Clustering and Classification of Maintenance Logs using Text Data Mining

Brett Edwards **Michael Zatorsky** **Richi Nayak**
CRC for Integrated Engineering Asset Management
Faculty of Information Technology
Queensland University of Technology
PO Box 2434, Brisbane 4001, Queensland
bj.edwards@aanet.com.au miczat@gmail.com r.nayak@qut.edu.au

Abstract

Spreadsheets applications allow data to be stored with low development overheads, but also with low data quality. Reporting on data from such sources is difficult using traditional techniques. This case study uses text data mining techniques to analyse 12 years of data from dam pump station maintenance logs stored as free text in a spreadsheet application. The goal was to classify the data as scheduled maintenance or unscheduled repair jobs.

Data preparation steps required to transform the data into a format appropriate for text data mining are discussed. The data is then mined by calculating term weights to which clustering techniques are applied. Clustering identified some groups that contained relatively homogeneous types of jobs. Training a classification model to learn the cluster groups allowed those jobs to be identified in unseen data. Yet clustering did not provide a clear overall distinction between scheduled and unscheduled jobs.

With some manual analysis to code a target variable for a subset of the data, classification models were trained to predict the target variable based on text features. This was achieved with a moderate level of accuracy.

Keywords: Text Data Mining.

1 Introduction

Relational databases allow data to be stored in a clean and consistent format that allows reports to be developed using well understood software development techniques. Often though, the skills to design and develop structured databases are not held by or cheaply available to individuals with the need to store and report on data. Where the value of the data is great enough, the individuals will "find a way" even if the techniques used to store the data would not be considered best practice by information technology specialists.

A common result of this situation is the creation of basic spreadsheet applications requiring no software development experience. Unfortunately such applications rarely enforce data quality standards. Users add information into unstructured free text fields with little consistency between values. Unlike standard database applications, reporting on information in unstructured formats is not a trivial software development task.

This paper presents a case study of reporting on unstructured free text data using text data mining techniques. The aim is to determine if textual features of the data can be used to classify records into structured attributes. If the accuracy of the classification is high enough, then business decisions can be based on data rather than best guesses.

The client in this study maintains and repairs pump stations for dams and weirs; including the pump motors, electrical systems, fire extinguishing systems, air conditioning systems and external buildings and grounds. They recorded fault logs dating from 1994 to 2006 in a Microsoft Excel spreadsheet where all relevant information was kept in free text fields with little data quality controls.

The client's desire was to compare scheduled maintenance work to unscheduled fault repairs. That is, they wanted to distinguish work that is expected and easily budgeted compared to unexpected work that is potentially avoidable. Any further information specifically about the type of component that failed would make the results more useful to the client.

To perform this analysis manually would be labour intensive. A subject matter expert to read through and classify twelve years of data relating to pump stations supporting more than two dozen dams. Instead of performing the analysis record by record, an investigation was started into whether the desired results could be obtained by through clustering and prediction of the free text data.

A sample of data consisting of 842 records from three pump stations was provided for this study. Given the small data set and low data quality, this study became an investigation into what results can be expected when usual standards of data quality and size are not met.

The analysis followed the CRISP-DM model (Chapman et al. 1999). First steps were to define the business issue and evaluate the input data available. The input data was found to be missing a target column, which was then coded by the authors to denote scheduled versus unscheduled jobs for training a classifier. Data was then prepared for data mining by spell checking,

combining the relevant text columns in to a single field, removing punctuation and replacing phrases to ensure their intention is maintained. Modelling involved determining the text weights and the corresponding singular value decomposition (SVD) components followed by the generation of text clusters. A series of classification models were then trained to predict the clusters or the scheduled jobs target variable. This paper presents the results of the classification models and set of findings from the study. The data mining tool set used in this case study was SAS Enterprise Miner 9.

2 Related Work

Prior to beginning the case study, previous examples of applications using text weights to classify documents were sought. A number of useful examples were found that reported useful accuracy for their application.

Kolyshkina and van Rooyne (2006) described an insurance claim cost prediction application. This application predicted if claims result in a top 10% pay-out using a combination of free-format narrative text and codified database fields. Their training data contained a binary target variable that identified if the claim was in the top 10%. Their mining process was iterative with much experimentation to find the optimal algorithm settings and required input from subject matter experts to derive domain relevant concepts. Overall, Kolyshkina and van Rooyne found that by combining text concepts from the free-format text with the codified database fields, prediction of the target insurance claim outcome was improved by 10% using a decision tree classifier.

Likewise, Drucker *et al* (1999) used text weights to train an email spam filter. Drucker evaluated the performance of a number of different classifiers on the task of distinguishing normal email messages from unsolicited spam. Best performance was an overall equal error rate of 0.0193 using a support vector machine (SVM).

Grivel (2005) applied text mining to categorise customer feedback on new cars obtained through phone survey results and transcribed phone calls. The system assigned documents into predefined and dynamically created categories respectively. Grivel claimed a 90% recall and precision.

The previous approaches used bag of words based approaches that analyse the frequency of words within text. In similar problems, Popowich (2006) and Grivel (2005) found that that structured linguistic analysis yields more accurate results than pure stochastic analysis. The linguistically driven process requires more input however from subject matter experts.

While linguistic analysis may be preferable, like many of the common tools currently available, the tool set used for this study (SAS Enterprise Miner 9) only supports the bag of words approached. This study aims to determine what is easily possible for this kind of problem today, hence will use the algorithms available in the tool set.

3 Problem Definition and Objective

The ability to accurately predict maintenance budgets is of financial importance to the client. To do this they need to know how often maintenance is required and how regularly systems fail. This information would help identify locations where improved maintenance would reduce the number of unexpected system failures.

Unfortunately the data required to accurately perform this analysis had been recorded in free text maintenance logs for the previous twelve years. Text data mining techniques were required to divide the data into scheduled maintenance jobs versus unscheduled repairs.

To achieve this task the data was first prepared by transforming into a format suitable for text analysis. Text mining methods were then applied to determine the text weights and create a feature vector based on the free text data. From the text weights cluster analysis was performed to identify describe natural groupings within the data. At this point we were specifically looking for fault and maintenance activities and groups that describe types of equipment. Next, a number of classification models were trained using decision trees and neural networks algorithms to learn the cluster groups or a target variable.

Finally, we assess how well the classification models allow the records to be classified as scheduled and unscheduled maintenance.

4 Input Data Description

A sample of data from three pump stations consisting of a total of 842 maintenance records was provided as a Microsoft Excel spreadsheet. Records from each pump station were stored in separate worksheets within the one spreadsheet. While the data in each worksheet contained effectively the same columns, small differences in column names and number of columns meant all three worksheets needed to be treated separately before extracting into a common format.

An initial assessment of the data was completed to identify data quality problems that impacted the analysis of the data. Problems identified included long text fields split over multiple columns, missing white space between words, spelling errors, inconsistent capitalisation and punctuation, inconsistent use of acronyms and terms, repeated words, and missing values in many columns.

The input data also lacked a target column to reliably differentiate scheduled maintenance jobs from unscheduled faults. The target column was required to train the classification models and to assess performance assessment of text clustering results. Ideally subject matter experts with knowledge of pump station maintenance and repair would be asked to categorise a sample of the data into either scheduled or unscheduled jobs. In this case, the authors have made a “best guess” at the meaning of the rows and created a simple text file with the unique identifier of each job alongside a target variable where 1 means scheduled maintenance and 0 identifies unscheduled maintenance. The distribution of this variable was split 65/35% between scheduled and unscheduled tasks respectively. The target column was coded to contain no missing values, which allows classification algorithms to use all data rows. But the data set contained noise in a form where the text is not clear. This required significant pre-processing of the data to get it ready for mining.

5 Data Preparation

The text mining algorithms require a data set with a single text column containing all the data to be analysed. The data preparation task involved all steps required to reformat the provided input data into this format.

5.1 Select Text Data

All text columns that described the type of maintenance or repair job were included in the analysis. Columns that identified the specific location, person performing the repair or the time the repair occurred were excluded to avoid the clustering and classification algorithms learning a specific repair job that could not be generalised to other pump stations.

5.2 Clean Data

Misspellings were common in the fault logs. A common error in the input data was words concatenated together where spaces should have been used to separate the words, e.g. “fixedthe” was typed where the words should be “fixed the”.

Since the data set was relatively small Excel’s spell checking feature was employed. Excel performs a reasonable job of detecting errors and proposing fixes. While the task required user guidance, fixing the spelling errors was completed quickly and accurately without requiring other external tools.

5.3 Construct Data

For text analysis, a single data file with all the relevant text in a single field is required. Constructing the data involved deriving a combined text column from the relevant text fields in the source data.

LONG_TEXT columns in the pump station data files contained the free text description of the job written by the maintenance worker. Each column was limited to 256 characters. Where the description of the job was longer than 256 characters the data was spread across multiple columns in Excel, only the first of which was labelled. There were effectively eight long text columns in each worksheet that need to be concatenated to recover the actual long text value.

Finally, all other text columns included in the analysis were concatenated with the combined long text column to form the into a single text column data set.

5.4 Integrate Data

Data from the pump stations was given in three separate Excel worksheets. These worksheets were combined into a single flat file table. Columns for each worksheet were identical requiring no special mappings to integrate the data. The hand coded target variable was then joined the pump station data using the unique identifier. These steps created a single table encompassing all data from all three pump stations.

5.5 Formatting Data

Formatting refers to syntactic transforms that do not change the meaning of the data, but assist the modelling algorithms. The first formatting step was to set all text to lower case. While not strictly necessary for text analysis, setting all fields to lowercase assists other formatting functions.

Punctuation in the text fields, particularly in the long text description, was often omitted or used extraneously. No consistent information was apparent in the use of punctuation. Therefore, to simplify the text mining all punctuation was removed and replaced with spaces. Specifically, the following characters were replaced:

~`!@#$%^&*()_+-={}[]|\;':'<>?/,.

Cause Text	Transformed to
no code	nocause
no cause code	nocause
no code - preventative maintenance	nocause - preventative maintenance
no code - electrical trip	nocause - electrical trip
information unavailable from list	Information unavailable from list

Table 1: Cause Text transforms

Three short free text columns contained values that implied that a record was a maintenance or repair job. On their own they contained too many null values and inconsistent use of other values to accurately classify the records. These fields included a Damage Text field that indicated whether there was or was not damage to equipment, a Cause Text field that indicated the possible cause of a fault, and a Breakdown field that indicated if the job was a part of a larger job.

Values in these fields were typically single “yes” or “no” values. Inconsistent use meant the values “no code”, “no cause code” and “no cause code - preventative maintenance” all used in the Cause Text column to signify the same value. The Damage Text column used similar values replacing the word “cause” for “damage”. If left as-is and the text analysis performed on words rather than phrases, then the modifier “no” would be lost and the meaning of “cause” or “damage” may be misinterpreted. To avoid the loss of meaning, values that included the term “no” before the word “code” had the text between these two terms replaced with the single words “nocause”, “nodamage”, and “nobreakdown” for the Cause Text, Damage Text and Breakdown fields respectively. Examples of the cause text transform are shown in table 1.

Finally as part of the formatting functions, common phrases that comprise of more than a word were combined into a single word to assist the text analysis. Common phrases included “Pump station”, “Air condition” and “Programmable logic controller”. Frequently these terms were used inconsistently in the data due to the use of various forms of spelling, abbreviations and acronyms. Table 2 presents examples of the phrases replaced with concatenated words. Note that the list in table 2 was compiled knowing the words would be stemmed during later text analysis. Suffixes to the phrases such as “er” and “ing” were ignored at this point.

6 Text Mining and Clustering

The prepared input data lacked numeric or categorical fields suitable for traditional clustering and classification models. Instead, the text descriptions of pump station

maintenance jobs were transformed into vectors of term weights, which could then be used for clustering and classification. This task was performed using SAS Enterprise Miner's Text Miner component. SAS Text Miner creates three outputs:

1. Converts the free text into term weights
2. Performs singular value decomposition (SVD) on term weights
3. Creates text clusters

Input data was split into training and validation sets using simple random partitions since the distribution of the target variable was not considered skewed enough to require a stratified sample. Data volumes in each group were selected by balancing the training sets need to have sufficient data to create a reliable model against the validation sets need to have enough samples to provide a useful estimation of the model's performance. The final split was a training set containing 75% of the data or 632 records and a validation set with the remaining 25% or 210 records.

Phrase	Replace With	Reason
"pump station"	"pumpstation "	Phrase pump station as a single word to avoid confusion with the singular terms "pump" and "station".
" pstn "	" pumpstation "	Abbreviation for pump station.
"pump stat "	"pumpstation "	Abbreviation for pump station.
"air condition"	"aircondition "	Phrase air conditioner as a single phrase to avoid confusion with the singular terms "air" and "condition".
"air con "	"aircondition "	Abbreviation for air conditioner
"aircon "	"aircondition "	Abbreviation for air conditioner
"program logic controller"	"plc "	Expanded version of the acronym PLC, which is also widely used in the data
"program logic contr"	"plc "	Synonym for PLC
"programmable logic controller"	"plc "	Synonym for PLC

Table 2: Example replacements for common phrases

6.1 Text Mining

The SAS Text Miner component requires input in the form of a single text column from a tabular data. From this it performs all text processing functions from extraction of terms to the creation of clusters. The functions are broadly grouped into parse, transform and cluster steps.

Parse tokenizes the text into individual terms. During the parsing process the number of extracted terms is reduced by stemming, filtering by a stop word list, and combining terms using a synonym list.

During the transform step Text Miner creates numeric vectors from the text terms, thus converting the text into a format suitable for clustering or predictive mining. SAS Enterprise Miner calculates two useful feature vectors from text documents: the term frequency by inverse document frequency (tf×IDF) term weights plus the singular value decomposition (SVD) of the term weights. Tf×IDF weights (Grossman & Frieder 2004, p. 13) calculate a vector for each record with each element representing a single term identified in the parse step. Terms that are frequent within a document but rare across all documents are weighted highest. SVD dimensions (Grossman & Frieder 2004, p. 129) reduce the dimensionality of the tf×IDF vectors by projecting them onto a smaller set of dimensions.

To limit the complexity of the models, only the 100 highest weighted terms were kept. Terms in the top 100 that were irrelevant to the classification goal were excluded from the analysis by adding the terms to the stop word list. This was repeated until all terms in the top 100 were found relevant.

The goal of text clustering was to produce an interpretable number of clusters with a high level of intra-class similarity and a low level of inter class similarity. For the pump station data, clusters that clearly split the records into scheduled maintenance or fault repairs were preferred. Expectation maximisation clustering was used as this method returns an easy to interpret "bag of words" to describe the clusters. Clusters were based on SVD dimensions, the default, as the process of dimension reduction simplifies the work of the clustering algorithm.

The number of terms to describe the clusters was set to 30. At this number of terms the cluster descriptions contain the key terms with little overlap between the clusters. Initially, Text Miner was set to automatically determine the number of clusters. After many clustering attempts, the best separation of the target variable was found when the number of clusters was set to 14. This approach was based on one of Francis' (2006, p70) methods for determining the number of clusters.

The final output of the text mining was a database table containing all fields from the input data, the cluster label, the top 100 term weights, and the SVD dimension components. Descriptions of the final clusters are presented in section 8. While some clusters clearly indicated either the type of task or the equipment involved, the complete set could not conclusively determine if a record was a scheduled or unscheduled job.

7 Classification of Text Mining Results

The clusters provided potentially useful information to the client. Further to this information we wanted to predict the cluster of a new record based on the text mining term weights or SVD components.

First a decision trees was trained using the term weights. While the accuracy of the tree was not expected to be high as it uses one word at a time in classification, it was expected the decision tree would provide some insight into which words best described each cluster. The "cluster" decision tree attempts to predict which of the 14 natural clusters a new unseen record would fit into.

A second classifier was trained to predict the cluster groups, this time aiming for accuracy over interpretability. For this purpose a neural network was trained using the 17 SVD dimensions as inputs and the cluster groups as outputs.

As the clusters could not accurately determine the scheduled maintenance target variable, two more classifiers were trained to predict the target (scheduled maintenance or unscheduled maintenance). As per the cluster classification models, a decision tree was trained using the term weights as input while a neural network was trained using the SVD components. Using the term weights for the decision tree identifies terms that best predict the target variable. Using the SVD weights simplifies the training of the neural network by decreasing the number inputs, thereby increasing the accuracy of the model.

No.	Cluster Description	No. of Records (% of total)	
1	Various Maintenance	119	(18.8%)
2	Repair Jobs	64	(10.1%)
3	Discharge and Actuators	49	(7.8%)
4	Gantry Cranes	23	(3.6%)
5	Cooling Water Pumps	37	(5.9%)
6	Pump Impeller Maintenance	15	(2.4%)
7	Upgrade Projects	32	(5.1%)
8	Maintenance Tests	71	(11.2%)
9	PLC and Pumps	55	(8.7%)
10	Vibration Tests	43	(6.8%)
11	Flowmeter Servicing	50	(7.9%)
12	Battery Replacement	18	(2.8%)
13	Information Unavailable	34	(5.4%)
14	Fire Alarm and Ventilation Repairs	22	(3.5%)
Total		*632*	**(100.0%)**

Table 3: Cluster descriptions

8 Results

8.1 Cluster Descriptions

A set of 14 clusters were identified and described based on the common terms within the cluster, the equipment involved and the type of job. The cluster descriptions are shown in table 3. Of the 14 clusters, only 6 can be considered homogenous. These are gantry cranes (4), pump impellers maintenance (6), upgrade projects (7), maintenance Tests (8), vibration tests (10), and battery replacement (12). The remaining clusters were heterogeneous, containing jobs that covered multiple infrastructure components and a mixture of scheduled maintenance and repair work.

Figure 1 compares the text clusters to the scheduled maintenance target variable. Clusters 2 (29.7%), 9 (25.5%) and 11 (34) contain relatively low percentages of scheduled maintenance records indicating clusters containing predominantly faults. Looking at the percentages, all other clusters consist of a majority of scheduled maintenance records. Clusters 1 (73.9%), 3 (63.3%), 12 (66.7%) and 14 (72.7%) are too close to random chance to clearly class as scheduled maintenance. Since the total percentage of scheduled maintenance is 65.7%, then taking a random sample from the training set would lead to similar results. This leaves clusters 4 (78.3%), 5 (81.1%), 6 (100.0%), 7 (84.4%), 8 (87.3%), 10 (88.4%), and 13 (82.4%) that can be clearly considered scheduled maintenance. It can be concluded that 40.3% of records form the 7 clusters that can be classified as scheduled maintenance; 32.9% of records form the 3 clusters that can be classified as scheduled maintenance; and 26.8% of records form the 4 clusters that can not be deterministically classified.

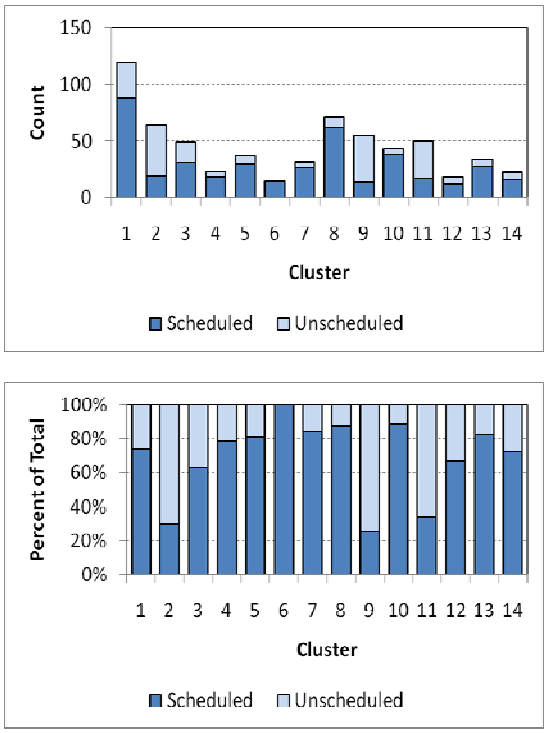

Figure 1: Clusters by scheduled maintenance

Model	Inputs	Output	Misclassification Rate	
			Training Set	Validation Set
Decision Tree	Term Weights	Cluster Labels	0.367	0.443
Neural Network	SVD components	Cluster Labels	0.120	0.167
Decision Tree	Term Weights	Target	0.148	0.150
Neural Network	SVD components	Target	0.147	0.172

Table 4: Misclassification rates on training and validation sets

8.2 Classification Results

Misclassification rates for the four classification models are shown in table 4. For the decision tree trained to recognise the cluster labels the misclassification rate on

the test set was 44.3%. This result is worse than random chance since selecting all records as scheduled would result in a misclassification rate of 34.3%, the total percentage of unscheduled repairs.

The problem search space was too large with 100 input terms used to learn 14 cluster labels and the search too restricted with small sample sizes and splitting on one term at a time for the decision tree to learn the mapping accurately.

It was hoped that the cluster decision tree would provide some insight into the words that best describe the cluster. As expected from the high misclassification rate suggests, the rules learnt are ambiguous. For example, table 5 shows the rules that split the cluster 9, moderately heterogeneous class regarding PLCs, and cluster 6, a completely homogenous cluster regarding pump impeller maintenance. It is not clear from a business context why a greater use of the word "pump" would lead to one class containing all scheduled jobs over a second class containing mostly unscheduled work.

Cluster 6	Cluster 9
IF 7.39 >= + pump AND + cool < 2.8 AND flowmeter < 1.81 AND + vibration < 2 AND + valve < 2 AND + center < 1.68	IF 3.10 >= + pump < 7.39 AND + cool < 2.8 AND flowmeter < 1.81 AND + vibration < 2 AND + valve < 2 AND + center < 1.68

Table 5: Examples of cluster decision tree rules

Scheduled Maintenance	Unscheduled Repairs
Leaf 16: n=124, 87% correct IF + reset < 2.61 AND high < 2.53 AND + damage < 1.60 AND + repair < 1.70	*Leaf 17: n=7, 57% correct* IF 2.601 >= + reset AND high < 2.53 AND + damage < 1.60 AND + repair < 1.70
Leaf 13: n=5, 60% correct IF 2.72 >= + initiate AND + sign < 2.11 AND 1.70 >= + repair	*Leaf 9: n=4, 100% correct* IF 2.53 >= high AND + damage < 1.60 AND + repair < 1.70
Leaf 14: n=2, 100% correct IF + damage < 1.60 AND 2.11 >= + sign AND 1.70 >= + repair	*Leaf 12: n=33, 91% correct* IF + initiate < 2.72 AND + sign < 2.11 AND 1.70 >= + repair
	Leaf 15: n=32, 78% correct IF 1.60 >= + damage AND 1.70 < + repair

Table 6: Examples of target decision tree rules

The other three classifiers all provided misclassification rates in the 15% to 17% range. Both neural network classifiers provided comparable misclassification rates when trained to learn the cluster labels, 16.7%, or the target variable, 17.2%, and using SVD values as inputs.

Interestingly, the best performance on this data set was obtained from the decision tree trained to learn the target variable using the term weights as input. For this classifier the misclassification rate was 15.0%. The difference between the decision tree and the neural network is more likely due to choice of input than the training algorithm. While the SVD components lead to a smaller problem for the neural networks to learn, dimension reduction smooths the data possibly loosing some descriptive detail.

Rules from the target variable decision tree provide a degree of comprehensibility (table 6). Leaf 16 predicts the most scheduled maintenance jobs and is associated with the low weights for the words "repair" and "damage". Similarly, leaf 12 predicts the high volume of unscheduled repairs and is associated with high weights for the term "repair".

9 Discussion and Conclusion

Misclassification rates around 15% to 17% are too high for many applications. Yet this rate can be acceptable for business decisions such as budgeting when compared to alternative decision making methods.

In context of the business objective, the text clusters found 14 distinct types of jobs. While these cannot clearly be categorised as scheduled or unscheduled from the cluster descriptions, some clusters were found to contain relatively homogeneous types of jobs. That is, they contained mostly one type of job or affected a specific type of equipment. Classification of these types of jobs can be predicted using the neural network cluster classification model. This would be useful if, for example, all battery replacements need to be identified.

Using a target variable for training scheduled maintenance versus unscheduled repairs could be predicted using a classification model. This case study tested a decision tree trained using term weights and a neural network trained on SVD components and obtained misclassification rates of 15.0% and 17.2% respectively. Therefore, with some input from subject matter experts, text features can be used to classify documents on small data sets with a moderate level of accuracy.

The case study provided a number of insights on how to use text data mining techniques on low quality data sets often found in spreadsheet applications.

- Formatting the data for text mining required appending many columns into a single text column. Semi-structured data values could be transformed by including a token, e.g. the column name, to identify the column. This ensured the context of the value was not lost when combined with other free text columns.
- Determining text mining stop words and common phrases replacements is an iterative process. Clusters needed to be recomputed until all of the top terms were relevant to the data mining goal and the clusters matched expectations.
- Clustering provided interesting classes that covered subsets of the desired task. Clusters did not provide mutually exclusive splits on desirable characteristics such as scheduled versus unscheduled jobs or by different types of equipment. The subsets provide useful information for the business, but use of clusters in this fashion is more opportunistic than by design.
- Classification models were able to learn both the cluster labels and a binary target variable using information derived from free text fields. This was

achieved with moderate levels of accuracy relative to the application. To train the classification models required manual classification of a subset of the data to create the target variable. This activity would best be performed by subject matter experts.

- When trained to learn the cluster labels a decision tree using text terms as inputs failed to learn the mapping. The problem search space was too large for the decision tree to learn one term at a time. A neural network using the SVD components as inputs learnt the mapping successfully.
- When learning a binary target variable the decision tree trained using text terms as inputs marginally outperformed a neural network trained using SVD components. The reduction to a single binary output variable simplified the problem search space allowing the decision tree to learn the mapping successfully. The process of dimension reduction, while simplifying the classification task, smooths detail in the inputs reducing the accuracy of the neural network.

This case study demonstrates that applying text data mining techniques in low quality data situations is viable provided the value of the data justifies the effort required applying text data mining.

10 Acknowledgement

We will like to sincerely thank CRC for Integrated Engineering Asset Management (CIEAM) to provide us the dataset to conduct this case study. We will also like to thank Colin Fidge and Lin Ma to support this research.

11 References

Chapman, P., Clinton, J., Khabaza, T., Reinartz, T., & Wirth, R. (1999): The CRISP-DM process model. Technical Report, Crisp Consortium. http://www.crisp-dm.org/. Accessed on 11 Jul 2008.

Drucker, H., Wu, D. & Vapnik, V. (1999): Support Vector Machines for Spam Categorization. *IEEE Transactions on Neural Networks*, **10**(5):1048-1054.

Francis, L.A. (2006): Taming Text: An Introduction to Text Mining. *Casualty Actuarial Society Forum*, Winter, 51-88.

Grivel. L. (2005): Customer feedbacks and opinion surveys analysis in the automotive industry. In Zanasi, A. (ed). (2005): *Text Mining and its applications to intelligence, CRM and Knowledge Management*., WITpress.

Grossman, D. & Frieder, O. (2004): *Information Retrieval: Algorithms and Heuristics*. 2nd edn., Springer.

Kolyshkina, I ., & van Rooyen, M. (2006) Text Mining for Insurance Claim Cost Prediction. In G.J. Williams, & S.J. Simoff (Eds.), *Data Mining LNAI* 3775, Berlin, 192-202, Springer-Verlag.

Popowich, F. (2005) Using Text Mining and Natural Language Processing for Health Care Claims Processing. *SIGKDD Explorations*, **7**(1):41-48.

Rayid, G., Probst, K., Liu, Y., Krema, M. and Fano, A. (2006) Text Mining for Product Attribute Extraction. *SIGKDD Explorations*, 8(1), pp41-48.