technical note technical note

Textual Data-Mining Literature Survey Report

Matthew Davino
SARTP Intern

April 2019

DOT/FAA/TC-TN19/18

U.S. Department of Transportation
Federal Aviation Administration

NOTICE

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
DOT/FAA/TC-TN19/18		

4. Title and Subtitle	5. Report Date
TEXTUAL DATA MINING LITERATURE SURVEY REPORT	April 2019
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
Matthew Davino	

9. Performing Organization Name and Address	10. Work Unit No. (TRAIS)
Aviation Research Division Software and Systems Branch Software and Electronics Section – ANG-E271 FAA William J. Hughes Technical Center Atlantic City International Airport, NJ 08405	
	11. Contract or Grant No.

12. Sponsoring Agency Name and Address	13. Type of Report and Period Covered
Aviation Research Division Software and Systems Branch Software and Electronics Section – ANG-E271 FAA William J. Hughes Technical Center Atlantic City International Airport, NJ 08405	Tech Note
	14. Sponsoring Agency Code ANG-E271

15. Supplementary Notes

The FAA William J. Hughes Technical Center Aviation Research Division COR was Alanna Randazzo.

16. Abstract

Because of the high volume of reports filed within the U.S. Department of Transportation and its various branches, the U.S. Department of Transportation has accumulated high volumes of largely unstructured data. This information could be invaluable because it may uncover unnoticed patterns with respect to accidents or malfunctions. It could also help to efficiently use resources, uncovering what a large organization, such as the U.S. Department of Transportation, might be spending too many or too few resources on. This report examines several documents regarding this issue, exploring potential solutions to those issues caused in analyzing unstructured data that do not necessarily exist in structured data.

17. Key Words	18. Distribution Statement
Unstructured Data, Structured Query Language (SQL), Aviation Safety Information Analysis and Sharing, ASIAS, AIRES, NoSQL, Unstructured Query Language, TMI, Textual Data Mining Infrastructure, Machine-learning, Kalman filter algorithm, Data-mining, Apache Mahout, Hierarchical Distributed Dynamic Indexing System, HDDIS, Information Gain	This document is available to the U.S. public through the National Technical Information Service (NTIS), Springfield, Virginia 22161. This document is also available from the Federal Aviation Administration William J. Hughes Technical Center at actlibrary.tc.faa.gov.

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	12	

TABLE OF CONTENTS

LIST OF ACRONYMS

AIRES	ASIAS Information Retrieval and Extraction System
ASIAS	Aviation Safety Information Analysis and Sharing
HDDIS	Hierarchical Distributed Dynamic Indexing System
NoSQL	Unstructured Query Language
NTSB	National Transportation Safety Board
SQL	Structured Query Language
TMI	Textual Data Mining Infrastructure

EXECUTIVE SUMMARY

Because of the high volume of reports filed within the U.S. Department of Transportation and its various branches, the U.S. Department of Transportation has accumulated high volumes of largely unstructured data. This information could be invaluable because it may uncover unnoticed patterns regarding accidents or malfunctions. It could also efficiently use resources, uncovering what a large organization, such as the U.S. Department of Transportation, might be spending too many or too few resources on. This report examines several documents regarding this issue, exploring potential solutions to those issues caused in analyzing unstructured data that do not necessarily exist in structured data.

BACKGROUND

Within the FAA and the greater US Department of Transportation as a whole, vast amounts of both structured and unstructured data have been accumulated. The large amounts of information have caused a shift in how data are stored, from Structured Query Language (SQL) databases to Unstructured Query Language (NoSQL) databases.

SQL databases, as their name implies, have a structure. When adding data to the SQL database, the data must be able to conform to the database's existing structure. This works well for collecting such data as how many "likes" a post has on a social media site and other information, such as the date it was posted, the name of the user, and other bits of data that can be easily expressed in structures similar to that of a table.

NoSQL databases do not require the same kind of structure to store data. The lack of a necessary structure allows for easier storage of documents and forms, and the data within them. However, without standard structures, NoSQL databases require different techniques to search through the database and retrieve relevant data. To circumvent the issue caused by NoSQL databases, a wide variety of statistical methods and data-encoding and machine-learning techniques are being developed and used to search through the contents of forms and documents.

To get the best use of the libraries of documents at the US Department of Transportation's disposal, the implementation of systems that can mine text and phrases from documents within their databases would greatly increase the abilities of the Department. By implementing a system that can accurately search NoSQL databases for relevant documents and other works, the US Department of Transportation and its smaller administrations will be able to perform their tasks more easily and efficiently, and would be less likely to use resources to repeat a project that was lost within the database.

SUMMARY OF TASK

This report was made to overview literature within databases accessible from within the FAA pertaining to methods of mining textual data from documents within NoSQL databases and to find which ones are applicable. The report also provides an overview of any existing methods of mining textual data that are readily at the disposal of the FAA.

SUMMARY OF LITERATURE SURVEY

The following section will provide summaries of the reports, the potential use of what is detailed in the report, and any potential issues regarding what is detailed.

DATA MINING FROM DOCUMENT-APPEND NOSQL

The primary purpose of this paper is to compare existing textual mining tools, such as Apache Mahout™, with an adapted version of the Kalman filter algorithm for data-mining tasks such as topic extraction, term clustering, and others.

The Kalman filtering algorithm is a dynamic algorithm for which, at specific increments of time, the algorithm estimates joint probability distribution, keeping track of the estimates, the uncertainty, and the variance of previous estimates to update the current estimate. Until now, the Kalman filter algorithm has found wide use in navigation-control systems and limited use within data mining.

For optimal results, the algorithm heavily benefits from the search history to filter search results. The base relationships for this algorithm are:

$K \subset Dic$

$T \subset Db$

Where *K* means specified keywords, *T* means constructed terms, *Dic* represents the dictionary and thesaurus, and *Db* is a subset of the dataset in the NoSQL database. For instances in which the term *K* does not exist within *Dic* because of such situations as the term not being found in the dictionary or lacking synonyms, then $K=T$. Whether *T* exists within a large data source is represented probabilistically between 1 and 0, with 1 meaning that *T* does exist in the dataset and 0 meaning that *T* does not. To handle storage issues within the log of the search history, less-searched keywords get deleted from the log.

To evaluate this algorithm's effectiveness, as used in NoSQL_miner, 2 terabytes of hemophilia-related data and 10 terabytes of psychiatry-related data were used, filling the dictionary with ~6000 medical jargon terms.

The author tested NoSQL_miner against Apache Mahout and R. The tests focused on four major stages of data mining: text extraction, term organization, term classification, and term clustering. The test was run twice, once using the 2 terabytes of hemophilia data and once using the 10 terabytes of psychiatry data.

By defining the variable *Truthfulness* to be *True Positive + True Negative*, and by also defining *Falsehood* to be *False Positive + False Negative*, the author was able to compare the results between the three data miners. Although NoSQL_miner underperformed in truthfulness compared to the other two, with Apache Mahout coming out on top and R being a close second, NoSQL_miner dominated in tests for accuracy, with Apache Mahout only being more accurate than R by a relatively small margin. When it came to the falsehood tests, NoSQL_miner scored the lowest with R scoring the highest in falsehood and Apache_Mahout scoring a close second. In terms of speed, NoSQL_miner was much faster than Apache Mahout and R because of faster processing of data on the part of NoSQL_miner.

Although the results of NoSQL_miner are superior, it is difficult to conclude to what extent NoSQL_miner outperforms Apache Mahout or R for several reasons. One of the two largest

reasons is that this paper is not totally unbiased because the authors have created the NoSQL_miner themselves, so this test is not totally independent. The other reason is that NoSQL_miner has a better system for reading data. These tests do not assess just the algorithms of Apache Mahout or R, but their whole system, leaving room for improvement with Apache Mahout and R that may cause them to outperform NoSQL_miner.

A SOFTWARE INFRASTRUCTURE FOR RESEARCH IN TEXTUAL DATA MINING

The Textual Data Mining Lab in Lehigh University's Department of Computer Science and Engineering had developed a Textual Data Mining Infrastructure (TMI) in C++ to address challenges researchers face in the field of textual data mining, believing that the existing tools for textual data mining are either too specific in their applications or are in their infancy, and are not able to be used as general-purpose textual-mining tools. The TMI can find use in both traditional text-processing applications and those that combine text processing and machine learning.

Traditional text-processing applications that the TMI can employ include search and retrieval of documents. The TMI extends and enhances the Hierarchical Distributed Dynamic Indexing System (HDDIS). This system also was created at Lehigh University. The HDDIS is a system that dynamically creates a hierarchical system from distributed document collections by first creating nodes from which a knowledge base is created, and subtopic regions within the base are identified. The TMI also supports a wide range of evaluation methodologies and metrics, and its search and retrieval applications use gold-standard evaluation collections (however, this paper was written in 2004, so what was once considered "gold standard" may no longer be).

The main focus of the TMI seems to be its use with machine learning. The TMI can combine text processing with machine learning, allowing it to be used in such applications as trend detection. The TMI can be used with both supervised and unsupervised machine learning for any machine-learning application. The TMI is compatible with the University of Waikato's WEKA and MLC++'s machine learning libraries and also has several advantages over Data to Knowledge because the TMI is focused solely on text mining.

AN EVALUATION OF AIRES AND STATISTICA TEXT-MINING TOOLS AS APPLIED TO GENERAL AVIATION ACCIDENTS

In this report, the authors try to use text-mining tools to find patterns leading to general aviation accidents from accident reports within the National Transportation Safety Board (NTSB) database. To do this, the authors tested two text-mining applications, STATISTICA and ASIAS Information Retrieval and Extraction System (AIRES).

AIRES was developed by the MITRE Corporation in conjunction with the FAA's Aviation Safety Information Analysis and Sharing (ASIAS) initiative in an attempt to solve data-sufficiency problems and address problems that deal with insufficient information within the document. AIRES attempts to do this by analyzing other, more complete documents within the ASIAS system so that it may have a better grasp on documents that are not nearly as complete.

STATISTICA is a privately developed, commonly used text-mining software. This software was more work-intensive to operate than AIRES. Although AIRES is an automated program, STATISTICA requires the user to go through a process to reduce the problem into something more manageable by using frequent words. STATISTICA then generates concepts for a document, each concept representing the amount of variation in the text. The concepts consume computational resources, so users of STATISTICA want to minimize concepts while also maximizing the variance.

To demonstrate the capabilities of these two tools, the authors used AIRES and STATISTICA to search the NTSB and gather data related to fatal accidents. They hoped that the tools would be able to identify significant patterns in the documents relating to the fatal accidents. To quantify the success of AIRES, the authors developed several criteria. The first criterion is Information Gain, which represents to what extent each pattern predicts whether the accident will be fatal. The second is Precision, which represents the ratio of useful/relevant received documents and accidents; in this case, this is the ratio of the number of accidents retrieved versus the number of retrieved accidents that are fatal. The next is Recall, which is the number of relevant documents retrieved through use of the algorithm. Finally, they calculate the F-Measure to be the mean of Precision and Recall.

The AIRES tool results demonstrated that accidents are caused by a large number of differing patterns rather than a small number of similar patterns. The top patterns that showed the greatest F-Measure included: into, low, weather, instrument, night, and maneuvering. Although this information may have some initial use, the actual utility of these data in finding patterns is limited and requires further context to better describe the problems caused by these larger patterns.

When using STATISTICA to find patterns within the NTSB regarding fatal accidents, the authors graphed words on a plot graph, with their placement on the graph being proportional to the words' variations. Using this method, they were able to find word clusters including words that AIRES did not pick up.

Because of the variation between the two tools, it is hard to directly compare them. In this case, these tools failed to provide significant useful information and insight into the topic of patterns of fatal accidents on a national level. However, the results are far clearer when using them at more regional levels, providing a different set of word clusters and information, giving a greater insight on causes of fatal crashes in specific regions compared to the insight gathered on a national level.

CONCLUSION

When considering the information gained from *Data Mining From Document-Append NoSQL, An Evaluation of AIRES and STATISTICA Text Mining Tools as Applied to General Aviation Accidents* and other less significant works, we are given several options on how to best gather textual data from the NTSB database. Whereas AIRES and STATISTICA had gathered minimal

information from the data, their use case may have been too broad for their methods to be used effectively. Not much information was found on either system regarding the NTSB database, which might imply that they are no longer used and that a similar amount of effort would need to be made to implement these tools compared to implementing any other system.

It is best to use an Apache system, such as Apache Spark or Apache Mahout, R, or JMP when considering the best tool for this case. Apache systems and R have existed for many years and have a considerable number of libraries regarding data analytics in such tools as Hortonworks, making the job both easier and more efficient. While in JMP, text analytics are also simple and refined. Whichever tool is used should best apply to the use case; each of these systems are equally capable of performing text mining and analytics.

One possible use case for these tools is to mimic the fatal accidents tests from the STATISTICA and AIRES test. Although the information collected may have not been useful because of the words gathered (such as "night," "weather," or "tools"), providing no useful insight into issues regarding fatal accidents, some words that were collected, like "into," were useless and did little more than take up space in the top patterns. This was space that could have been used for something useful. By testing JMP, Apache tools, or R the same way and comparing the data gathered with what was gathered by AIRES or STATISTICA, we can definitively say which text-mining systems are best-suited for the NTSB database.

BIBLIOGRAPHY

Bazargan, M., Johnson, M., & Vijayanarayanan, A. (2013). *An Evaluation of AIRES and STATISTICA Text Mining Tools as Applied to General Aviation Accidents.* Atlantic City International Airport, NJ: Federal Aviation Administration. Retrieved from http://www.tc.faa.gov/its/worldpac/techrpt/tctn13-7.pdf

Delen, D., Fast, A., Mill, T., Elder, J., Miner, G., & Nisbet, B. (2012). *ractical Text Mining and Statistical Analysis for Non-Structured Text Data Applications.* Waltham, MA.: Elsevier.

Holzman, L. E., Fisher, T. A., Galitsky, L. M., Kontostathis, A., & Pottenger, W. M. (2003). A software infrastructure for research in textual data mining. *15th IEEE International Conference on Tools with Artificial Intelligence*, (pp. 112-121).

Lomotey, R. K., & Deters, R. (2014). Data Mining from NoSQL Document-Append Style Storages. *2014 IEEE International Conference on Web Services*, (pp. 385-392). Anchorage, AK. doi:10.1109/ICWS.2014.62

Pottenger, W. M., Kim, Y., & Meling, D. D. (2001). HDDI™: Hierarchical Distributed Dynamic Indexing. In R. L. Grossman, C. Kamath, P. Kegelmeyer, V. Kumar, & R. R. Namburu (Eds.), *Data Mining for Scientific and Engineering Applications. Massive Computing* (Vol. 2). Boston, MA: Springer.

Rushing, W. M., & Wisnowski, J. (2015). Harness the Power of Text Mining; Analyse FDA Recalls and Inspection Observations.

Tudorica, B. G., & Bucur, C. (2011). A comparison between several NoSQL databases with comments and notes. 2011 RoEduNet International Conference 10th Edition: Networking in Education and Research,. Iasi.