Predicting Time-to-Failure of Industrial Machines with Temporal Data Mining

Jean Nakamura

A dissertation submitted in partial fulfillment
of the requirement for the degree of

Masters of Science

University of Washington

2007

Committee:
Committee Chair: Isabelle Bichindaritz, Ph.D.
Committee Member: Don McLane

Program Authorized to Offer Degree: Institute of Technology - Tacoma

University of Washington

Abstract

Predicting Time-to-Failure of Industrial Machines with Temporal Data Mining

Jean Nakamura

Chair of the Supervisory Committee:
Professor Isabelle Bichindaritz
Computing and Software Systems

The purpose of this project is to perform analysis of temporal vibration data results to predict the time until a machine failure. The project performs temporal data mining, which is a method of choice to predict future events based on known past events. The difficulty in determining time-to-failure (TTF) of industrial machines is that the failure mode is not a linear progression. The progression of a severity of a fault increases at a higher rate as the machine approaches failure. Through experience, it is know that discrete frequencies in the vibration spectra are associated with machine faults and will reach expected amplitudes at the point of machine failure. Currently, there are no commercially available tools to predict machine time to failure. This project determined that it is possible to analyze a machine's temporal vibration data results to produce an estimated time to a failure based on the progression of identified faults when there is a repetition of instances with good accuracy (80%) using discretized data, but not using raw continuous data. Results of the data mining project are provided to support this conclusion.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

My gratitude extends to those at DLI Engineering for their professional experience and knowledge and to DLI Engineering for providing a database and engineering expertise for this project. Special thanks go to Dr. Isabelle Bichindaritz for guiding and assisting me throughout the project. Her knowledge and commitment were essential in putting this project together.

Chapter 1
INTRODUCTION

Machine Condition Analysis software is a predictive maintenance tool that analyzes vibration data. The process includes the collection of vibration data, the analysis of data resulting in generating faults, and the prescription of repair recommendations to avoid machine failure. "A fault is an abnormal state of a machine or system such as dysfunction or malfunction of a part, an assembly, or the whole system" [9]. Each fault is associated with a severity indicating its degree of seriousness. An increase in severity indicates a progression to machine failure. Predicting machine failure allows for the repairing of the machine before it breaks, saving cost and minimizing machine downtime.

Experts in the field are able to conclude that machine failure can occur in days to weeks, weeks to months, or months to years. Experts know rules and have an accumulation of experiences [1]. "When a new experience takes place, it isn't simply added to the data base of prior experiences. Learning from experience means changing what you know to accommodate the new knowledge" [1].

A piece of information that is lacking in machine predictive maintenance is a good estimation of Time to Failure (TTF) [7]. This project will determine TTF of a single machine by reviewing a progression of its own developing faults. This project will also use case-based reasoning (CBR) to determine TTF of similar machine configurations. The aspects that can affect TTF are the severity of the specific faults generated, the Machine Identification (MID) which is the machine configuration, and other external influences.

Machine faults start with a very gradual slope, and over time, as the severity of these faults increase, the slope rises sharply. Machine failure progresses at very different rates. Machine TTF is not linear [3]. This project will determine the rate of severity change for faults and adjust the rates accordingly as the dataset changes. The system will learn over time to predict the best estimation of TTF of a specific machine based on early identification of a machine anomaly.

Another aspect that can assist in determining TTF is the analysis of data based on similar machinery. The DLI Engineering database contains MIDs which group machines with common model numbers and applications. With this feature, MID grouped machine faults can be analyzed together to provide more experience to the TTF analysis.

External influences also affect machine TTF where the information is not available for consideration. For example, machines not maintained properly or that run seven days a week, 24 hours a day will reach failure at a much faster rate than a machine that runs a few hours a day and is maintained properly.

DLI Engineering is sponsoring this project by providing the database and engineering expertise. DLI Engineering is a marine and mechanical engineering company performing vibration and acoustic consulting.

Chapter 2
BACKGROUND INFORMATION

This project affects machinery maintenance, utilizes results from Machine Condition Analysis software, and applies data mining and CBR concepts.

2.1 Machinery Maintenance

Machinery maintenance practices have changed greatly over the years. Originally, a machine would fail (run-to-failure) before maintenance is performed. This type of maintenance is sometimes called "crisis maintenance" [12]. Then machines with no problems had preventive maintenance performed according to some schedule improved machine uptime. Now, with predictive maintenance, early identification of machine faults results in maintenance being performed before failure [12]. With a TTF estimate, maintenance can be scheduled at the most efficient and convenient time.

The most important goal of any maintenance program is the elimination of machine breakdowns [12]. Very often, a catastrophic breakdown will cause significant peripheral damage to the machine, greatly increasing the cost of the repair. The second goal is to anticipate and plan for maintenance needs [12]. This enables planning for down time, ordering of parts, and scheduling appropriate staff. The third goal is to increase plant production by reducing machine breakdowns during operations [12]. Predicting TTF can assist in achieving these goals.

2.2 Machine Condition Analysis software

The Machine Condition Analysis software includes an Expert System that is a "Rule-based" system that generates a multitude of information. The Expert System is run against a "test," a set of vibration data associated with a data collection timestamp. Each test may have

multiple Expert System runs where a vibration engineer may choose to change machine setup parameters. The last Expert System run is the record of choice by the engineer. In other words, all prior runs are ignored. This information used in this TTF System includes messages, severity, and diagnosis. The messages returned may indicate problems with data collection and therefore with enabling the TTF system to flag those records as "Invalid". The severity is the degree of the diagnosis. A severity < 101 is "Slight," 101-300 is "Moderate," 301-600 is Serious, and > 600 is "Extreme." In this TTF System, an Extreme severity is considered a machine failure. The diagnosis describes the machines problem. Throughout this document, the words diagnosis and fault are used interchangeably.

2.3 Data Mining

Like many areas of industry, data is collected but very little is converted to useful information. Data mining is extracting or "mining" knowledge from large amounts of data [8]. Data mining is very important with the ever-growing amounts of stored data. Data mining automates the analysis of large volumes of data to produce useful information [2]. Data mining is an application or interface of statistics and pattern technology and concerned with secondary analysis of data to predict future behavior [2] [4]. There are two kinds of models, predictive and descriptive. The predictive model makes predictions based on historical data. This model may be able to determine which customers would be interested in a new product. The descriptive model summarizes data illustrating subgroups of data. Data Mining consists of different approaches and algorithms. The approach is the algorithm that ties together the relevant data. The algorithm is the technique used to associate data: statistics, clustering, trees, neural networks, instance-based learning (case-based), and so forth.

2.4 Case-Based Reasoning

CBR is a "…model of reasoning that incorporates problem solving, understanding, and learning, and integrates all of them with memory process" [10]. "These tasks are performed using typical situations, called cases, already experienced by a system" [10]. Cases may also

be atypical, rare, or exceptional. CBR solves new problems by adapting solutions to older problems. When there is a new problem and an identical old case is found, the solution of the old case can be applied as a solution to the new problem. If an identical case is not found, an adaptation occurs and a solution is retrieved. "In case-based reasoning, the retrieval usually provides just the most similar case whose solution has to be adapted to fit the query course. As in compositional adaptation we take the solutions of a couple of similar cases into account" [11].

The four parts of a CBR system are retrieving, reusing, revising, and retaining. Retrieving is the part that returns an old case that is determined to be identical or similar to the new problem. "To carry out effective case retrieval, there must be selection criteria that determine how a case is judged to be appropriate for retrieval and a mechanism to control how the case base is searched" [10]. Reusing is the part that applies the solution of the retrieved old case, and adapts the retrieved solutions to solve the new problem. Revising is the step that corrects the adapted solution after user feedback. Lastly, retaining is the storing of valid confirmed cases.

Chapter 3
PROBLEM STATEMENT

This project will determine if it is possible to data mine temporal vibration data to produce an estimated time to a machine failure based on identified machine faults. There is a need in the predictive maintenance industry to develop a mechanism to estimate TTF. Temporal data mining is the analysis of a sequence of specific values changing over time. Data exists at irregular intervals. The goal is to use CBR to predict a future event. The REDI-PRO system, a naval application, used Figure of Merit (FOM) to monitor a system and determine remaining life of mechanical systems [5]. FOM is not available in the database evaluated in this system.

Chapter 4
DESCRIPTION OF DATA

The database contains vibration data analyzed through the Expert System resulting in severities and their associated diagnoses used in this system. It contains nearly 10,000 machines and over 142,000 tests. The database includes tests dating back to 1981. A partial database schema is included in Appendix A.1.

Two new tables, VibrationCase and VibrationCasetest, were created to store the case library. Each case is uniquely identified by a CaseID in the new VibrationCase table. This table represents each unique case in the system. Each case is associated with at least three test records stored in the new VibrationCaseTest table. VibStandardEquipmentID is a nominal value representing the machine configuration. VibStandardDiagnosisID is a nominal value representing the diagnosis. DiagnosisGroupID is a nominal value representing the diagnosis group. VibStandardEquipmentID is a nominal value representing the machine configuration group. TotalDaystoFailure is a real value that represents the total days to failure for the case. The CaseTestID is a unique identifier for each test in a case. DayPosition is a real value representing the day the test occurred in the case. VibStandardSeverityIndex is a real value that represents the severity of the diagnosis in the test. VesselID is a unique identifier for a site, EquipmentID is a unique identifier for a Machine, and TestResultID is a unique identifier for a test. VesselID, EquipmentID, and TestResultID are nominal values that associate the test back to the rest of the database. These values are keys that are not used in the analysis of the CBR system. CaseIsActive and CaseTestIsActive are nominal value that flags a case or case test that should or should not be used in the CBR system. These tables are displayed in table 4.1 and 4.2.

Table 4.1 VibrationCase table

VibrationCase			
Column Name	Key	Type	
CaseID	Primary Key	Integer	Nominal
VibStandardEquipmentID		Integer	Nominal
VibStandardDiagnosisID		Integer	Nominal
DiagnosisGroupID		Integer	Nominal
VibStandardEquipmentGroupID		Integer	Nominal
TotalDaysToFailure		Integer	Real
CaseType		Integer	Nominal
CaseIsActive		Small Int	Nominal

Table 4.2 VibrationCaseTest table

VibrationCaseTest			
Column Name	Key	Type	
TestCaseID	Primary Key	Integer	Nominal
VesselID	Foreign Key	Integer	Nominal
EquipmentID	Foreign Key	Integer	Nominal
TestResultID	Foreign Key	Integer	Nominal
CaseID	Foreign Key	Integer	Nominal
DayPosition		Integer	Real
VibStandardSeverityIndex		Integer	Real
CaseTestIsActive		Small Int	Nominal

An example case is as follows:

@attribute CaseID nominal
@attribute VibStandardEquipmentID nominal
@attribute VibstandardDiagnosisID nominal
@attribute Casetype nominal
@attribute TotalDaysToFailure real
@attribute DiagnosisGroupID nominal
@attribute VibStandardEquipmentGroupID nominal
@attribute VesselID nominal
@attribute EquipmentID nominal
@attribute DayPosition real
@attribute VibDiagnosisSeverityIndex real
@attribute TestCaseID nominal

33,879,395,0,267,4,82,6,6253,0,0,147
33,879,395,0,267,4,82,6,6253,139,130,148
33,879,395,0,267,4,82,6,6253,267,640,149

Table 4.3 Case Example

Attribute	Value	Description
CaseID	33	Unique key for case
VibStandardEquipmentID	879	Cargo Refrigeration Compressor
VibstandardDiagnosisID	395	Motor Bearing Looseness
CaseType	0	Type of case
TotalDaysToFailure	267	Number of days to failure for the case
DiagnosisGroupID	4	Looseness
VibStandardEquipmentGroupID	82	Motor driven reciprocating compressor no ball bearings
VesselID	6	Confidential
EquipmentID	6253	No.1 Cargo refrigeration compressor
DayPosition	0, 139, 267	Day position of the test within the case
VibDiagnosisSeverityIndex	0, 130, 640	The degree of the diagnosis
TestCaseID	147, 148, 149	Unique key for each test included in the case

Chapter 5
DATA PRE-PROCESSING

The database requires modifications to incorporate TTF functionality. Preprocessing includes new tables, new columns to existing tables, grouping associations of MIDs and Diagnoses, and identifying bad test data. Appendix B includes images of the tool used to assist in this processing.

The Mid Grouping tab displays all MIDs and MID groups in the database. Different icons are used to indicate whether an MID is already included in a group. To create a new MID group, the user enters a group name and associates which MIDs to include in the group (see Appendix B.1). This same methodology applies to diagnosis groups (see Appendix B.2). The Selection tab is simply a query tool that assists the user in determining which diagnosis or MID belongs to a specific group (see Appendix B.3). The last Tab is for miscellaneous functions. It currently only includes a button to 'exclude' bad tests.

5.1 Invalid Data

Data collected improperly results in invalid data being stored in the database. The Expert System produces the following messages when run on invalid data: "Questionable data at pickup…," "Manual review of the spectra is warranted," and "Unavailable pickups…". The invalid records are excluded in this system's analysis. A column VibResultInvalid (small integer) was added to the VibrationAnalysisExpertSystem table. The VibResultInvalid column contains "-1" for invalid data and "0" otherwise.

5.2 Grouping

Diagnoses and MIDs have been grouped where the diagnoses basically define the same problem and where the MIDs are basically describing the same machine configuration. The database contains 656 diagnoses, some of which are very similar. These similar diagnoses have been grouped and stored in a new table, DiagnosisGroups. This table contains two columns: DiagnosisGroupID and VibStandardEquipmentGroupDesc as seen in table 5.1. A column, DiagnosisGroupID was added to table VibrationStandardDiagnosis to associate similar diagnoses. An example of a diagnosis group is displayed in table 5.2.

Table 5.1 DiagnosisGroups table

DiagnosisGroups		
Column Name	Key	Type
DiagnosisGroupID	Primary Key	Integer
DiagnosisGroupDesc		Varchar(80)

Table 5.2 Diagnosis Group Example

Diagnosis Group 1	Diagnosis Group 2
Coupling Wear	Ball Bearing Noise
Coupling Wear or Looseness	Ball Bearing Wear
	Bearing Wear or Defect

The database contains over 1900 MIDs some of which are very similar. These similar MIDs have been grouped and stored in a new table, VibrationStandardEquipmentGroups. The table contains two columns: VibrationStandardEquipmentGroupID PK integer and VibrationStandardEquipmentGroupDesc varchar(80) as seen in table 5.3. A column, VibrationStandardEquipmentGroupID, was added to table VibrationStandardEquipment to associate similar MIDs. An example of a MID group is displayed in table 5.4.

Table 5.3 VibrationStandardEquipmentGroups table

VibrationStandardEquipmentGroups		
Column Name	Key	Type
VibrationStandardEquipmentGroupID	Primary Key	Integer
VibrationStandardEquipmentGroupDesc		Varchar(80)

Table 5.4 MID Group Example

Group 1	Group 2
AC Chill Water Pump	Air Conditioning Salt Water Circulating Pump
A/C Chill Water Pump	Air Conditioning Sea Water Service Pump
Air Conditioning Chill Water Pump	A/C S/W CIRC PUMP
Air Conditioning Chilled Water Pmp	

MIDs are basically grouped in terms of the driver components: motors, turbines, diesels, and so forth and how it is connected to the driven component: geared or belted, or driven (direct drive). The driven component consists of rotary screw, centrifugal, piston, and so forth and the driven component consists of pump, compressor, fan, blower, and so forth. The groupings are also based on the type of fluid pushed: hydraulics, lube oil, fuel oil, water, and so forth, and speed where high speed is anything around 3000 – 3600 rpm.

There are some other separators; such as if the motor has no fan or the driven machine has no ball bearings. They are noted in the group title.

5.3 Pre-Processing Results

Two copies of the database were preprocessed. One copy, Database 1 contains tests up through 11/28/2005. A second copy of the database, Database 2 contains tests up through

09/13/2006. Database 2 contains about 17% more tests and cases used for evaluation. Table 5.5 displays the details of these counts in the two databases.

The pre-processing resulted in about 49% of the test with valid Expert System runs with faults, greater than 96% of the MIDs belong to a MID group, and greater than 57% of the Diagnosis belongs to a Diagnosis group. Database 2 contains 16.8 more valid Expert System runs with faults. Detailed results are displayed in table 5.5.

Table 5.5 Pre-processing results

	Database 1	Database 2	Difference	Percent of Database 1 to Database 2
Tests	142,324	169,713	27,389	83.86%
Valid Expert System runs	128,965 (90.61%)	154,808 (91.21%)	25,843	83.31%
Valid Expert System runs with Faults	69,228 (69228 / 142324 =48.64%)	83,207 (83207 / 169713= 49.03%)	13,979	83.20%
Total MIDs	1917	2080	163	92.16%
MID Groups	114	116	2	98.28%
MIDs in Group	1852 (1852 / 1917 = 96.6%)	2058 (2058 / 2080 = 98.9%)	206	90.00%
Total Diagnoses	656	659	3	99.54%
Diagnosis Groups	11	11	0	100%
Diagnoses in Group	380 (380 / 656 = 57.9%)	380 (380 / 659 = 57.7%)	0	100%

Chapter 6
TIME TO FAILURE SYSTEM

This project predicts TTF based on analysis of existing temporal data. The most important features of this system include the TTF Database Setup Tool to perform data preprocessing, the Build Case Tool to build cases, the TTF user interface that displays TTF on the selected machine and the database for storage of the case library.

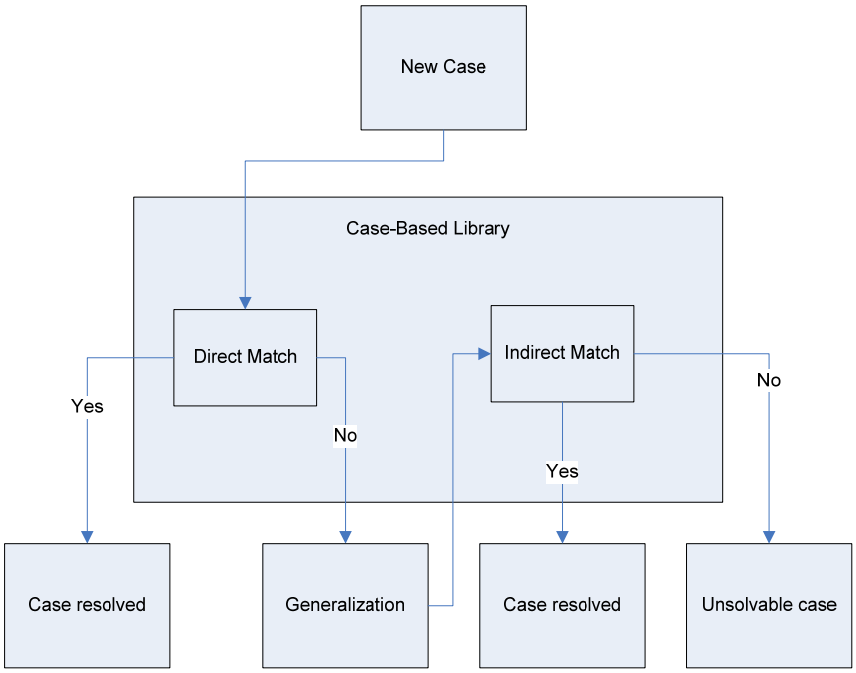
This system does not determine remaining life of a machine running in a normal state. If a machine has no faults, the analysis will result in "Insufficient data to determine TTF." The system begins its analysis from the first fault identified. If a machine is repaired and returns to a normal running state, the system will return "Insufficient data to determine TTF."

This system does not specifically indicate machine failure. An extreme fault will be the indicator of imminent machine failure.

6.1 Case-Based Reasoning System

This project uses CBR to provide solutions to new problems from previous cases (see figure 6.1). In this project, a case will be defined as a machine with tests beginning from a normal state and running to a critical state. When solving a new problem, the system will first attempt to make a direct match by searching for a previous case from the case library, for a machine with the same MID as the current problem. If one is found, the solution is applied and the problem is solved. If a direct match is not found, a generalization step is performed to try to find an indirect match.

Figure 6.1 Case base program flow



6.2 Generalization

When the database was initially implemented, an MID associated with machines was site specific. This site specificity caused duplication of MIDs within the database. These MIDs across sites are very similar but may contain slight variations. These MIDs across sites are grouped for use in finding indirect case matches. For example, the MIDs "AC Chill Water Pump," "A/C Chill Water Pump," and "Air Conditioning Chill Water Pmp" have been grouped to belong to one MID group.

A variety of diagnoses generated by the Expert System can imply the same problem. These similar diagnoses are grouped and used in finding of indirect case matches. For example, the diagnoses "Ball bearing wear," "Ball bearing noise" and "Ball bearing wear or defect" belongs to one diagnosis group.

6.3 CBR – retaining

Each case consists of a minimum of three consecutive tests. The tests in a case consist of one in which the machine had a severe fault, one in which the fault did not exist and one in which the fault exists but is less than severe. There are four types of cases. Case type 0 consists of a minimum of three consecutive tests on a machine where the last test has an extreme fault and the first test does not have the fault. Case type 1 consists of a minimum of three consecutive tests on a machine where at last test has an extreme grouped fault and the first test does not have the fault. Case type two and three are built on the same criteria with the exceptions listed in table 6.1.

Table 6.1 Case base logic

Case type	Description
Type 0	Same diagnosis and same machine
Type 1	Grouped diagnosis and same machine
Type 2	Normalize same diagnosis within the grouped MID
Type 3	Normalize grouped diagnosis within the grouped MID

6.4 CBR – retrieval

Cases are retrieved based on the different types of cases stored in the case library. An instance of a direct case is a case on the same machine with the same fault but does not include the current test instance. An instance of an indirect case is one of the following: same diagnosis on one of the grouped MIDs, grouped fault on the same MID, or grouped fault with one of the grouped MIDs. The specific instances of the case retrieval types are listed in table 6.2.

The system first attempts to find a direct match (1). If a case is not found, the system attempts to find an indirect match (2), and continues through the match types until a case is found. When a case is found, the solution is applied to the new problem.

If multiple cases are retrieved, the system selects the best case by finding a case that is failing at the closest rate as the new problem. It takes the severity and normalized date and selects the case with the closest severity to the normalized date. This is an attempt to find a case that is failing at the same rate as the new problem.

With an attempt to improve accuracy, the algorithm was tweaked to retrieve 2-nearest neighbor and 3-nearest neighbor. The system would average the TTF values of the two or three nearest neighbor cases.

Table 6.2 Case Base retrieval logic

Match type		Case type	Description
1	Direct	0	Same diagnosis and same machine
2	Indirect	0	Same diagnosis, same MID and different machine
3	Indirect	0	Same diagnosis, MID group and different machine
4	Indirect	0	Diagnosis group, MID group and different machine
5	Indirect	1	Grouped diagnosis and same machine
6	Indirect	1	Grouped diagnosis, same MID and different machine
7	Indirect	1	Grouped diagnosis, MID group and different machine
8	Indirect	2	Same diagnosis, MID group and different machine
9	Indirect	2	Diagnosis group, MID group and different machine
10	Indirect	3	Diagnosis group, MID group and different machine

6.5 CBR – reusing

After the system retrieves a case, this known solution is applied to the new problem. The test date of the new problem is normalized, applied to the previous solution and the new solution, TTF, is calculated. Normalization of date is done by calculating the number of days from the current test to the first prior test where the diagnosis in question is not present. For example, let D_1 equal to the date of the current test where a specific diagnosis exists. Let D_2 be the date of the first prior test where the specific diagnosis does not exist and let D_{result} be the calculated normalized date, in days. $D_{result} = D_1 - D_2$. TTF is calculated by subtracting D_{result}

from the TTF from the previous solution, TTF_{case}. The solution to the new problem is TTF_{result} - D_{result}.

6.6 CBR – revising

A user may review all cases in the case library and evaluate for correctness. The user may decide to flag invalid cases or individual case tests. This allows the user control in excluding invalid cases or individual case tests from being used during TTF determination. This is done in the user interface displayed in Appendix D.3.

Chapter 7
ANALYSIS

This section includes the analysis of the CBR system as well data analysis using SPSS 15.0 for Windows by Lead Technologies, Inc. and WEKA 3.4.10, the University of Waikato. Both SPSS and WEKA are data mining tools providing a variety of data mining algorithms.

7.1 CBR

The system was evaluated on two copies of the database. One copy, Database 1 contains tests up through 11/28/2005. A second copy of the database, Database 2 contains tests up through 09/13/2006. Database 2 contains about 17.5% more cases in the case library. Table 7.1 displays the details of these counts in the two databases. Raw data results from the VibrationCase and VibrationCase table are listed in Appendix C.1 and C.2.

Table 7.1 Case Library

Case	Database 1 (training)	Database 2	Difference (test)	Percent Database 2 Cases
Case type 0	355	430	85	17.44%
Case type 1	39	40	1	2.5%
Case type 2	53	85	32	37.65%
Case type 3	39	61	22	36.07%
TOTAL	486	616	130	21.10%

The TTF accuracy was based on two (<=62 days) and three months (<=93 days). The case library was analyzed to determine the TTF precision of each MID and diagnosis combination. A comparison was done on Case Type 0 records. The difference between the minimum TTF and the maximum TTF with the same MID and same diagnosis ranged from 2

to 1785 days and an average difference of 488 days. The majority of the cases have a standard deviation less than 250 days. This is a very broad range for the same machine. There were only 7 instances where the difference was less than 93 days. Some of these cases have very long TTF values. These values are displayed in Appendix C.3 and a standard deviation distribution chart is displayed in Appendix C.4. This may be due to some repairs being performed on a machine and postponing TTF resulting abnormally large TTF values. The repairs may affect the cases built and would affect the TTF values.

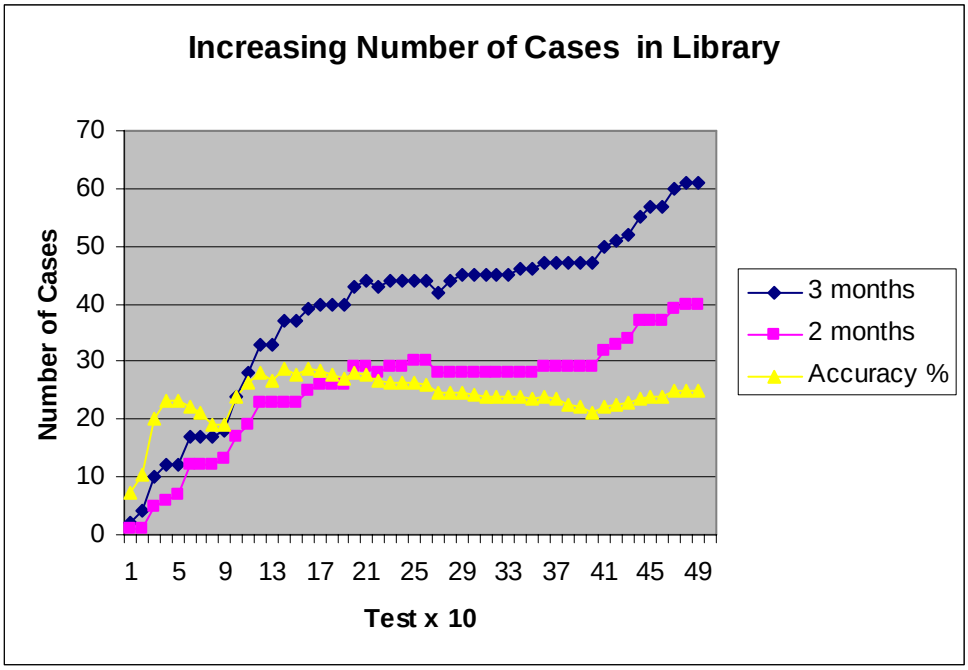
The first run of the TTF system was on the training set which produced results as expected. It generated an accuracy rate of 98.15%. The training results are displayed in Appendix C.5. The next was run based on the system knowing the TTF. If the system retrieved multiple cases it chose the case based on the known TTF. This run produced the best results the system can produce at less than 50%. See appendix C.6. The run resulted in 192 of the 478 new problems found no case matches, 193 new problems found one case match and 93 new problems found multiple cases. The accuracy of each case type selected showed type 1 at 100%. This is promising, but it only contained two matches so it was truly difficult to evaluate (see Appendix C.7).

The first real test of the TTF system was quite disappointing. It produced only a 25% accuracy rate (see Appendix C.8). The run resulted in 213 of the 413 new problems found no case matches, 78 new problems found one case match and 182 new problems found multiple cases matches. The breakdown of the accuracy based on case type selected did not show any promising results (see Appendix C.9). In an attempt to improve the accuracy, runs were performed using 2-nearest neighbor and 3-nearest neighbor. The accuracy of 2-nearest neighbor showed no significant difference at 24% accuracy (see Appendix C.10). The accuracy of 3-nearest neighbor decreased to 15% (see Appendix C.11). This shows the nearest neighbor is retrieving cases that are not similar enough to improve the accuracy.

The database contains only a small number of cases for each MID-diagnosis combination. The database contains 270 cases which have a unique MID-diagnosis combination, and 115 cases make up 41 MID-diagnosis combinations where the MID-diagnosis combination is included of more than one case. This results in about 50% of the new cases which have no match.

A test was derived to determine if increasing the number of cases in the case library would improve accuracy. The fist test used 10 cases, the second used 20 cases, and so on. This test displayed an increase in accuracy up until 50 cases were utilized. Then the accuracy plateaued between 20 and 30%.

Figure 7.1 Increasing number of cases in library



7.2 SPSS Linear Regression

SPSS linear regression was used to evaluate the relationship of the independent variables, VibrationStandardEquipmentGroupID, DiagnosisGroupID, VibrationStandardEquipmentID, VibrationStandardDiagnosisID and VibDiagnosisSeverityIndex that best predicts the value of the dependent variable, DayPosition. The analysis of the regression results was done using the Results Coach in SPSS to interpret the significance of the values. The Pearson correlation coefficient is a measure of linear association between two variables. The results range from 1 to -1 and the larger the absolute value, the stronger the relationship. The strongest relationship was with VibStandardSeverityIndex at .298 which is still not very strong a relationship. The

significance level (p-value) of <0.05 is significant and the two variables are linearly related. This is true for the relationships between DayPosition and DiagnosisGroupID, DayPosition and VibStandardEquipmentGroupID, VibDiagnosisSeverityIndex and DayPosition, VibStandardEquipmentID and DiagnosisGroupID, VibStandardDiagnosisID and DiagnosisGroupID, VibStandardEquipmentGroupID and DiagnosisGroupID, VibDiagnosisSeverityIndex and DiagnosisGroupID, VibStandardEquipmentGroupID and VibDiagnosisSeverityIndex. SPSS Correlations displayed in table 7.2

Table 7.2 SPSS Correlations

Correlations

		DayPosition	VibStandard EquipmentID	C. Vibstandard DiagnosisID	Diagnosis GroupID	VibStandard Equipment GroupID	Vib Diagnosis SeverityIndex
Pearson Correlation	DayPosition	1.000	-.024	.015	-.068	.089	.298
	VibStandardEquipmentID	-.024	1.000	.043	-.110	-.045	-.021
	C. VibstandardDiagnosisID	.015	.043	1.000	-.249	-.019	.025
	DiagnosisGroupID	-.068	-.110	-.249	1.000	-.228	.053
	VibStandardEquipment GroupID	.089	-.045	-.019	-.228	1.000	-.128
	VibDiagnosisSeverity Index	.298	-.021	.025	.053	-.128	1.000
Sig. (1-tailed)	DayPosition	.	.195	.298	.007	.001	.000
	VibStandardEquipmentID	.195	.	.062	.000	.053	.228
	C. VibstandardDiagnosisID	.298	.062	.	.000	.251	.186
	DiagnosisGroupID	.007	.000	.000	.	.000	.030
	VibStandardEquipment GroupID	.001	.053	.251	.000	.	.000
	VibDiagnosisSeverity Index	.000	.228	.186	.030	.000	.
N	DayPosition	1268	1268	1268	1268	1268	1268
	VibStandardEquipmentID	1268	1268	1268	1268	1268	1268
	C. VibstandardDiagnosisID	1268	1268	1268	1268	1268	1268
	DiagnosisGroupID	1268	1268	1268	1268	1268	1268
	VibStandardEquipment GroupID	1268	1268	1268	1268	1268	1268
	VibDiagnosisSeverity Index	1268	1268	1268	1268	1268	1268

R values range from 0 to 1 with larger R values indicating stronger relationships. This model produced and R value of only .330. R squared values range from 0 to 1 with larger values indicating that the model fits the data well. This system only produced a value of .109 which shows that the model does not fit the data well. See table 7.3 for the SPSS Model Summary.

Table 7.3 SPSS Model Summary

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics				
					R Square Change	F Change	df1	df2	Sig. F Change
1	.330[a]	.109	.105	363.014	.109	30.818	5	1262	.000

a. Predictors: (Constant), VibDiagnosisSeverityIndex, VibStandardEquipmentID, C.VibstandardDiagnosisID, VibStandardEquipmentGroupID, DiagnosisGroupID

b. Dependent Variable: DayPosition

The table 7.4 summarizes the results of an analysis of variance. The output for Regression displays information about the variation accounted for by this model and the output for Residual displays information about the variation that is not accounted for by this model. A very high residual sum of squares indicates that the model fails to explain a lot of the variation in the dependent variable.

Table 7.4 SPSS ANOVA

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	20306194	5	4061238.886	30.818	.000[a]
	Residual	1.7E+008	1262	131779.459		
	Total	1.9E+008	1267			

a. Predictors: (Constant), VibDiagnosisSeverityIndex, VibStandardEquipmentID, C. VibstandardDiagnosisID, VibStandardEquipmentGroupID, DiagnosisGroupID

b. Dependent Variable: DayPosition

Using the coefficients in the table 7.5, regression was applied using the formula:

DayPosition = 133.508 + (-.011 * VibStandardEquipmentID)
+ (-.015 * VibStandardDiagnosisID)
+ (-8.004 * DiagnosisGroupID)
+ (1.311 * VibStandardEquipmentGroupID)
+ (.266 * VibDiagnosisSeverityIndex)

Table 7.5 SPSS Coefficients

Coefficients(a)

Model		Unstandardized Coefficients		Standardized Coefficients	T	Sig.
		B	Std. Error	Beta		
1	(Constant)	133.508	46.836		2.851	.004
	VibStandardEquipmentID	-.011	.015	-.019	-.709	.479
	C.VibstandardDiagnosisID	-.015	.071	-.006	-.205	.837
	DiagnosisGroupID	-8.004	3.636	-.063	-2.201	.028
	VibStandardEquipmentGroupID	1.311	.319	.114	4.111	.000
	VibDiagnosisSeverityIndex	.266	.023	.316	11.768	.000

a Dependent Variable: DayPosition
Note: Only a partial table is displayed

This regression produced 133 of 478 correctly predicted problems, within three months, for a 27.8% accuracy. This is only a 2 % improvement over the case base method. Summary of results is listed in table 7.6.

Table 7.6 Result summary

Test – Database 1	Three months - Accuracy
Initial Test	25.39%
2-Nearest Neighbor	24.24%
3-Nearest Neighbor	15.00%
SPSS Regression	27.8%

7.3 Weighting Parameters

Another analysis of the data was done by changing the weighting of the attributes to determine the importance of these attributes on the accuracy of the system. The calculation was done based on the following formula where W_i is the weight for the matching attributes and $Wtotal_i$ is the weight of the attribute evaluated:

$$\frac{\sum W_i F_i}{\sum Wtotal_i}$$

The first test included the following attributes with a weight of 1:

VibStandardSeverityIndex (converted to nominal values of None, Slight, Moderate, Severe, or Extreme)
VibStandardEquipmentID
VibStandardEquipmentGroupID
VibStandardDiagnosisID
DiagnosisGroupID
DayPosition (converted to nominal values of <1month, 1-2months, 2-3months, 3-4months, 4-5months, 5-6months, 6-7months, 8-9months, 10-11months, 11-12months, and >12months)

The first test resulted in a 22.38% accuracy. The second test changed the DayPosition to nominal values of 3 month intervals, <3month, 3-6months, 6-9months, 9-12months, and

>12months. The 3 month interval test resulted in 25.31% accuracy. The third test changed the VibStandardSeverityIndex to None, Slight-Moderate, and Severe-Extreme, and retaining the 3 month intervals of DayPosition from the second test. This third test resulted in 25.73% accuracy. The next test, the DayPosition interval was changed to <3months and >3months and resulted in a 19.25% accuracy. With the third test having the best accuracy rate, these values were used to evaluate different combinations of weighting with the attributes evaluated. The best accuracy of 26.57% resulted when all attributes were used except for VibStandardEquipmentGroupID. The worst accuracy of 11.92% resulted when only VibStandardSeverityIndex and DayPosition were considered in the weighting. Summary of the weighting results are shown in table 7.7.

Table 7.7 Weighting Summary

Num	Severity	DayPosition	MID	MIDGrp	Dx	DxGrp	Accuracy
1	1	1	1	1	1	1	25.73%
2	1	1	0	0	0	0	11.92%
3	1	1	1	0	1	0	24.89%
4	0	0	0	1	0	1	23.64%
5	1	1	1	1	1	0	25.10%
6	1	1	1	0	1	1	26.57%
7	0	0	1	1	1	1	20.50%
8	0	0	1	0	1	0	20.08%
9	1	1	1	0	0	0	17.99%

Note:
Num = Weighted test number
Severity = VibStandardSeverityIndex
MID = VibStandardEquipmentID
MIDGrp = VibStandardEquipmentGroupID
Dx = VibStandardDiagnosisID
DxGrp = DiagnosisGroupID

The results in table 7.8 were produced by SPSS linear regression applying multi-step regression. It shows the low R value with the addition of each predictor.

Table 7.8 SPSS Multi-step regression – See Predictors

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.154[a]	.024	.023	312.827

a. Predictors: (Constant), VibDiagnosisSeverityIndex

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.146[a]	.021	.021	313.856

a. Predictors: (Constant), VibStandardEquipmentGroupID, VibDiagnosisSeverityIndex

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.139[a]	.019	.019	300.691

a. Predictors: (Constant), DiagnosisGroupID, VibDiagnosisSeverityIndex, VibStandardEquipmentGroupID

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.266[a]	.071	.068	408.944

a. Predictors: (Constant), VibStandardEquipmentID, VibDiagnosisSeverityIndex, VibStandardEquipmentGroupID, DiagnosisGroupID

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.327[a]	.107	.103	365.374

a. Predictors: (Constant), VibstandardDiagnosisID, VibDiagnosisSeverityIndex, VibStandardEquipmentID, VibStandardEquipmentGroupID, DiagnosisGroupID

7.4 WEKA Algorithms and dataset

All numeric attributes included in the analysis was discretized to nominal values. The dataset was modified where each row contains each test severity and days to failure at the current test. Sev2 represents the severity value for the second test, Sev3 represents the severity value for test 3, Sev4 represents the severity value for test 4, if available, Diff2 is the number of days to TTF calculated from the actual TTF minus the current day position of the test, and so forth. The dataset includes DiagnosisGroupID (DxGrpID), VibStandardEquipmentGroupID (MIDGrpID), Sev2, Sev3, Sev4, Diff2, Diff3, Diff4.

The different test options used are the training set, 10-fold cross validation, and ratio validation of 50%, 66%, 75%, and 80% training. Training set as a test set provides optimal classification accuracy. 10 fold cross-validation technique averages the classification results on ten different random samples generated from the dataset. It provides more robust results when there is only one dataset available. Cross-validation tries to diversify the samples in order to get a better estimate with fewer samples. It uses averages so the method is not as accurate. The ratio validation uses a different percentage of records for the training and test set.

The algorithms used and defined by WEKA are the J48 tree, ID3 tree, Multilayer Perceptron, Logistic regression, Apriori, Predictive Ariori, and K*. J48 generates a pruned or un-pruned C4 tree where values may be missing, attributes may be numeric, and can deal with noisy data. ID3 generates an un-pruned decision tree. Attributes must be nominal and there cannot be any missing values. Empty leaves may result in unclassified instances. Multilayer Perceptron is a neural network that uses back propagation to train. "Back propagation learns by iteratively processing a data set of training tuples, comparing the network's prediction for each tuple with the actual known target value" [8]. Logistic regression is used for building and using a multinomial logistic regression model with a ridge estimator. Apriori generates association rules from frequent item sets and Predictive Apriori finds association rules sorted by predictive accuracy. K* is an instance-based classifier, that is the class of a test instance is based upon the class of those training instances similar to it, as determined by some similarity function.

7.5 WEKA data discretized into 3 groups

The data set was discretized into three groups by equi-depth, meaning each group has the same number of cases. The range of values in each group was quite large as seen in table 7.9. The range for Diff3 in Group2 is 268 days or almost nine months.

Table 7.9 Three group data distribution

	Diff2	Diff3	Diff4
Group 1	-inf-118	-inf-.5	-inf-1
Group 2	118-340.5	0.5-268.5	1-237.5
Group 3	340.5-inf	268.5-inf	237.5-inf

Diff3 produced the best results with all algorithms tested; J48, K*, Multilayer Perceptron, and Logistic regression. Diff2 is significantly less accurate with all algorithms because with Diff2 there is only one data point analyzed, and the analysis is based on just the initial test. Of the algorithms tested, comparing the 10-fold Cross Validation results, J48 tree produced the best results for Diff4 and Diff3. Multilayer Perceptron produced the best result for Diff2. See table 7.10, 7.11, 7.12, and 7.13.

Table 7.10 J48 tree – three groups

J48	Diff4		Diff3		Diff2	
	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances
Training	221	75.1131 %	430	80 %	430	41.6279 %
10-fold	221	**72.8507 %**	430	**79.3023 %**	430	31.1628 %
50% train	107	71.9626 %	215	80 %	215	38.1395 %
66% train	77	70.1299 %	147	77.551 %	147	31.9728 %
75% train	54	77.7778 %	108	80.5556 %	108	34.2593 %
80% train	41	78.0488 %	86	83.7209 %	86	38.3721 %

Table 7.11 Multilayer Perceptron – three groups

	Diff4		Diff3		Diff2	
	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances
training	221	99.095%	430	96.0456%	430	74.6512 %
	221	65.6109%	430	74.186 %	430	**40 %**
50% train	107	65.4206%	215	71.6279 %	215	34.8837 %
66% train	77	61.039%	147	78.2313 %	147	33.3333 %
75% train	54	62.963%	108	77.7778 %	108	27.7778 %
80% train	41	65.8537%	86	77.907%	86	34.8837 %

Table 7.12 Logistic – three groups

	Diff4		Diff3		Diff2	
	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances
training	221	90.9502 %	430	88.3721 %	430	57.4419 %
10-fold	221	62.4434 %	430	70.9302 %	430	37.4419 %
50% train	107	54.2056 %	215	74.4186 %	215	35.814 %
66% train	77	62.3377 %	147	70.068 %	147	34.0136 %
75% train	54	70.3704 %	108	70.3704 %	108	30.5556 %
80% train	41	65.8537 %	86	74.4186 %	86	33.7209 %

Table 7.13 K* – three groups

	Diff4		Diff3		Diff2	
	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances
Training	221	97.7376 %	430	95.814 %	430	76.0465 %
10-fold	221	65.1584 %	430	73.7209 %	430	37.4419 %
50% train	107	57.0093 %	215	73.4884 %	215	37.2093 %
66% train	77	55.8442 %	147	72.1088 %	147	39.4558 %
75% train	54	72.2222 %	108	76.8519 %	108	38.8889 %
80% train	41	70.7317 %	86	76.8519 %	86	36.0465 %

7.6 WEKA data discretized into 3 groups with Sev and Diff only

This dataset used the attributes normalized into 3 groups but also excluded DxGrpID and MIDGrpID and included the Diff2, Diff3, Diff4, Sev2, Sev3, and Sev4. Of the algorithms

tested, comparing the 10-fold Cross Validation results, Logistic regression produced the best results for Diff4 and Diff3. Multilayer Perceptron continued to produce the best result for Diff2. See table 7.14 and 7.15.

Table 7.14 J48 tree – Sev and Diff only

J48	Diff4		Diff3		Diff2	
	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances
Training	221	75.1131 %	430	80 %	430	40.2326 %
10-fold	221	73.3032 %	430	79.0698 %	430	36.0465 %
50% train	107	71.9626 %	215	80 %	215	38.6047 %
66% train	77	70.1299 %	147	77.551 %	147	33.3333 %
75% train	54	77.7778 %	108	80.5556 %	108	34.2593 %
80% train	41	82.9268	86	83.7209 %	86	36.0465 %

Table 7.15 Multilayer Perceptron – Sev and Diff only

	Diff4		Diff3		Diff2	
	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances
training	221	83.2579%	430	81.3953 %	430	33.4884 %
10-fold	221	70.1357%	430	78.8372 %	430	38.8372 %
50% train	107	60.7477%	215	78.6047 %	215	38.6047 %
66% train	77	66.2338%	147	78.2313 %	147	35.3741 %
75% train	54	61.1111%	108	80.5556%	108	37.037 %
80% train	41	85.3659%	41	85.3659%	86	36.0465 %

Multilayer Perceptron options were modified in an attempt to improve results: learningRate changed to 0.1 and momentum changed to 0 rather than 0.3 and 0.2 respectively, nominalToBinaryFilter set to False rather than True, and set trainingTime to 600 rather than 500. These changes produced improved accuracy but Logistic regression still performed slightly better for Diff4 and Diff3. See table 7.16 and table 7.17.

Kappa statistics were recorded on the best results as displayed in tables 7.16 and 7.17. Kappa statistic compares the agreement against what is predicted and what is observed, correcting for agreement occurring by chance. The values range from 1 representing perfect agreement, 0 representing no agreement and 1 representing total disagreement. The Kappa statistic of 60-70% for Diff3 and Diff4 are much better than random classifier. Multilayer

Perceptron's Diff2 Kappa statistic was very low at only 0.086. Logistic regression Diff4 and Diff3 Kappa statistic was .6358 and .6697 respectively. Table 7.18 displays the results from K* but the results are not as good as that produced by Multilayer Perceptron and Logistic regression.

Table 7.16 Multilayer Perceptron –Sev and Diff only – modified options

	Diff4			Diff3			Diff2		
	Total # of Instances	Correctly Classified Instances	Kappa	Total # of Instances	Correctly Classified Instances	Kappa	Total # of Instances	Correctly Classified Instances	Kappa
training	221	78.733 %	0.6768	430	79.5349 %	0.6671	430	39.0698 %	0.0849
10-fold	221	75.5656 %	0.6288	430	79.5349 %	0.6642	430	**39.0698 %**	0.086
75% train	54	79.6296 %	0.6935	108	81.4815 %	0.6842	108	37.037 %	0.091

Table 7.17 Logistic – Sev and Diff only

	Diff4			Diff3			Diff2		
	Total # of Instances	Correctly Classified Instances	Kappa	Total # of Instances	Correctly Classified Instances	Kappa	Total # of Instances	Correctly Classified Instances	Kappa
training	221	77.3756 %	.6561	430	80.4651 %	.678	430	40.2326 %	.1037
10-fold	221	**76.0181 %**	.6358	430	**80 %**	.6697	430	36.0465 %	.0409
50% train	107	71.028 %	.5604	215	80 %	.6672	215	38.6047 %	.084
66% train	77	80.5195 %	.7064	147	79.5918 %	.664	147	35.3741 %	.0428
75% train	54	87.037 %	.8037	108	80.5556 %	.67	108	34.2593 %	.0373
80% train	41	85.3659 %	.7796	86	81.3953 %	.6764	86	36.0465 %	.0705

Table 7.18 K* –Sev and Diff only

	Diff4		Diff3		Diff2	
	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances
training	221	81.9005 %	430	80 %	430	40.2326 %
10-fold	221	71.4932 %	430	79.0698 %	430	36.0465 %
50% train	107	64.486 %	215	79.0698 %	215	38.6047 %
66% train	77	59.7403 %	147	77.551 %	147	35.3741 %
75% train	54	74.0741 %	108	81.4815 %	108	34.2593 %
80% train	41	80.4878 %	86	82.5581 %	86	36.0465 %

ID3 tree does not allow missing values. Due to missing values in Diff4, only Diff3 and Diff2 could be analyzed. ID3 did not produce any improved results (see table 7.19).

Table 7.19 ID3 tree –Sev and Diff only

	Diff3		Diff2	
	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances
Training	430	81.3953 %	430	40.2326 %
10-fold	430	76.9767 %	430	36.0465 %
50% train	215	74.4186 %	215	38.6047 %
66% train	147	77.551 %	147	33.3333 %
75% train	108	79.6296 %	108	34.2593 %
80% train	52	78.8462 %	52	42.3077 %

7.7 WEKA data discretized into 6 groups

The dataset is identical with the prior dataset except Sev2, Sev3, Sev4, Diff2, Diff3, and Diff4 was discretized into six groups rather than three. The range is smaller but is still quite large as seen in table 7.20.

Table 7.20 Six group data distribution

	Diff2	Diff3	Diff4
Group 1	-inf-72.5	-inf-.5	-inf-1
Group 2	72.5-119.5	0.5-93.5	1-93.5
Group 3	119.5-194.5	93.5-186	93.5-196.5
Group 4	194.5-340.5	186-337	196.5-321
Group 5	340.5-572	337-642.5	321-689
Group 6	572-inf	642.5-inf	689-inf

This test was only run using J48 against Diff4 because the results were significantly lower than when three groups were used so no further testing was performed.

Table 7.21 J48 tree – six groups

	Diff4		
	Total Number of Instances	Ignored Class Instances	Correctly Classified Instances
training data	221	209	66.9683 %
10-fold	221	209	59.276 %
Split 50% train	107	108	50.4673 %
Split 66% train	77	70	49.3506 %
Split 80% train	41	45	60.9756 %

7.8 Progression of Severity

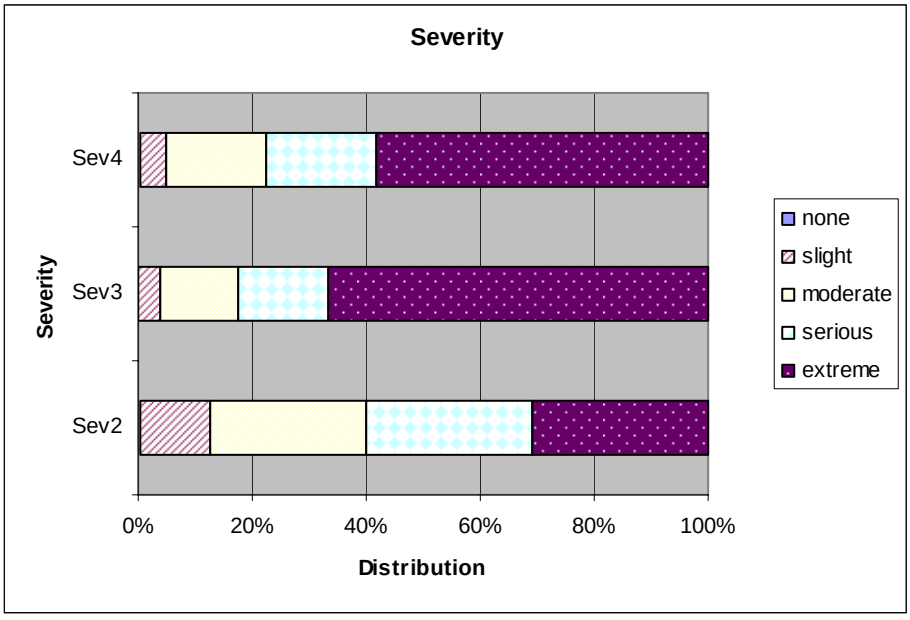
In order to determine if the progression of severity assists in determining TTF, another attribute was added to demonstrate a decrease, increase or stable severity. Sev_1-2 is the severity change between Sev1, Sev2, Sev_2-3 is the severity change between Sev2 and Sev3, and so forth. The data was analyzed to show the distribution of each type of severity in Sev2, Sev3, and Sev4. Sev2 contains the most “slight” records as compared to that in Sev3 and Sev4. Sev2 includes a comparable number of “moderate”, “serious”, and “extreme” records. Sev3

includes the most “extreme” records as compared to that in Sev2 and Sev4. Sev4 contains the most “extreme” records as compared to other records within Sev4. See table 7.22 and figure 7.2

Table 7.22 Severity distribution

	none	slight	moderate	serious	extreme	total
Sev2	2	52	118	125	133	430
Sev3	0	16	59	68	287	430
Sev4	1	10	39	42	129	221

Fig 7.2 Severity distribution graph



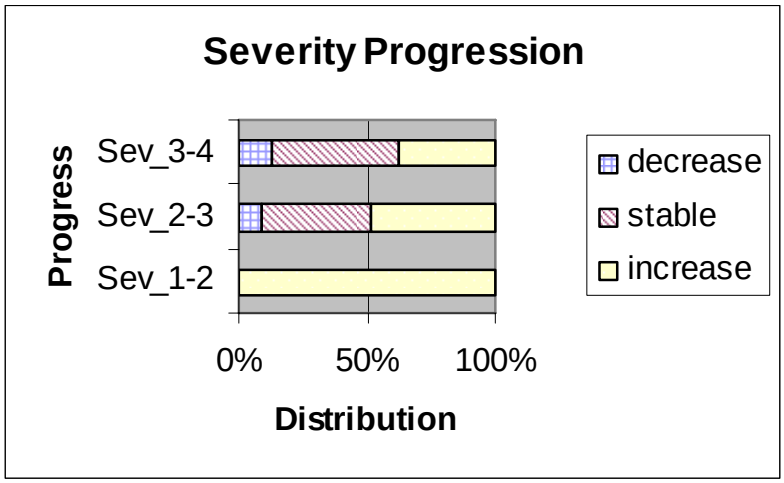
The data shows that the severity increases between Sev1 and Sev2 in almost all records and about 50% of the time between Sev2 and Sev3. Sev _3-4 has the highest percent of stable records. See table 7.23 and figure 7.3

Table 7.23 Severity progression

	decrease	Stable	increase	total
Sev_1-2	0	2	428	430
Sev_2-3	39	184	207	430

Sev_3-4	28	110	83	221

Figure 7.3 Severity progression graph



The dataset includes the attributes: Sev2, Sev3, Sev4, Diff2, Diff3, Diff4, Sev_1-2, Sev_2-3, and Sev_3-4. With J48 tree, Diff4 was better and Diff3 and Diff2 remained the same. With Multilayer Perceptron, modified options applied, Diff4 was worse, Diff3 was better, and Diff2 was identical when comparing results in table 7.16 Multilayer Perceptron – Sev and Diff only – modified results. With Logistic, Diff4 and Diff3 were worse and Diff2 was just slightly better when comparing with table 7.17 Logistic –Sev and Diff only. With K*, Diff4 and Diff3 was worse and Diff2 remained the same when compared to table 7.18 K* - Sev and Diff only. With this dataset, J48 had the best results for Diff4, Multilayer Perceptron and Logistic regression had the best identical results for Diff3 and Multilayer Perceptron had the best results for Diff2. The dataset with Sev and Diff, tables 7.16 and 7.17, produced the best overall results for Diff4 and Diff3 with the Logistic regression and Diff2 with the Multilayer Perceptron. The results from the progression analysis are displayed in table 7.24, 7.25, 7.26, and 7.27.

Table 7.24 J48 tree – progression

	Diff4		Diff3		Diff2	
	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances
training	221	78.2805 %	430	80 %	430	40.2326 %
10-fold	221	**74.6606 %**	430	79.0698 %	430	36.0465 %
75% train	54	77.7778 %	108	80.5556 %	108	34.2593 %

Table 7.25 Multilayer Perceptron – modified options – progression

	Diff4		Diff3		Diff2	
	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances
training	221	80.0905 %	430	80.4651 %	430	39.0698 %
10-fold	221	71.4932 %	430	**79.7674 %**	430	**39.0698 %**
75% train	54	71.4932 %	108	81.4815 %	108	34.2593 %

Table 7.26 Logistic – progression

	Diff4		Diff3		Diff2	
	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances
training	221	77.8281 %	430	80.6977 %	430	40.2326 %
10-fold	221	71.9457 %	430	**79.7674 %**	430	36.7442 %
75% train	54	77.7778 %	108	81.4815 %	108	34.2593 %

Table 7.27 K* – progression

	Diff4		Diff3		Diff2	
	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances
training	221	83.2579 %	430	80.6977 %	430	40.2326 %
10-fold	221	68.7783 %	430	78.6047 %	430	36.0465 %
75% train	54	68.7783 %	108	78.7037 %	108	34.2593 %

7.9 WEKA Associations - data discretized into 3 groups

The dataset includes the following attributes: Sev2, Sev3, Sev4, Diff2, Diff3, and Diff4. The rules produced by Apriori showed an association between Diff4 having a value in "-inf-1" and Sev4 being "extreme", which is a fact that when a Diff# is zero, Sev# is "extreme". The attribute Diff2 having a value in "118-340.5", the attribute Diff3 having a value in "-inf-.5" and Sev3 being extreme, shows there is a decrease in value from Diff# to Diff#-1 progressing to a Severity of "extreme". Rule 3 showed that if Sev2 is "extreme" and Diff3 is zero, Sev3 is

"extreme", Rule 4 shows that is Diff4 is large, Diff2 is also large, and Rule 5 shows that if Diff3 and Diff4 are large, Diff2 is also large. Predictive Ariori produced similar results listed below.

Apriori - Best rules found:

1. Diff4='(-inf-1]' 83 ==> Sev4=extreme 83 conf:(1)
2. Diff2='(118-340.5]' Diff3='(-inf-0.5]' 75 ==> Sev3=extreme 75 conf:(1)
3. Sev2=extreme Diff3='(-inf-0.5]' 73 ==> Sev3=extreme 73 conf:(1)
4. Diff4='(237.5-inf)' 69 ==> Diff2='(340.5-inf)' 69 conf:(1)
5. Diff3='(268.5-inf)' Diff4='(237.5-inf)' 68 ==> Diff2='(340.5-inf)' 68 conf:(1)
6. Diff3='(0.5-268.5]' Diff4='(-inf-1]' 65 ==> Sev4=extreme 65 conf:(1)
7. Diff3='(-inf-0.5]' 214 ==> Sev3=extreme 212 conf:(0.99)
8. Diff4='(237.5-inf)' 69 ==> Diff2='(340.5-inf)' Diff3='(268.5-inf)' 68 conf:(0.99)
9. Diff2='(340.5-inf)' Diff4='(237.5-inf)' 69 ==> Diff3='(268.5-inf)' 68 conf:(0.99)
10. Diff4='(237.5-inf)' 69 ==> Diff3='(268.5-inf)' 68 conf:(0.99)

PredictiveApriori – Best rules found:

1. Diff3='(-inf-0.5]' 214 ==> Sev3=extreme 212 acc:(0.99495)
2. Diff4='(-inf-1]' 83 ==> Sev4=extreme 83 acc:(0.99494)
3. Diff4='(237.5-inf)' 69 ==> Diff2='(340.5-inf)' 69 acc:(0.99489)
4. Diff4='(237.5-inf)' 69 ==> Diff2='(340.5-inf)' Diff3='(268.5-inf)' 68 acc:(0.99403)
5. Diff2='(-inf-118]' Diff3='(0.5-268.5]' 20 ==> Sev4=extreme 20 acc:(0.99337)
6. Sev3=moderate Sev4=moderate Diff2='(340.5-inf)' 16 ==> Diff3='(268.5-inf)' 16 acc:(0.99223)
7. Sev3=moderate Sev4=moderate Diff3='(268.5-inf)' 16 ==> Diff2='(340.5-inf)' 16 acc:(0.99223)
8. Sev4=significant Diff3='(0.5-268.5]' 15 ==> Diff4='(1-237.5]' 15 acc:(0.99176)
9. Sev2=slight Sev3=extreme Diff2='(118-340.5]' 15 ==> Diff3='(-inf-0.5]' 15 acc:(0.99176)
10. Sev2=moderate Sev4=moderate Diff2='(340.5-inf)' 15 ==> Diff3='(268.5-inf)' 15 acc:(0.99176)
11. Sev2=moderate Sev4=moderate Diff3='(268.5-inf)' 15 ==> Diff2='(340.5-inf)' 15 acc:(0.99176)
12. Sev2=significant Diff2='(118-340.5]' Diff4='(-inf-1]' 14 ==> Sev4=extreme Diff3='(0.5-268.5]' 14 acc:(0.99116)
13. Sev4=significant Diff2='(118-340.5]' 12 ==> Diff4='(1-237.5]' 12 acc:(0.98939)
14. Sev2=extreme Sev4=extreme Diff4='(237.5-inf)' 12 ==> Sev3=extreme Diff2='(340.5-inf)' 12 acc:(0.98939)
15. Sev2=extreme Sev4=extreme Diff4='(237.5-inf)' 12 ==> Sev3=extreme Diff3='(268.5-inf)' 12 acc:(0.98939)

16. Sev3=significant Diff2='(118-340.5]' Diff4='(-inf-1]' 12 ==> Sev4=extreme Diff3='(0.5-268.5]' 12 acc:(0.98939)
17. Sev4=moderate Diff2='(340.5-inf)' 32 ==> Diff3='(268.5-inf)' 31 acc:(0.98741)
18. Sev2=significant Diff3='(268.5-inf)' 30 ==> Diff2='(340.5-inf)' 29 acc:(0.98585)
19. Sev2=moderate Sev3=extreme Diff2='(-inf-118]' 24 ==> Diff3='(-inf-0.5]' 23 acc:(0.97697)
20. Sev3=extreme Sev4=extreme Diff3='(268.5-inf)' 24 ==> Diff2='(340.5-inf)' 23 acc:(0.97697)
21. Sev2=significant Sev3=extreme Diff2='(-inf-118]' 40 ==> Diff3='(-inf-0.5]' 38 acc:(0.96093)
22. Sev2=extreme Diff2='(-inf-118]' 61 ==> Sev3=extreme 58 acc:(0.95323)
23. Sev2=extreme Diff3='(0.5-268.5]' 34 ==> Sev4=extreme 32 acc:(0.94227)
24. Sev3=moderate Diff3='(268.5-inf)' 33 ==> Diff2='(340.5-inf)' 31 acc:(0.93793)
25. Sev4=moderate Diff3='(268.5-inf)' 33 ==> Diff2='(340.5-inf)' 31 acc:(0.93793)
26. Diff3='(268.5-inf)' 108 ==> Diff2='(340.5-inf)' 100 acc:(0.9274)
27. Diff2='(-inf-118]' 143 ==> Sev3=extreme 132 acc:(0.9247)
28. Sev3=extreme Diff2='(-inf-118]' 132 ==> Diff3='(-inf-0.5]' 121 acc:(0.91709)
29. Sev2=significant Diff2='(-inf-118]' 43 ==> Diff3='(-inf-0.5]' 40 acc:(0.9155)
30. Sev2=extreme 133 ==> Sev3=extreme 120 acc:(0.89766)
31. Sev2=none 2 ==> Diff2='(118-340.5]' 2 acc:(0.85914)

7.10 Summary of Results

The initial concept of determining TTF for VibrationStandardEquipmentID, VibrationStandardEquipmentGroupID, VibrationStandardDiagnosisGroupID and VibrationStandardDiagnosisID and using continuous, numerical values for the Severity, DaysToFailure, and TTF were much too specific for the data available. Through extensive data analysis, it is apparent there are not enough cases when analysis was performed on VibrationStandardEquipmentID, VibrationStandardEquipmentGroupID, VibrationStandardDiagnosisID, and VibrationStandardDiagnosisGroupID. There are too few cases specific to each of these attributes. Even when attempting to generalize the VibrationStandardEquipmentID and VibrationStandardDiagnosisID into groups, the groups were still too specific. This was shown when the best results were generated from the dataset that only included the Sev# and Diff# discretized into equi-depth groups.

In order to gain accuracy, precision would be lost. When the dataset was discretized to six groups rather than three groups in an attempt to improve precision, the group range was much

smaller but the accuracy dropped significantly. With the data available, high precision and accuracy is not possible. With the addition of the progression of severity, there was no improvement in accuracy but it produced just slightly worse accuracy.

When applying the progression of severity, the results were quite similar but showed no improvements. The progression of severity also predicts well for Diff3 and Diff4 but not for Diff2.

Since the best results were produced by the Logistic regression algorithm, a system should use this type of classification. Multilayer Perceptron would be the second best algorithm of choice to develop. The system should not base an estimate on Diff2 because it produced less than 40% correctly classified instances. Diff3 was the best attribute to determine TTF.

Chapter 8
DISCUSSION

Greitzer and Ferryman studied the "investigation of a generalized statistical method for characterizing and predicting system degradation" [5]. This project predicted 'failure' using Figure of Merit (FOM), the system's quantified current degree of fault, progression of fault, and the level of fault that will produce a failure of the system. "The specification of these factors, which is necessary to perform diagnostics and prognostics, is typically done through engineering/analytical studies such as Failure Modes and Effects Analysis. These analyses and expert judgments yield descriptions of how the system fails, what faults can be measured given available sensors and the values expected for these sensors when these failures occur" [5]. The issue with the use of FOM is that some databases, including the database used in this project, does not monitor or store FOM.

Greitzer, Stahlman, Ferryman, Wilson, Kangas, and Sisk studied "health monitoring of complex mechanical systems for diagnostics, prognostics, and maintenance scheduling" [6]. The project, the Pacific Northwest National Laboratory's LEAP project, used linear regression to predict the time to failure. To improve this time to failure, they estimated the "candidate statistical methods include multivariate regression, Bayesian regression methods, time-series analysis, and discrimination or clustering analysis" [6]. The project focused on two different types of applications. One application used real-time data for gas turbine engines and another application used oil analysis data on large diesel electric engines. This system is quite limiting since it is applied to only these two application areas.

Chapter 9
FUTURE WORK

Begin by converting the numerical values, Sev# and Diff# to discretized nominal values. Use just the Sev# and Diff# values and apply the Logistic regression or Multilayer Perceptron algorithm in predicting TTF.

When evaluating the cases shown in Appendix C.3, the Machines with the same MID showed a broad range of TTF values. This may be due to a machine having some maintenance performed and postponing TTF. Incorporating repair history while determining cases can potentially improve the accuracy of the TTF values.

Another aspect to investigate is to have a broader acceptable range when the fault is first detected and narrowing the range as the fault severity increases and TTF approaches. It is not as important to know what day a machine will fail when the system is estimating a TTF that is still over six months in the future.

Chapter 10
EDUCATIONAL STATEMENT

This project drew upon the knowledge obtained from several courses in the Computing and Software Systems Master's Curriculum and my work experience. Work experience built the foundation to work with databases and understand database structure. This project had a practical application of efficiently accessing data. TCSS543 Advanced Algorithms built the foundation for the mathematical aspects of this project. Algorithms and statistical analysis were studied and researched to achieve the goal of this project. TCSS598 Master's Seminar greatly helped in writing, researching, and reading technical papers. TCSS560 Software Engineering was very helpful in designing the project. TCSS555 Data Mining and TCSS435 Artificial Intelligence stated the need to extract knowledge from databases and presented the methodologies for attaining new knowledge. The project allowed me to apply many concepts introduced throughout the Master's program.

Chapter 11
CONCLUSION

This paper has shown that determining TTF using CBR and regression with the initial generalization groupings and continuous values produced low accuracy rates. Good results were obtained by discretizing in small number of groups with equi-depths. Due to the low numbers of cases for each specific MID and Diagnoses, it is best to determine TTF based on Sev# and Diff# only. Many databases at DLI Engineering do not include repair history information but the database used in this project does include repair history information. Future work will determine if incorporating repair history can improve accuracy.

BIBLIOGRAPHY

[1] Chi, R.T.H. and Kiang, M.Y. (1991). An Integrated Approach of Rule-Based and Case-Based Reasoning for Decision Support. *ACM Annual Computer Science Conference* (pp255-267), San New York, NY: ACM Press.

[2] Data Mining. Retrieved April 9, 2005, from http://encyclopedia.laborlawtalk.com/Data_mining

[3] Davidson, G. (Oct 2003). Dilemma – to Call or Not to Call, *The Sound and Vibration* (10-11).

[4] Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. (Fall 1996). From Data Mining to Knowledge Discovery in Databases. *American Association for Artificial Intelligence*, 17(3), (37-54). Retrieved April 23, 2005 from

[5] Greitzer, F.L., and Ferryman, T.A. (April 2-3, 2001). Predicting Remaining Life of Mechanical Systems, *Intelligent Ship Symposium IV*.

[6] Greitzer, F.L., Stahlman, E.J., Ferryman, T.A., Wilson, B.W., Kangas, L.J., and Sisk, D.R. (1999, September 2). *Development of a Framework for Predicting Life of Mechanical Systems: Life Extension Analysis and Prognostics (LEAP)*. Paper presented at the International Society of Logistics (SOLE) 1999 Symposium.

[7] Hadden, G.D., Bergstrom, P., Samad, T., Bennett, B.H., Vachtsevanos, G.J., and Van Dyke, J. (2000). Application Challenges: System Health Management for Complex Systems. Retrieved on April, 13 2005, from http://www.adventiumlabs.org/Publications/haddenhpc2000.pdf

[8] Han, J. and Kamber, M. (2001). *Data Mining: Concepts and Techniques*. San Francisco, Morgan Kaufmann Publishers.

[9] Olsson, E., Funk, P. and Xiong, N. (2004). Fault Diagnosis in Industry Using Sensor Readings and Case-Based Reasoning, *Journal of Intelligent & Fuzzy Systems*, 15, (41-46).

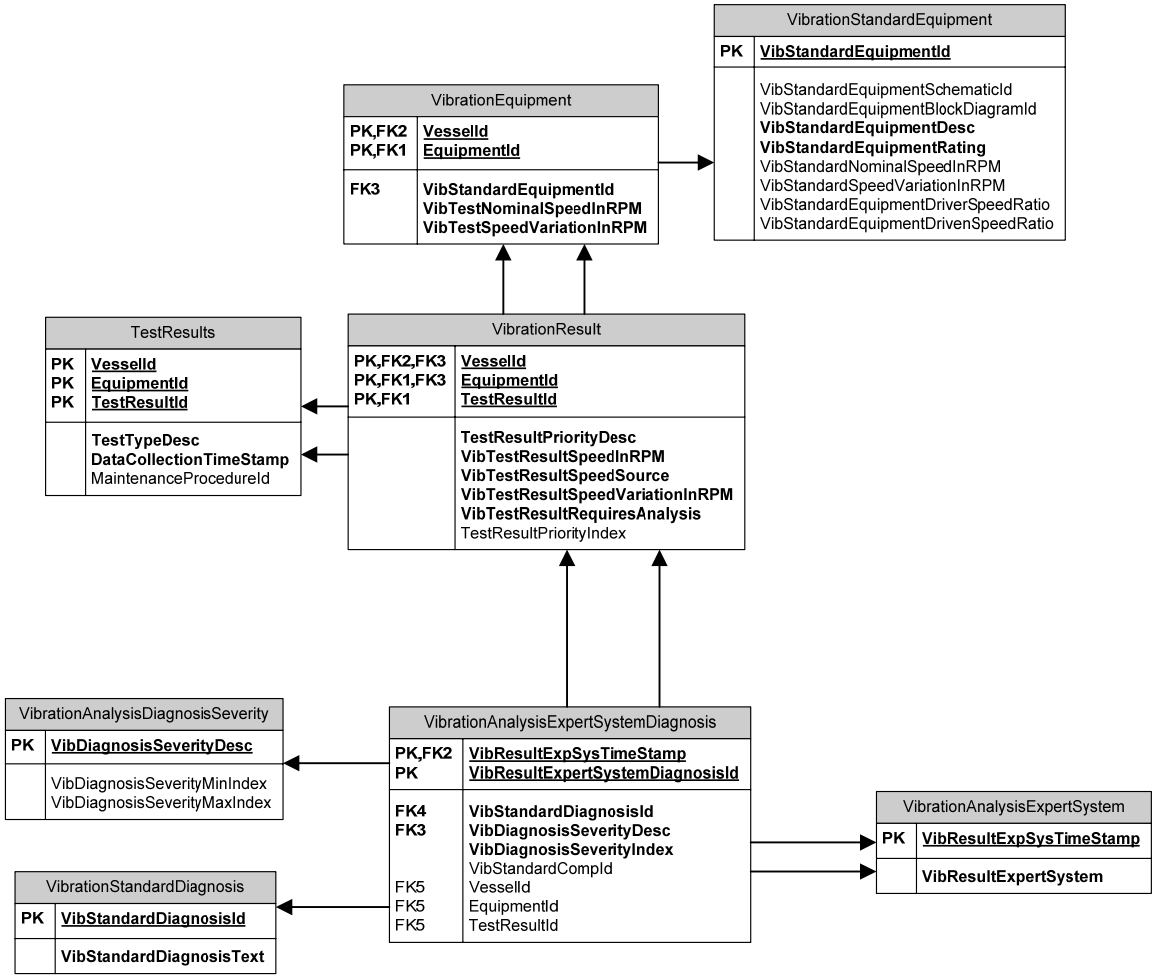
[10] Pal, S. K. and Shiu, S.C.K. (2004). *Foundations of Soft Case-Based Reasoning*. Hoboken, New Jersey: John Wiley & Sons, Inc.

[11] Scmidt, R., and Grierl, L. (2003). Predicting Influenza Waves with Health Insurance Data. In Permer, P., Brause, R., and Holzhűtter, H. (Eds.). *Medical Data Analysis: 4th International Symposium*, ISMDA 2003. (pp. 91-98).

[12] White, G. (1993). *Introduction to Machine Vibration*. Bainbridge Island, Washington: DLI Engineering.

Appendix A
DATABASE

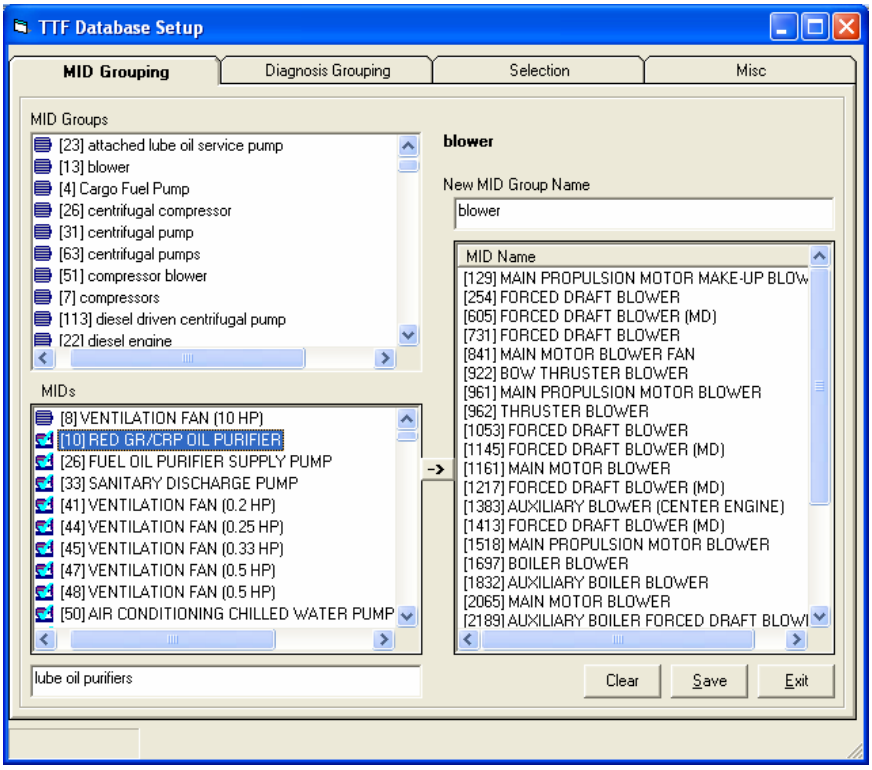
A.1 Partial Database Schema



Appendix B
PRE-PROCESSING TOOL

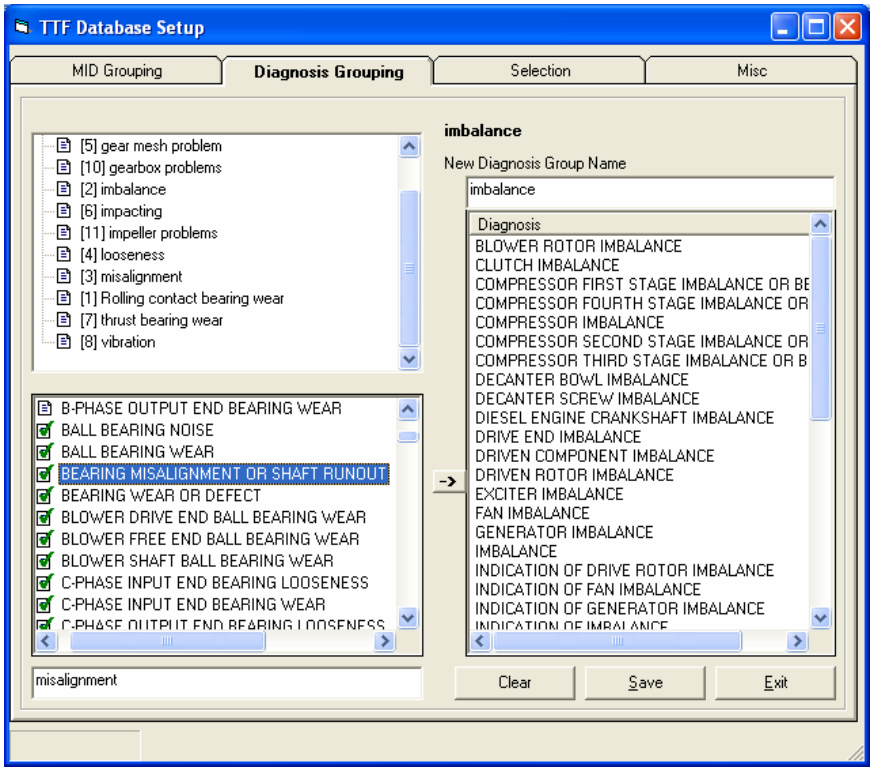
B.1 MID Grouping

MID Grouping tab is used to assist the user in grouping MIDs.



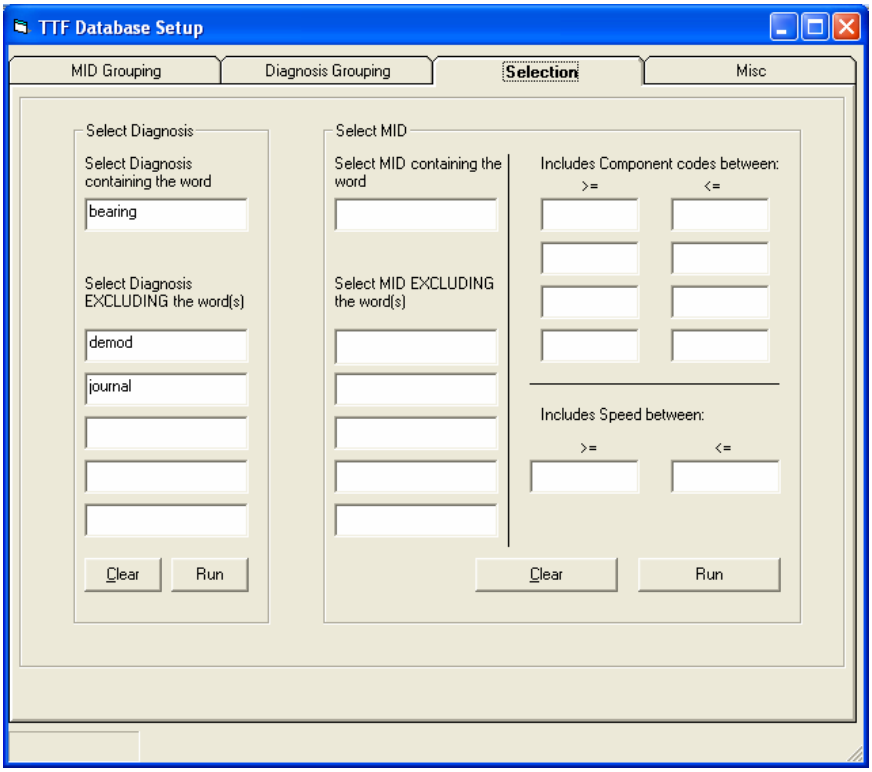
B.2 Diagnosis Grouping

Diagnosis Grouping tab is used to assist the user in grouping Diagnoses.


TTF Database Setup
MID Grouping
Diagnosis Grouping
Selection
Misc
[5] gear mesh problem
[10] gearbox problems
[2] imbalance
[6] impacting
[11] impeller problems
[4] looseness
[3] misalignment
[1] Rolling contact bearing wear
[7] thrust bearing wear
[8] vibration
B-PHASE OUTPUT END BEARING WEAR
BALL BEARING NOISE
BALL BEARING WEAR
BEARING MISALIGNMENT OR SHAFT RUNOUT
BEARING WEAR OR DEFECT
BLOWER DRIVE END BALL BEARING WEAR
BLOWER FREE END BALL BEARING WEAR
BLOWER SHAFT BALL BEARING WEAR
C-PHASE INPUT END BEARING LOOSENESS
C-PHASE INPUT END BEARING WEAR
misalignment
imbalance
New Diagnosis Group Name
imbalance
Diagnosis
BLOWER ROTOR IMBALANCE
CLUTCH IMBALANCE
COMPRESSOR FIRST STAGE IMBALANCE OR BE
COMPRESSOR FOURTH STAGE IMBALANCE OR
COMPRESSOR IMBALANCE
COMPRESSOR SECOND STAGE IMBALANCE OR
COMPRESSOR THIRD STAGE IMBALANCE OR B
DECANTER BOWL IMBALANCE
DECANTER SCREW IMBALANCE
DIESEL ENGINE CRANKSHAFT IMBALANCE
DRIVE END IMBALANCE
DRIVEN COMPONENT IMBALANCE
DRIVEN ROTOR IMBALANCE
EXCITER IMBALANCE
FAN IMBALANCE
GENERATOR IMBALANCE
IMBALANCE
INDICATION OF DRIVE ROTOR IMBALANCE
INDICATION OF FAN IMBALANCE
INDICATION OF GENERATOR IMBALANCE
->
Clear
Save
Exit

B.3 Selection

The Selection tab assists the user in selecting certain diagnoses or MIDs based on different criteria entered.



Appendix C

TTF SYSTEM

C.1 VibrationCase – raw data (partial)

Column names: CaseID, VibStandardEquipmentID, VibStandardDiagnosisID, DiagnosisGroupID, VibStandardEquipmentGroupID, TotalDaysToFailure, CaseType,CaseIsActive

10,483,238,,,560,0,1
14,491,459,,,1095,0,1
16,528,398,,,567,0,1
18,3063,481,,,263,0,1
24,882,397,,,70,0,1
26,817,484,,,609,0,1
28,389,219,,,263,0,1
30,743,397,,,219,0,1
32,845,605,,,185,0,1
356,528,,1,,1390,1,1
357,743,,1,,344,1,1
358,727,,1,,864,1,1
359,275,,10,,566,1,1
360,571,,1,,1518,1,1
361,1148,,1,,643,1,1
362,632,,1,,136,1,1
363,202,,1,,403,1,1
364,214,,1,,985,1,1
365,217,,1,,828,1,1
366,79,,10,,484,1,1
398,,17,,11,1725,2,1
399,,132,,13,159,2,1
400,,151,,18,811,2,1
401,,180,,21,506,2,1
402,,294,,22,715,2,1
403,,204,,31,449,2,1
404,,398,,35,314,2,1
405,,106,,37,1019,2,1
406,,475,,38,175,2,1
448,,,1,1,1018,3,1
449,,,1,9,1257,3,1
450,,,9,11,280,3,1
451,,,1,13,972,3,1
452,,,10,21,451,3,1
453,,,1,35,403,3,1
454,,,9,37,229,3,1
455,,,1,40,1016,3,1

C.2 VibrationCaseTest – raw data

Column Names: TestCaseID, VesselID, EquipmentID, TestResultID, CaseID, DayPosition, VibDiagnosisSeverityIndex, CaseTestIsActive

61,2,2613,10224,10,0,0,1
62,2,2613,10323,10,275,260,1
63,2,2613,10561,10,560,630,1
74,3,6636,12922,14,0,0,1
75,3,6636,12921,14,414,276,1
76,3,6636,13187,14,1095,714,1
84,3,7384,15635,16,0,0,1
85,3,7384,16485,16,212,312,1
86,3,7384,16752,16,281,333,1
87,3,7384,17284,16,567,617,1
2012,71,935,4367,399,0,0,1
2009,71,935,4843,399,33,100,1
2008,71,949,4844,399,35,45,1
2011,67,935,3243,399,60,85,1
2010,67,949,3166,399,69,45,1
2006,72,949,2745,399,70,28,1
2007,72,935,1125,399,114,38,1
2013,71,935,4428,399,159,995,1
2060,13,734,17693,402,0,0,1
2058,59,865,1728,402,155,144,1
2059,41,11165,15167,402,538,234,1
2061,13,734,18901,402,715,1050,1
2066,76,657,2380,403,0,0,1
2067,76,657,2477,403,81,760,1
2062,76,758,2484,403,256,450,1
2063,76,757,2483,403,276,443,1
2064,41,668,14882,403,405,293,1
2065,36,768,12860,403,449,257,1
2109,20,11918,14760,406,0,0,1
2103,74,11905,660,406,1,343,1
2106,50,11905,1485,406,100,40,1
2110,20,11918,15154,406,100,789,1
2104,52,11930,2165,406,121,77,1
2107,34,11905,17013,406,141,6,1
2105,52,11905,2502,406,164,174,1
2108,20,11905,18135,406,175,486,1
2132,37,970,12818,409,0,0,1
2133,37,970,12921,409,11,2194,1
2130,17,957,15727,409,77,1551,1
2129,17,970,15720,409,89,2191,1
2128,37,957,18637,409,233,311,1
2131,14,970,14254,409,265,231,1

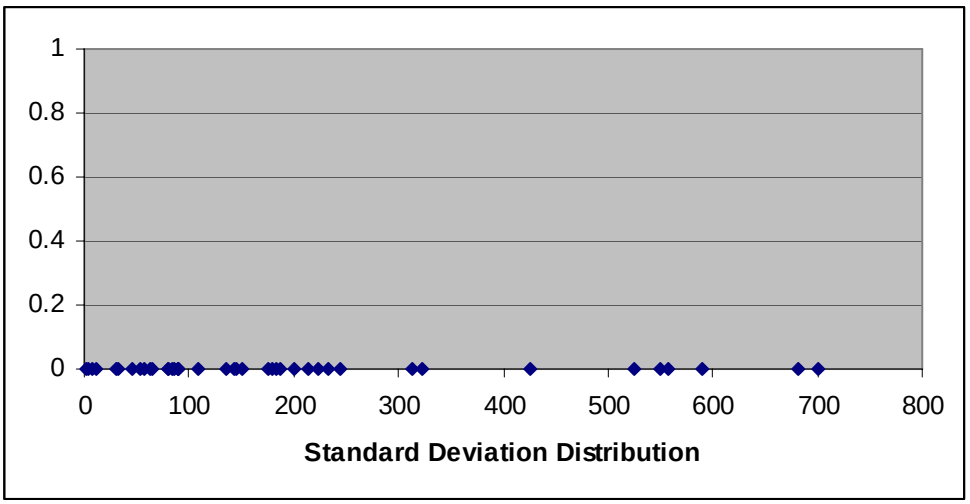
C.3 Case Evaluation

Case Type	MID	Dx	Case Count	TTF Values	Mean	Difference	SD
0	229	489	2	818, 820	819.00	2	1.00
0	548	398	2	217, 223	220.00	6	3.00
0	1239	395	2	178, 193	185.50	15	7.48
0	920	211	2	528, 552	540.00	24	12.00
0	229	461	2	191, 254	222.50	63	31.50
0	1685	459	2	128, 192	160.00	64	32.00
0	1192	395	2	276, 368	322.00	92	46.00
0	1886	363	2	1065, 753	909.00	312	156.00
0	637	486	2	311, 427	369.00	116	58.00
0	621	598	2	154, 281	217.50	127	63.50
0	886	395	2	341, 470	405.50	129	64.50
0	65	508	3	674, 732, 867	757.67	135	80.86
0	365	238	2	504, 666	585.00	162	81.00
1	79	10	2	484, 653	568.50	169	84.50
0	1173	36	2	247, 419	333.00	172	86.00
0	1314	459	3	372, 553	497.67	181	89.07
0	202	34	2	110, 289	199.50	179	89.50
0	1596	198	5	151, 92, 74, 316, 227	172.00	242	89.67
0	204	317	2	174, 393	283.50	219	109.50
0	222	54	2	64, 98	366.00	272	136.00
0	1939	113	2	131, 418	274.50	287	143.50
0	59	398	2	333, 624	478.50	291	145.50
0	91	398	3	626, 939, 956	840.33	330	151.72
0	2091	198	6	359, 259, 642, 254, 624, 218	392.67	424	175.32
0	1453	459	2	467, 827	647.00	360	180.00
0	1182	481	2	194, 559	376.50	365	182.50

0	1979	238	2	171, 547	359.00	376	188.00
0	1587	459	2	15, 710, 386, 788	587.00	402	201.00
0	2091	206	5	222, 213, 764, 439, 219	371.40	551	214.17
0	1310	459	2	567, 1012	789.50	445	222.50
0	204	395	2	375, 841	608.00	466	233.00
0	366	459	2	834, 1321	1077.50	862	243.50
0	483	238	4	460, 1059, 1170, 1422	1052.75	862	313.42
0	920	198	4	148, 267, 930, 157	375.50	782	323.55
1	2091	10	2	642, 1494	1068.00	852	426.00
0	1311	459	2	272, 1323	797.50	1051	525.50
0	1155	398	7	173, 181, 369, 389, 612, 1262, 1723	672.71	1550	550.10
0	186	395	2	206, 1322	764.00	1116	558.00
0	169	198	9	166, 230, 253, 309, 500, 540, 718, 1748, 1752	690.67	1586	589.62
0	359	398	4	320, 414, 635, 2007	844.00	1687	681.13
0	79	282	4	357, 609, 670, 2142	944.50	1785	701.26

Note:
MID: VibStandardEquipmentID
Dx: VibrationStandardDiagnosisID
Case Count: Number of cases
TTF Values: TTF for each case
Mean: Mean TTF
Difference: Maximum TTF minus Minimum TTF
SD: Standard deviation

C.4 Case Evaluation Distribution



C.5 Training Results

Case	Training - Database 1			
	Three months	Percent	Two months	Percent
Type 0	280/280	100.0%	280/280	100.0%
Type 1	33/33	100.0%	33/33	100.0%
Type 2	37/39	94.87%	37/39	94.87%
Type 3	23/27	85.19%	22/27	81.48%
Total	373/379	98.42%	372/379	98.15%

C.6 Test knowing TTF

Case	Test – Database 2 New tests in db				Test (Database 1)			
	Three months	Percent	Two months	Percent	Three months	Percent	Two months	Percent
Type 0	23/49	49.94%	18/49	36.73%	105/227	46.26%	72/227	31.72%
Type 1	1/2	50.00%	0/2	0.00%	15/28	53.57%	11/28	39.29%
Type 2	4/11	36.36%	4/11	36.36%	8/16	50.00%	6/16	37.50%
Type 3	8/20	40.00%	7/20	35.00%	6/15	40.00%	6/15	40.00%
Total	36/82	43.90%	29/82	35.37%	134/286	46.85%	95/286	33.22%

C.7 Test knowing TTF-Case selection breakdown

Test (Database 1)							
Case Selected	3 months		2 months		Out of range		Total
	Count	Percent	Count	Percent	Count	Percent	
1	2	100.0%	1	50.00%	0	0.0%	2
2	10	43.48%	6	26.09%	13	56.52%	23
3	47	44.76%	30	28.57%	58	55.24%	105
4	47	58.75%	34	42.50%	33	41.25%	80
5	0	0.0%	0	0.0%	9	100.0%	9
6	0	0.0%	0	0.0%	13	100.0%	13
7	0	0.0%	0	0.0%	0	0.0%	0
8	9	75.00%	6	50.00%	3	25.00%	12
9	10	45.45%	9	40.90%	12	54.54%	22
10	10	50.00%	8	40.00%	10	50.00%	20
Total							286

C.8 Initial test

Case	Test (Database 1)			
	Three months	Percent	Two months	Percent
Type 0	49/207	23.67%	39/207	18.84%
Type 1	6/24	25.00%	5/24	20.83%
Type 2	6/16	37.50%	4/16	25.00%
Type 3	4/9	44.44%	4/9	44.44%
Total	65/256	25.39%	49/256	20.31%

C.9 Initial test – case selection breakdown

Test (Database 1)							
Case	3 months		2 months		Out of range		Total
Selected	Count	Percent	Count	Percent	Count	Percent	
1	0	0.0%	0	0.0%	1	100.0%	**1**
2	0	0.0%	2	11.76%	15	88.24%	**17**
3	21	22.11%	14	14.74%	74	77.89%	**95**
4	22	29.73%	17	22.98%	52	70.27%	**74**
5	0	0.0%	0	0.00%	9	100.0%	**9**
6	0	0.00%	0	0.00%	15	100.0%	**15**
7	0	0.00%	0	0.00%	0	0.00%	**0**
8	0	0.00%	3	50.00%	3	50.00%	**6**
9	8	38.10%	7	33.33%	13	61.90%	**21**
10	8	29.63%	6	27.27%	14	63.63%	**22**
Total							260

C.10 2-Nearest Neighbor

Case	Test (Database 1)			
	Three months	Percent	Two months	Percent
Type 0	51/229	22.27%	36/229	15.72%
Type 1	8/33	24.24%	6/33	18.18%
Type 2	7/18	38.89%	6/18	33.33%
Type 3	6/17	35.29%	6/17	35.29%
Total	72/297	24.24%	54/297	18.18%

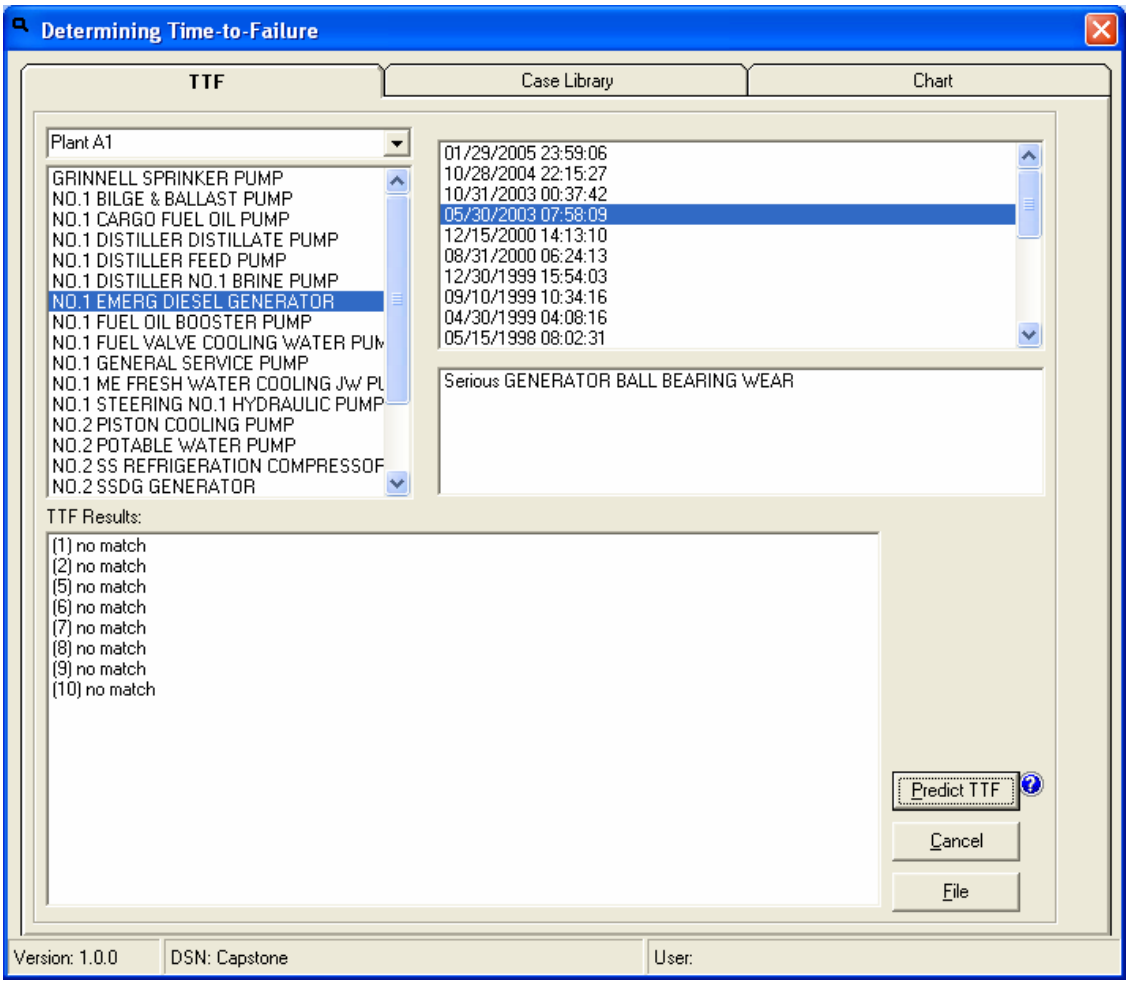
C.11 3-Nearest Neighbor

Case	Test (Database 1)			
	Three months	Percent	Two months	Percent
Type 0	27/233	11.59%	39/233	16.74%
Type 1	6/32	18.75%	5/32	15.63%
Type 2	6/18	33.33%	7/18	38.89%
Type 3	6/17	35.29%	6/17	35.29%
Total	45/300	15%	58/300	19.33%

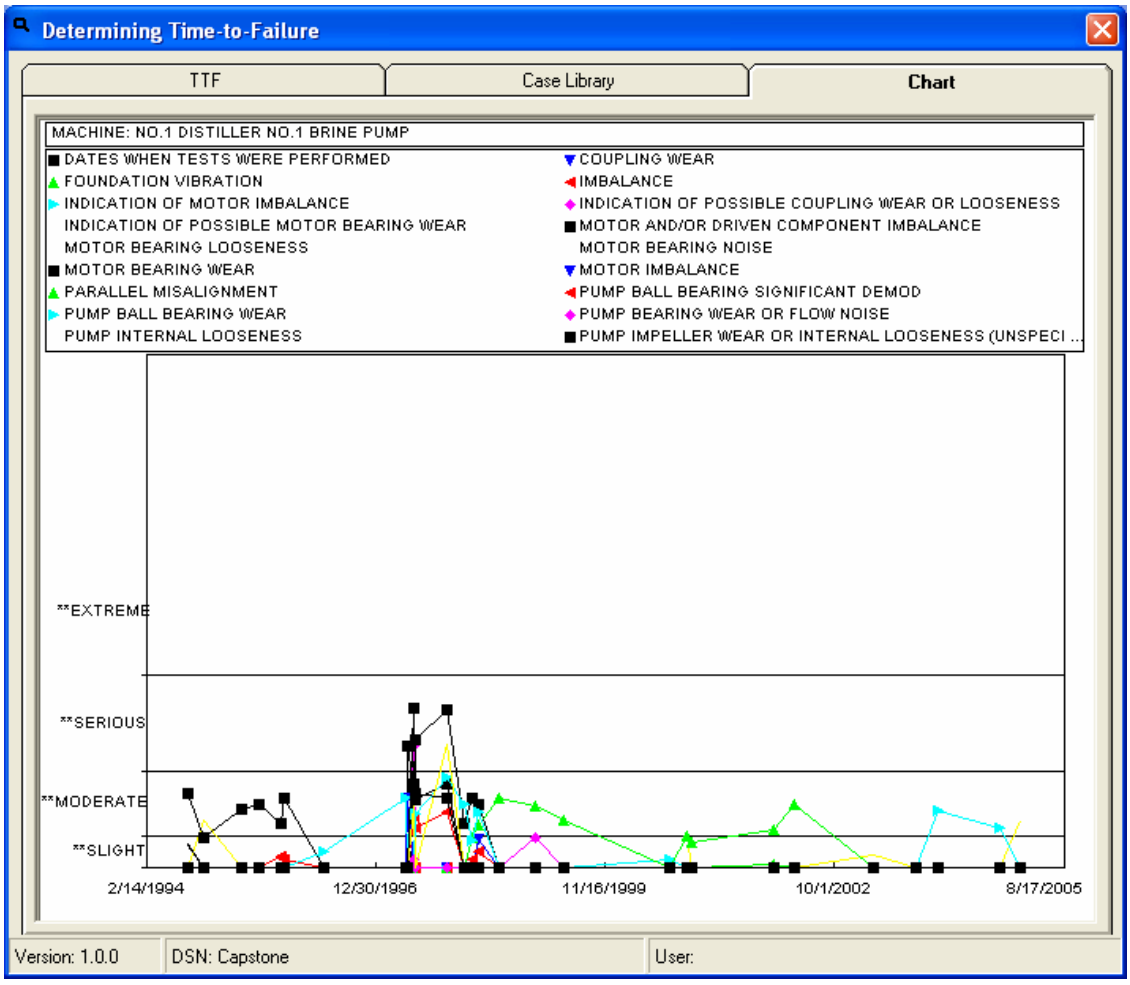
Appendix D

TTF USER INTERFACE

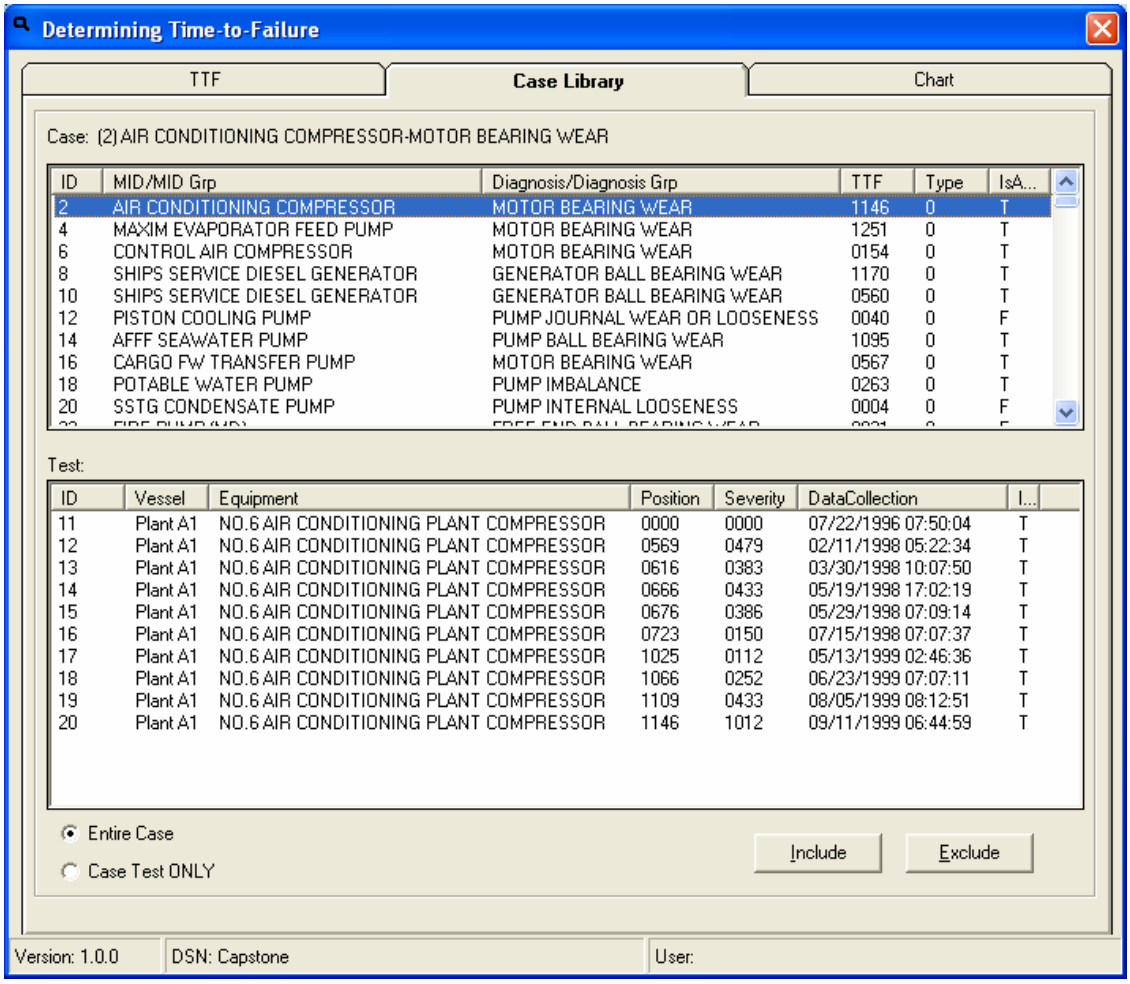
D.1 Determining TTF User Interface



D.2 Machine fault trend



D.3 Case Library

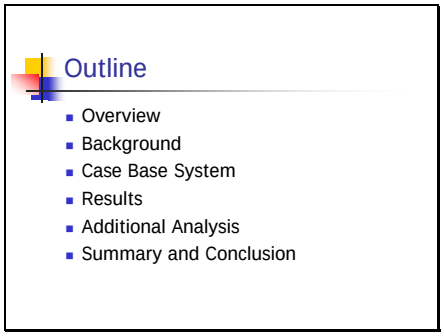


Appendix E

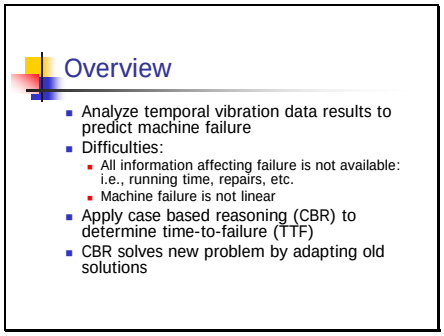
Power Point Presentation

Slide 1

Slide 2



Slide 3



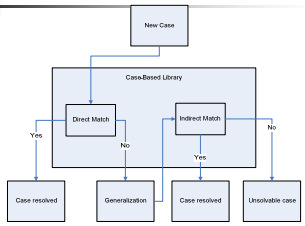
Slide 4

Overview - CBR

- The four parts of a CBR system
  - Retrieving – returns an old case that is identical or similar to the new problem
  - Reusing – applies the solution of the retrieved old case
  - Revising – adapts the retrieved solutions to resolve the new problem
  - Retaining – storing of valid cases

Slide 5

Overview



Slide 6

Background

- DLI Engineering
- Machine Condition Analysis software that analyzes vibration data for predictive maintenance
- The software generates:
  - Faults and a corresponding severity.
  - Recommendations and a corresponding priority, etc.

Slide 7

Background

- A FAULT is a description of a problem with the machine
- SEVERITY is the degree of the Fault
- A RECOMMENDATION is suggested repair action to resolve the Fault
- A PRIORITY is the degree of the Recommendation

Slide 8

Background

- Predictive Maintenance
  - Perform tests to determine what is starting to need a repair
  - Fixing the right problem rather than guessing what should be fixed
- Preventive Maintenance – repairs done on a schedule

Slide 9

Background – Expert Report

- **Main Service Pump #1**
- MID: 6
- Report generated on: 2/4/2005 12:28 PM
- Acquired: 7/27/2004 07:36 AM 1xM = 1781 RPM Averages: 4
- **Figure of Merit = 197.**
- **Maximum level: 111 (+14) VdB at 1.00x on 2A**
- **RECOMMENDATIONS:**
- **IMPORTANT: INSPECT COUPLING AND CHECK SHAFT ALIGNMENT**
- **DIAGNOSTICS:**
- **SERIOUS ANGULAR MISALIGNMENT**
- 111 (+14) VdB at 1.00xM on 2A in low range
- 107 (+11) VdB at 1.00xM on 2T in low range
- **SLIGHT PUMP FREE END BALL BEARING WEAR**
- 96 (+8.7) VdB at 22.6xP on 4A in high range
- 91 (+1.5) VdB at 19.8xP on 4A in high range
- 90 (+21) VdB at 5.88xP on 4A in low range

Slide 10

Background

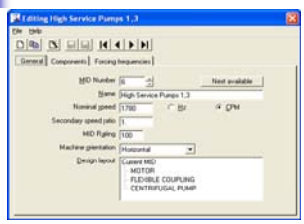
- Machine Identification (MID)
- Machinery configuration
- Nominal speed, orientation, fault frequencies, etc.

Slide 11

Background - Machine

Slide 12

Background - MID



Slide 13

Data Pre-processing

- Modify database to include TTF system
- Flag invalid data
  - Incorrectly collected data
  - Expert System reports: “Manual review of Spectra is warranted”, “Unavailable pickups at...” or “Questionable data at...”

Slide 14

Data Pre-processing

- Group MIDs
- Example group:
  - AC Chill Water Pump
  - A/C Chill Water Pump
  - Air Conditioning Chill Water Pump
  - Air Conditioning Chilled Water Pmp

Slide 15

Data Pre-processing

- Group Faults
- Example group:
  - Ball Bearing Noise
  - Ball Bearing Wear
  - Bearing Wear or Defect

Slide 16

Databases

- Two copies of the database were preprocessed.
- One copy, Database 1 contains tests up through 11/28/2005.
- A second copy of the database, Database 2 contains tests up through 09/13/2006.

Slide 17

Pre-Processing Results

	DB1	DB2	Diff	% of DB1 to DB2
Tests	142,324	169,713	27,389	83.86%
Valid Expert System runs	128,965 -90.61%	154,808 -91.21%	25,843	83.31%
Valid Expert System runs with Faults	69,228 (69228 / 142324 =48.64%)	83,207 (83207 / 169713= 49.03%)	13,979	83.20%
Total MIDs	1917	2080	163	92.16%
MID Groups	114	116	2	98.28%
MIDs in Group	1852 (1852 / 1917 = 96.6%)	2058 (2058 / 2080 = 98.9%)	206	90.00%
Total Diagnoses	656	659	3	99.54%
Diagnosis Grps	11	11	0	100%
Diagnoses in Grps	380 (380 / 656 = 57.9%)	380 (380 / 659 = 57.7%)	0	100%

Slide 18

Case Library Definition - Retaining

- Direct case – same fault, same machine (case type 0)
- Indirect case – grouped fault, same machine (case type 1)
- Indirect case – Normalize same fault within the grouped MID (case type 2)
- Indirect case – Normalize grouped fault within the grouped MID (case type 3)

Slide 19

Case Definition – Direct Case

- Case type 0
- Consists of a minimum of 3 consecutive tests on a machine with a specific fault:
  - Fault with an Extreme severity (considered machine failure)
  - Fault with any severity
  - Test where the fault does not exist.

Slide 20

Case Definition – Direct Case

Date	Severity	Diagnosis
01/14/04	NA	NA
04/07/04	Moderate	Ball Bearing Wear
09/03/04	Extreme	Ball Bearing Wear

Slide 21

Case Definition – Indirect case

- Case type 1
- Consists of a minimum of 3 consecutive tests on a machine with a group fault:
  - Group Fault with an Extreme severity
  - Group Fault with any severity
  - Test where the group fault does not exist.

Slide 22

Case Definition – Indirect case

Date	Severity	Diagnosis
01/14/04	NA	NA
04/07/04	Moderate	Ball Bearing Noise
09/03/04	Extreme	Ball Bearing Wear

Slide 23

Case Definition – Indirect case

- Case type 2
- Consists of a minimum of 3 consecutive tests on a MID group with a fault:
  - Fault with an Extreme severity
  - Fault with any severity
  - Test where the fault does not exist.

Slide 24

Case Definition – Indirect case

- Fault="Indication of Engine Overload or Injector Timing Problem"
- MID Group 22: Diesel Engine
- MID: 223 Main Propulsion Diesel Engine
- MID: 456 Auxiliary Propulsion Engine

Date	Day	Severity
01/14/04	0	NA
09/03/04	233	Extreme
01/22/06	0	NA
04/14/06	82	Moderate

Slide 25

Case Definition – Indirect case

- Case type 3
- Consists of a minimum of 3 consecutive tests on a MID group with a fault group:
  - Group Fault with an Extreme severity
  - Group Fault with any severity
  - Test where the group fault does not exist.

Slide 26

Case Library Retrieval

1. Direct match (case type 0) – same diagnosis, same MID, same machine
2. Indirect match (case type 0) – same diagnosis, same MID, different machine
3. Indirect match (case type 0) – same diagnosis, MID group
4. Indirect match (case type 0) – diagnosis group, MID group
5. Indirect match (case type 1) – grouped diagnosis , same machine

Slide 27

Case Library Retrieval

6. Indirect match (case type 1) – grouped diagnosis , same MID, different machine
7. Indirect match (case type 1) – grouped diagnosis, MID group, different machine
8. Indirect match case (case type 2)– same diagnosis within the grouped MIDs
9. Indirect match case (case type 2) – diagnosis within the diagnosis group in the grouped MIDs
10. Indirect match case (case type 3)– grouped diagnosis within the grouped MIDs

Slide 28

Revising Case Library

- Removal of entire case
- Removal of individual test in case

Slide 29

Case-Library Summary

Case	DB1 (training)	DB2	Difference (test)	% DB2 Cases
Case type 0	355	430	85	17.44%
Case type 1	39	40	1	2.50%
Case type 2	53	85	32	37.65%
Case type 3	39	61	22	36.07%
TOTAL	486	616	130	21.10%

Slide 30

Case-Library Analysis

Case Type	MID	Dx	Case Count	TTF Values	Mean	Difference	SD
0	229	489	2	818, 820	819.00	2	1.00
0	548	398	2	217, 223	220.00	6	3.00
0	1239	395	2	178, 193	185.50	15	7.48
0	920	211	2	528, 552	540.00	24	12.00
0	229	461	2	191, 254	222.50	63	31.50
0	1685	459	2	128, 192	160.00	64	32.00
0	1192	395	2	276, 368	322.00	92	46.00
0	1886	363	2	1065, 753	909.00	312	156.00
0	637	486	2	311, 427	369.00	116	58.00
0	621	598	2	154, 281	217.50	127	63.50
0	886	395	2	341, 470	405.50	129	64.50
0	65	508	3	674, 732, 867	757.67	135	80.86
0	365	238	2	504, 666	585.00	162	81.00

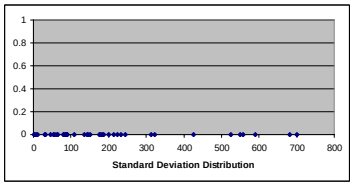
Slide 31

Case-Library Analysis – cont.

Case Type	MID	Dx	Case Count	TTF Values	Mean	Difference	SD
0	483	238	4	460, 1059, 1170, 1422	1052.75	862	313.42
0	920	198	4	148, 267, 930, 157	375.50	782	323.55
1	2091	10	2	642, 1494	1068.00	852	426.00
0	1311	459	2	272, 1323	797.50	1051	525.50
0	1155	398	7	173, 181, 369, 389, 612, 1262, 1723	672.71	1550	550.10
0	186	395	2	206, 1322	764.00	1116	558.00
0	169	198	9	166, 230, 253, 309, 500, 540, 710, 1748, 1752	690.67	1586	589.62
0	359	398	4	320, 414, 635, 2007	844.00	1687	681.13
0	79	282	4	357, 609, 670, 2142	944.50	1785	701.26

Slide 32

Case Evaluation Distribution



Slide 33

TTF – Criteria

- The TTF accuracy was based on:
  - two months (<=62 days)
  - three months (<=93 days)

Slide 34

CBR Results - training

Case	Training - Database 1			
	Three months	Percent	Two months	Percent
Type 0	280/280	100.0%	280/280	100.0%
Type 1	33/33	100.0%	33/33	100.0%
Type 2	37/39	94.87%	37/39	94.87%
Type 3	23/27	85.19%	22/27	81.48%
Total	373/379	98.42%	372/379	98.15%

Slide 35

CBR Results – Knowing TTF

Case	**Test – Database 2 New tests in db**				**Test (Database 1)**			
	Three months	**Percent**	**Two months**	**Percent**	**Three months**	**Percent**	**Two months**	**Percent**
Type 0	**23/49**	**49.94%**	**18/49**	**36.73%**	**105/227**	**46.26%**	**72/227**	**31.72%**
Type 1	**1/2**	**50.00%**	**0/2**	**0.00%**	**15/28**	**53.57%**	**11/28**	**39.29%**
Type 2	**4/11**	**36.36%**	**4/11**	**36.36%**	**8/16**	**50.00%**	**6/16**	**37.50%**
Type 3	**8/20**	**40.00%**	**7/20**	**35.00%**	**6/15**	**40.00%**	**6/15**	**40.00%**
Total	**36/82**	**43.90%**	**29/82**	**35.37%**	**134/286**	**46.85%**	**95/286**	**33.22%**

Slide 36

CBR Results – Initial Test

Case	Test (Database 1)			
	Three months	Percent	Two months	Percent
Type 0	49/207	23.67%	39/207	18.84%
Type 1	6/24	25.00%	5/24	20.83%
Type 2	6/16	37.50%	4/16	25.00%
Type 3	4/9	44.44%	4/9	44.44%
Total	65/256	25.39%	49/256	20.31%

Slide 37

CBR Results – 2 Nearest Neighbor

Case	Test (Database 1)			
	Three months	Percent	Two months	Percent
Type 0	51/229	22.27%	36/229	15.72%
Type 1	8/33	24.24%	6/33	18.18%
Type 2	7/18	38.89%	6/18	33.33%
Type 3	6/17	35.29%	6/17	35.29%
Total	72/297	24.24%	54/297	18.18%

Slide 38

CBR Results – 3 Nearest Neighbor

Case	Test (Database 1)			
	Three months	Percent	Two months	Percent
Type 0	27/233	11.59%	39/233	16.74%
Type 1	6/32	18.75%	5/32	15.63%
Type 2	6/18	33.33%	7/18	38.89%
Type 3	6/17	35.29%	6/17	35.29%
Total	45/300	15%	58/300	19.33%

Slide 39

SPSS

- Analytical software specializing in data mining
- www.spss.com

Slide 40

SPSS Linear Regression

- To evaluate the relationship of the independent variables:
  - VibrationStandardEquipmentGroupID
  - DiagnosisGroupID
  - VibrationStandardEquipmentID
  - VibrationStandardDiagnosisID
  - VibDiagnosisSeverityIndex
- To the dependant variable: DayPosition

Slide 41

SPSS Model Summary

- R values range from 0 to 1 with larger R values indicating stronger relationships.
- This model produced and R value of only .330.
- R squared values range from 0 to 1 with larger values indicating that the model fits the data well.
- This system only produced a value of .109 which shows that the model does not fit the data well.

Slide 42

SPSS Coefficients

Model	Unstandardized Coefficients		Standardized Coefficients	T	Sig.
	B	Std. Error	Beta		
(Constant)	133.508	46.836		2.851	.004
VibStandardEquipmentID	-.011	.015	-.019	-.709	.479
C.VibstandardDiagnosisID	-.015	.071	-.006	-.205	.837
DiagnosisGroupID	-8.004	3.636	-.063	-2.201	.028
VibStandardEquipmentGroupID	1.311	.319	.114	4.111	.000
VibDiagnosisSeverityIndex	.266	.023	.316	11.768	.000

Slide 43

SPSS Coefficients

- Using the coefficients in the prior slide, apply to equation in determining DayPosition:
- DayPosition = 133.508 + (-.011 * VibStandardEquipmentID) + (-.015 * VibStandardDiagnosisID) + (-8.004 * DiagnosisGroupID) + (1.311 * VibStandardEquipmentGroupID) + (.266 * VibDiagnosisSeverityIndex)

Slide 44

SPSS Coeffient - Results

Test – Database 1	Three months - Accuracy
Initial Test	25.39%
2-Nearest Neighbor	24.24%
3-Nearest Neighbor	15.00%
SPSS Regression	27.8%

Slide 45

Weighting Parameters

- Changing the weighting of the attributes to determine the importance of the attributes on the accuracy of the system.
- The calculation was done based on the following formula where Wi is the weight for the matching attribute and Wtotali is the weight of the attribute evaluated

$$\frac{\sum W_i F_i}{\sum Wtotal_i}$$

Slide 46

Weighting Dataset

- VibStandardSeverityIndex (converted to nominal values: None, Slight, Moderate, Serious, or Extreme)
- VibStandardEquipmentID
- VibStandardEquipmentGroupID
- VibStandardDiagnosisID
- DiagnosisGroupID
- DayPosition (converted to nominal values: <1month, 1-2months, 2-3months, 3-4months, 4-5months, 5-6months, 6-7months, 8-9months, 10-11months, 11-12months, and >12months)

Slide 47

Abbreviation used

- Num = Weighted test number
- Severity = VibStandardSeverityIndex
- MID = VibStandardEquipmentID
- MIDGrp = VibStandardEquipmentGroupID
- Dx = VibStandardDiagnosisID
- DxGrp = DiagnosisGroupID

Slide 48

Weighting Result Summary

Num	Severity	DayPosition	MID	MIDGrp	Dx	DxGrp	Accuracy
1	1	1	1	1	1	1	25.73%
2	1	1	0	0	0	0	11.92%
3	1	1	1	0	1	0	24.89%
4	0	0	0	1	0	1	23.64%
5	1	1	1	1	1	0	25.10%
6	1	1	1	0	1	1	26.57%
7	0	0	1	1	1	1	20.50%
8	0	0	1	0	1	0	20.08%
9	1	1	1	0	0	0	17.99%

Slide 49

WEKA

- Collection of machine learning algorithms for data mining tasks.
- www.cs.waikato.ac.nz/ml/weka/
- Open source under GPL
- Written in Java

Slide 50

Weka Algorithms

- J48 tree, ID3 tree, Multilayer Perceptron, Logistic regression, Apriori, Predictive Apriori, and K*
- J48 - generates a pruned or un-pruned C4 tree where values may be missing, attributes may be numeric, and can deal with noisy data.
- ID3 generates an un-pruned decision tree. Attributes must be nominal and there cannot be any missing values. Empty leaves may result in unclassified instances.
- Multilayer Perceptron is a neural network that uses back-propagation to train.

Slide 51

Weka Algorithms

- Logistic regression is used for building and using a multinomial logistic regression model with a ridge estimator.
- Apriori generates association rules from frequent item-sets
- Predictive Apriori finds association rules sorted by predictive accuracy.
- K* is an instance-based classifier, that is the class of a test instance is based upon the class of those training instances similar to it, as determined by some similarity function.

Slide 52

WEKA – Test Options

- Training set - provides optimal classification accuracy
- 10-fold cross validation
  - averages the classification results on ten different random samples generated from the dataset.
  - It provides more robust results when there is only one dataset available.
  - Tries to diversify the samples in order to get a better estimate with fewer samples.
  - It uses averages so the method is not as accurate.
- Ratio validation of 50%, 66%, 75%, and 80% training – uses a different percentage of records for the training and test set

Slide 53

WEKA – Dataset

- Modified so each row contains a case: each test severity and DayPosition
- Sev2 represents the severity value of the 2nd test, Sev3 represents the severity value of test 3, etc.
- Diff2 is the number of days to TTF calculated from the actual TTF minus the current day position of the test, and so forth.
- The dataset includes DiagnosisGroupID (DxGrpID), VibStandardEquipmentGroupID (MIDGrpID), Sev2, Sev3, Sev4, Diff2, Diff3, Diff4.

Slide 54

Discretized to 3 groups

- Discretized into three equi-depth groups
- Each group has same number of cases
- Produced large ranges
- The range for Diff3 in Group2 is 268 days or almost nine months.

	Diff2	Diff3	Diff4
Group 1	-inf-118	-inf-.5	-inf-1
Group 2	118-340.5	0.5-268.5	1-237.5
Group 3	340.5-inf	268.5-inf	237.5-inf

Slide 55

Sample Dataset

Dx Grp ID	MID Grp ID	Sev2	Sev3	Sev4	Diff2	Diff3	Diff4
1	127	moderate	extreme	?	118-340.5	-inf-0.5	?
1	127	slight	extreme	moderate	340.5-inf	268.5-inf	237.5-inf
1	127	moderate	moderate	extreme	340.5-inf	268.5-inf	-inf-1
1	96	serious	serious	serious	340.5-inf	268.5-inf	237.5-inf

Slide 56

3 Groups - Summary

10-fold Cross Validation	Diff4		Diff3		Diff2	
Algorithm	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances	Total # of Instances	Correctly Classified Instances
3 Grps-J48	221	**72.8507 %**	430	**79.3023 %**	430	31.1628 %
3 Grps -Multilayer Perceptron	221	65.6109%	430	74.186 %	430	**40 %**
3 Grps-Logistic	221	62.4434 %	430	70.9302 %	430	37.4419 %
3 Grps-K*	221	65.1584 %	430	73.7209 %	430	37.4419 %
Sev/Diff -J48	221	73.3032 %	430	79.0698 %	430	36.0465 %
Sev/Diff -Multilayer Perceptron	221	70.1357%	430	78.8372 %	430	38.8372 %
Sev/Diff -.**Multilayer Perceptron	221	75.5656 %	430	79.5349 %	430	**39.0698 %**
Sev/Diff -Logistic	221	**76.0181 %**	430	**80 %**	430	36.0465 %
Sev/Diff -K*	221	71.4932 %	430	79.0698 %	430	36.0465 %
Sev/Diff -ID3	NA	NA	430	76.9767 %	430	36.0465 %

**Multilayer Perceptron – modified options

Slide 57

Discretized to 6 groups

- Discretized into 6 groups
- Produced smaller ranges but still large

	Diff2	Diff3	Diff4
Group 1	**-inf-72.5**	**-inf-.5**	**-inf-1**
Group 2	**72.5-119.5**	**0.5-93.5**	**1-93.5**
Group 3	**119.5-194.5**	**93.5-186**	**93.5-196.5**
Group 4	**194.5-340.5**	**186-337**	**196.5-321**
Group 5	**340.5-572**	**337-642.5**	**321-689**
Group 6	**572-inf**	**642.5-inf**	**689-inf**

Slide 58

6 Groups - Results

J48 tree	Diff4		
	Total Number of Instances	Ignored Class Instances	Correctly Classified Instances
training data	**221**	**209**	**66.9683 %**
10-fold	**221**	**209**	**59.276 %**
Split 50% train	**107**	**108**	**50.4673 %**
Split 66% train	**77**	**70**	**49.3506 %**
Split 80% train	**41**	**45**	**60.9756 %**

Slide 59

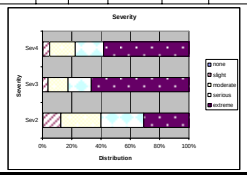
Progression of Failure

- Does the progression of severity assists in determining TTF?
- Another attribute was added to demonstrate a decrease, increase or stable severity.
- Sev_1-2 is the severity change between Sev1, Sev2, Sev_2-3 is the severity change between Sev2 and Sev3, and so forth.

Slide 60

Severity Distribution

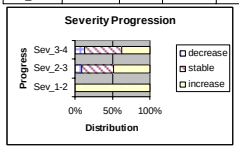
	none	slight	moderate	serious	extreme	total
Sev2	2	52	118	125	133	430
Sev3	0	16	59	68	287	430
Sev4	1	10	39	42	129	221



Slide 61

Severity Progression

	decrease	Stable	increase	total
Sev_1-2	0	2	428	430
Sev_2-3	39	184	207	430
Sev_3-4	28	110	83	221



Slide 62

Severity Progression-Summary

10-fold Cross Validation	Diff4 Total # of Instances	Diff4 Correctly Classified Instances	Diff3 Total # of Instances	Diff3 Correctly Classified Instances	Diff2 Total # of Instances	Diff2 Correctly Classified Instances
J48	221	**74.6606 %**	430	79.0698 %	430	36.0465 %
**Multilayer Perceptron	221	71.4932 %	430	**79.7674 %**	430	**39.0698 %**
Logistic	221	71.9457 %	430	**79.7674 %**	430	36.7442 %
K*	221	68.7783 %	430	78.6047 %	430	36.0465 %

**Multilayer Perceptron – modified options

Slide 63

WEKA Associations

- Apriori and Predictive Apriori
- The dataset includes the following attributes: Sev2, Sev3, Sev4, Diff2, Diff3, and Diff4.

Slide 64

Apriori

- 1. Diff4='(-inf-1]' 83 ==> Sev4=extreme 83 conf:(1)
- 2. Diff2='(118-340.5]' Diff3='(-inf-0.5]' 75 ==> Sev3=extreme 75 conf:(1)
- 3. Sev2=extreme Diff3='(-inf-0.5]' 73 ==> Sev3=extreme 73 conf:(1)
- 4. Diff4='(237.5-inf)' 69 ==> Diff2='(340.5-inf)' 69 conf:(1)
- 5. Diff3='(268.5-inf)' Diff4='(237.5-inf)' 68 ==> Diff2='(340.5-inf)' 68 conf:(1)

Slide 65

Apriori – cont.

- 6. Diff3='(0.5-268.5]' Diff4='(-inf-1]' 65 ==> Sev4=extreme 65 conf:(1)
- 7. Diff3='(-inf-0.5]' 214 ==> Sev3=extreme 212 conf:(0.99)
- 8. Diff4='(237.5-inf)' 69 ==> Diff2='(340.5-inf)' Diff3='(268.5-inf)' 68 conf:(0.99)
- 9. Diff2='(340.5-inf)' Diff4='(237.5-inf)' 69 ==> Diff3='(268.5-inf)' 68 conf:(0.99)
- 10. Diff4='(237.5-inf)' 69 ==> Diff3='(268.5-inf)' 68 conf:(0.99)

Slide 66

Predictive Apriori

- 1. Diff3='(-inf-0.5]' 214 ==> Sev3=extreme 212 acc:(0.99495)
- 2. Diff4='(-inf-1]' 83 ==> Sev4=extreme 83 acc:(0.99494)
- 3. Diff4='(237.5-inf)' 69 ==> Diff2='(340.5-inf)' 69 acc:(0.99489)
- 4. Diff4='(237.5-inf)' 69 ==> Diff2='(340.5-inf)' Diff3='(268.5-inf)' 68 acc:(0.99403)
- 5. Diff2='(-inf-118]' Diff3='(0.5-268.5]' 20 ==> Sev4=extreme 20 acc:(0.99337)

Slide 67

Predictive Apriori – cont.

- 6. Sev3=moderate Sev4=moderate Diff2='(340.5-inf)' 16 ==> Diff3='(268.5-inf)' 16 acc:(0.99223)
- 7. Sev3=moderate Sev4=moderate Diff3='(268.5-inf)' 16 ==> Diff2='(340.5-inf)' 16 acc:(0.99223)
- 8. Sev4=significant Diff3='(0.5-268.5]' 15 ==> Diff4='(1-237.5]' 15 acc:(0.99176)
- 9. Sev2=slight Sev3=extreme Diff2='(118-340.5]' 15 ==> Diff3='(-inf-0.5]' 15 acc:(0.99176)
- 10. Sev2=moderate Sev4=moderate Diff2='(340.5-inf)' 15 ==> Diff3='(268.5-inf)' 15 acc:(0.99176)

Slide 68

Summary of Results

- Poor results from continuous numeric values.
- Discretize into three equi-depth groups
- Best results (80% accuracy) from Logistic regression and secondly from Multilayer Perceptron
- Do not determine TTF based on Diff2
- Progression of severity comparably good results but no improvements.

Slide 69

Future Work

- Use discretized attributes Sev# and Diff# only
- Develop a system using the Logistic regression or Multilayer Perceptron algorithm
- See if repair history can eliminate those long TTF cases

Slide 70

Conclusion

- Determining TTF using CBR with the generalization groupings and continuous data produced low accuracy rates.
- Due to the low numbers of cases for each specific MID and Diagnoses, it is best to determine TTF based on Sev# and Diff# only.
- Include repair history information in an attempt to improve accuracy.
- Equi-depth groups with Logistic regression or Multilayer Perceptron algorithms produced good results.