S. Department of Defense Office of the Assistant Secretary of Defense*. Retrieved February 10, 2012, from http://www.defense.gov/releases/release.aspx?releaseid=12652

The White House. (2011). *The White House*. Retrieved October 7, 2011, from http://www.whitehouse.gov

Thuraisingham, B. (2002). Data Mining for Counter Terrorism. *AAAI Press*, *1*, 191-218.

Wallender, T. (1994). Guide to Analysis of Contractor Cost Data. *Air Force Materiel Command*, *65-501*, 1-70.

Wilson, A., & Chew, P. (2006). Term Weighting Schemes for Latent Dirichlet Allocation. *COLING-ACL 2006: 21st International Conference on Computational Linguistics and 44th annual meeting of the Association for Computational Linguistics.* (pp. 465-473). Stroudsburg: Association for Computational Linguistics.

Younossi, O. (2007). *Is weapon system cost growth increasing?: a quantitative assessment of completed and ongoing programs*. Santa Monica: RAND Project Air Force.

Yu, S., Tranchevent, L., Moor, B. D., & Moreau, Y. (2010). Gene Prioritization and Clustering by Multi-View Text Mining. *BMC Bioinformatics*, *11*(28), 1-22.

Zelikovitz, S., & Kogan, M. (2006). Using Web Searches on Important Words to Create Background Sets for LSI Classification. *FLAIRS Conference* , *1*, 298-603.

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 074-0188*

1. REPORT DATE *(DD-MM-YYYY)* 22-03-2012	**2. REPORT TYPE** Master's Thesis	**3. DATES COVERED** *(From – To)* Aug 2010-Mar 2012

4. TITLE AND SUBTITLE Acquisition Program Problem Detection Using Text Mining Methods	**5a. CONTRACT NUMBER** **5b. GRANT NUMBER** **5c. PROGRAM ELEMENT NUMBER**
6. AUTHOR(S) Miller, Trevor P, 2Lt., USAF	**5d. PROJECT NUMBER** **5e. TASK NUMBER** **5f. WORK UNIT NUMBER**
7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S) Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/EN) 2950 Hobson Way WPAFB OH 45433-7765	**8. PERFORMING ORGANIZATION REPORT NUMBER** AFIT/GCA/ENC/12-02
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Mr. Steven Miller., Civ, OSD/CAPE Office of the Secretary of Defense Cost Assessment and Program Evaluation Room BE-829 1800 Defense Pentagon Washington, DC 20301 (703) 697-5056	**10. SPONSOR/MONITOR'S ACRONYM(S)** OSD/CAPE **11. SPONSOR/MONITOR'S REPORT NUMBER(S)**

12. DISTRIBUTION/AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED
.

13. SUPPLEMENTARY NOTES

14. ABSTRACT
This research provides program analysts and Department of Defense leadership with an approach to identify problems in real-time for acquisition contracts. Specifically, we test the abilities of efficient algorithms using text mining techniques to detect unusual changes in acquisition programs' cost estimates at the completion of the programs. Currently, the government purchases monthly written reports, an informational tool on status of an acquisition program, but has not been integrated into problem prediction analysis. We center our research on the following two questions: First, can we quantify the qualitative written reports? Second, can we use these quantifications of the texts to predict cost growths in acquisition programs? Through using text mining techniques, we validate the worth of the written reports by creating algorithms that identify 80% percent of problems in acquisition programs, while increasing the probability of a problem existing given our algorithm detects by 56% from the current methods. These positive results for this analysis provide program offices with a method to detect potential problems in acquisition contracts; furthermore, this research helps the government more efficiently manage their resources as well as reduce cost and schedule overruns.

15. SUBJECT TERMS
Text Mining, Latent Dirichlet Allocation, Estimates at Complete

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** Dr. Edward White (AFIT/ENC)
a. REPORT U	**b. ABSTRACT** U	**c. THIS PAGE** U	UU	69	**19b. TELEPHONE NUMBER** *(Include area code)* 937-255-3636 ext 4540, Edward.White@afit.edu