Intelligent Knowledge Extraction from Service Notes for Equipment Diagnostics Support

Anil Varma and Vipin Ramani
Information Technology Laboratory
One Research Circle, Niskayuna, NY 12309. USA.
{varma, ramani}@crd.ge.com

Abstract

A valuable source of field diagnostic information for equipment service resides in the text notes generated during service calls. Intelligent knowledge extraction from such textual information is a challenging task. The notes are typically characterized by misspelled words, incomplete information, cryptic technical terms, and non-standard abbreviations. In addition, very few of the total number of notes generated may be diagnostically useful. A tool for identifying diagnostically relevant notes from the many raw field service notes and information is presented in this paper. N-gram matching and supervised learning techniques are used to generate recommendations for the diagnostic significance of incoming service notes. After some preprocessing and cleaning of the text, these diagnostic notes are indexed and made available in a laptop based tool to provide relevant retrieval in response to user queries and to help solve the current service call.

Introduction

Tools that embody domain knowledge and deploy it to offer expert level assistance to field engineers performing diagnostic tasks can provide a significant quality leverage to a service oriented organization. In the past couple of decades, many machine learning approaches like neural networks, decision trees etc. have been developed to learn over input data to provide prediction, classification and function approximation capabilities in the context of diagnosis. Often, such approaches have required structured and relatively static and complete input data sets for learning and produce models that resist real-world interpretation. An alternate approach, that of Case Based Reasoning is based on the observation that *experiential knowledge* (memory of past experiences - or cases) is an equally important component of human problem solving as learning rules or behaviors. Case Based Reasoning relies on relatively little pre-processing of raw knowledge, focusing instead on indexing, retrieval, reuse and archival of cases. In the diagnostics context, a case refers to a problem/solution description pair that represents an accurate diagnosis of the problem and the appropriate fix. The problem addressed in this paper is the separation of diagnostically useful and useless free-text machine repair records for structuring as cases that can be reused.

The approach underlying this project is generic in nature and applicable across diverse businesses - specially in the service field. The challenge of extracting diagnostic case information from free text typically occurs in any distributed repair and service oriented operation where a repair summary is entered as free text by the service personnel. Corporate retention and utilization of such information is also valuable since it represents mature, field validated knowledge about the failure modes of the equipment under consideration beyond troubleshooting manuals and can provide valuable inputs into the design and prioritization of future enhancements and upgrades

Background and Past Work

A variety of hybrid approaches have been used in the process of dealing with case based diagnostic system development. Uthurusamy et al. (1993) focus on improving the retrievability of free text service notes from General Motor's Technical Assistance Database. These researchers focus on natural-language processing techniques to correct, standardize and grammatically and semantically improve the source text. They then utilize a natural language parser to construct constraint rules from the input text. This approach is focused on supporting centralized technical assistance centers that respond to service calls. Creecy et al. (1992), present the PACE system which classifies free-text responses to US Census Bureau forms into hundreds of pre-determined categories and uses a knowledge base of over 100000 manually classified returns as the reasoning base. It uses the nearest neighbor technique with feature weighing. Liu et. al. (1995) apply case-based diagnostics using fuzzy neural networks (FNN) to the task to recalling help-desk call logs. To allow for the FNN to work, the call logs need to be mapped from an unstructured to a structured format with defined attributes and fuzzy if-then rules using linguistic properties are associated with each attribute. This approach uses pre-processing of incoming data to produce a well defined case structure. Varma et al. (1996) post-process responses to a user query to a structured case base to create flexible meta-information that allows the user to refine the case attribute definition. Both the approaches provide some flexibility in interacting with the data but only by mapping the data to distinct attributes

in either a pre or post processing stage.

The approach presented in this paper was developed to accommodate a business situation where the volume of service notes generated was extremely high with not all service dispatches necessarily containing diagnostic information. The diagnostic tool containing service notes was required to be deployed on field service engineers laptops with limited memory and processor speed. Thus there was a need to weed through the central database of service notes and select a small percentage with relevant diagnostic information. An additional task was to clean the quality of the text and provide a pick list of keywords that would provide starting points for accessing relevant service notes. This information was incorporated into an Microsoft Access based tool that was released to the field service engineers with periodic updates provided as new information was analyzed.

A hybrid approach is taken in this paper wherein supervised learning is used to try to predict whether a service note is diagnostically significant. This process is shown in Figure 1. A user is required to accept or reject a few service notes as being diagnostically relevant. Once a training set of accepted and rejected service notes is created, it is used to score new incoming service notes automatically as diagnostically relevant or irrelevant.

Once the subset of diagnostically useful problem/solution pairs is extracted, it is parsed into individual keywords that are run through a variety of filters. This process is shown in Figure 2. The purpose of this is two-fold. First, it is used to create a pick-list of keywords that users are assured will get them 'hits' in response to their query. Information learned during the first phase is used to assign significance values to terms that will appear in the keyword pick list. In addition, spelling correction and text reformulation is used to make the service notes easier to read and retrieve.

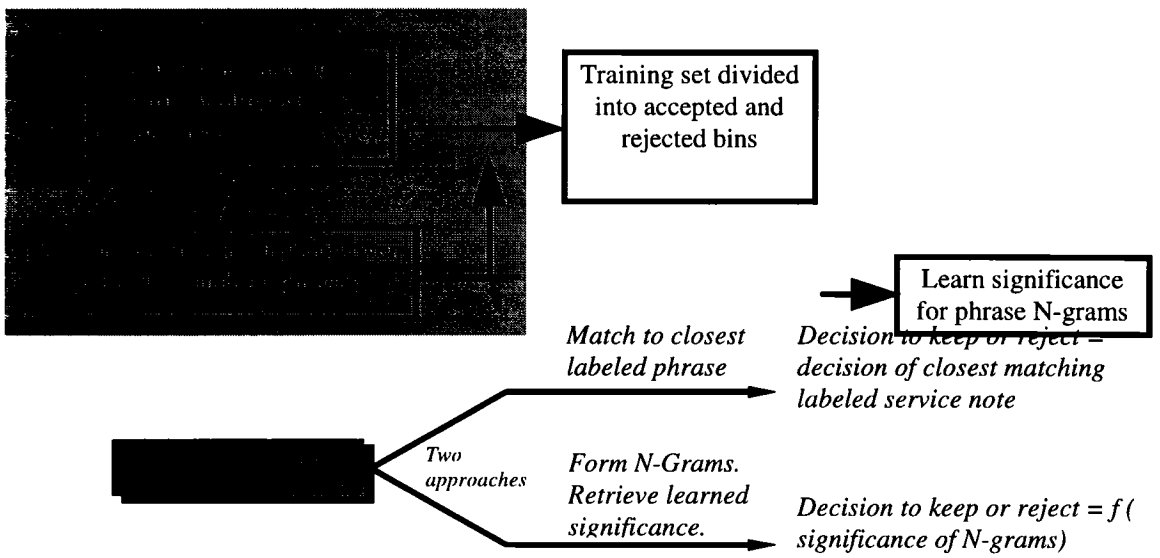

Figure 1. Training over labeled service notes and classifying new ones.

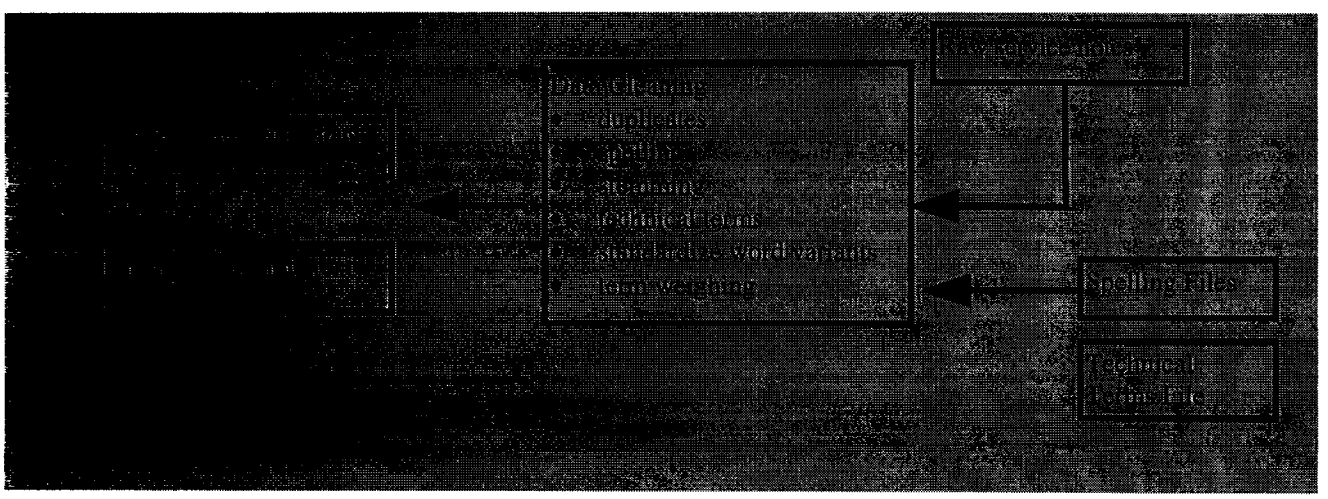

Figure 2. The Data Cleaning Process

Learning algorithm for diagnostic case extraction

The learning algorithm uses supervised learning to assign "significance factors" to phrase fragments that are subsets of the text constituting the symptom and solution fields in the dispatches. A domain expert is presented with a symptom/solution pair and asked to judge simply whether this should be included in the final set of cases that constitute the data in the tool.

A basic technique used in this paper is N-Gram matching. A N-gram refers to a fragment of N consecutive letters of a text phrase. For a given text phrase of length L, there are (L-N +1) N-grams, the first having as its first letter the first letter of the phrase, the second the second letter and so on. For example, the 3-grams (or trigrams) for the word diagnose would be {dia, iag,agn,gno,nos,ose}.

For each phrase that is labeled by the user in the training mode, all distinct N-grams are generated. A master list of distinct N-grams encountered so far is maintained along with two counts associated with each N-gram. The first count indicates how often this N-gram occurred in a phrase that was labeled as accept by the user and the second count indicated the number of times a phrase containing this N-gram was rejected.

N-grams are utilized to generate a similarity score between two text phrases. We refer to a parent phrase as one that has been assigned a label by the user and a candidate phrase is one that whose degree of similarity needs to be assessed with respect to the parent phrase. In its simplest version, this looks like

Simple n-gram similarity index =

$$\frac{[\,\#P_i \cap \#C_i\,]^2}{\#P_i \times \#C_i}$$

where

$\#P_i$ = Number of unique n-grams in Parent phrase
$\#C_i$ = Number unique n-grams in Candidate phrase

The system operates in a dual mode - incremental and batch. In the batch mode, every time a user labels a phrase, It uses simple n-gram matching to pull up symptom phrases that are extremely similar and assign to them, the same decision given to the parent phrase. This serves to present the minimum number of phrases for judgment to the user by ensuring that they are sufficiently distinct from phrases he/she has already encountered. To enforce this, the level of similarity needed for this to happen is assigned a relatively high value. This constitutes the batch mode of operation for the learning cycle. An typical example would be

```
1. VOID - DUPETCATE DISPATCH (User labeled
reject)
2. DUPLICATE DISPATCH VOIDED (automatically
rejected by system in batch mode)
```

Fig. 3 Example of service notes rejected by system.

Simultaneously, at every decision point, all the phrases that have had a decision associated with them are polled. They are split into all distinct n-grams and a count is maintained for each as to provide a relative frequency of occurrence in acceptable and rejected symptom/solution texts. By default, at the beginning of the process, all n-grams are assigned a weight of 1. A user-defined threshold determines how many occurrences of a particular N-gram in a phrase should be seen before its weights are sought to be adapted.

It is desirable that if N-grams are to be used to assess similarity between phrases, only those that provide distinguishing information regarding an accept or reject decision be utilized. For example, the word "the" is seldom indicative of whether the service note is useful whereas the word 'void' in the data set strongly indicates that the service note should be rejected. For this purpose, a "significance" was calculated for each N-gram. This was obtained by examining if the occurrence of a N-gram in a phrase is strongly skewed towards either of a "accept" or "reject" diagnosis. Specifically, the significance weight was calculated as

$$\frac{\#\text{Accepted} - \#\text{Rejected}}{\#\text{Total}}$$

Accepted = Number of accepted Service notes in which N-gram occurs
#Rejected = Number of Service notes in which N-gram occurs
#Total = Total number of distinct service notes in which N-gram occurs

Phrases that do not provide evidence of a significant (user controllable) skew are thresholded out to a weight of zero. The weights of the remaining n-grams is updated to the actual weight. Once all the training phrases have been presented to the user, the system is ready for automatically assigning an accept/reject recommendation to new service notes.

Diagnosis Phase

Once a sufficient number of service notes have been presented to the user (typically 5-10%), the learned weights for the N-grams are used for determining incoming service note relevance. We discuss two approaches for doing this.

Case based approach

If a service note is considered a text based diagnostic

case, then the traditional practice would be to compare the incoming service note with all available labeled service notes and calculate the degree of match with each. The decision associated with the closest matching labeled service note would be assigned to the incoming note. The match index using learned N-gram weights is calculated as

$$\text{Weighed n-gram match index} = \frac{[\Sigma W_{Ni}]^2,\ Ni \in (Pi \cap Ci)}{\Sigma W_{Pi} \times \Sigma W_{Ci}}$$

where
P_i = unique n-grams in Parent phrase
C_i = unique n-grams in Candidate phrase
W_{Pi} = Learned weight of $P_i \in (0,1)$
W_{Ci} = Learned weight of $C_i \in (0,1)$

The drawback with this approach is that each incoming service note needs to be compared to the entire archive of labeled service notes. As the volume of incoming notes as well as the archive is quite large, this quickly becomes very time consuming and inefficient.

Direct Summing Approach

An alternate approach was devised to overcome the problem with the case based approach. The basis for the direct summing approach is that the weight associated with each N-gram is a signed number between 0 and 1. The sign of the number indicates whether the N-gram indicates an accept (+ve) or reject (-ve) decision and the magnitude indicates relative strength of that recommendation. A first cut approach involved summing up the N-gram weights for the new service note. If the sum was above a certain limit, an automatic assignment of 'accept' was given. Similarly, if it was below a certain limit, an automatic assignment of 'reject' was given. The intermediate band between the definitely accept and definitely reject limit was used as an indicator that the system was inconclusive in its recommendation and human intervention was required. Since only learned N-gram weights corresponding to the new service note were required to be retrieved, this process was much faster.

Results

A set of 6975 service notes were collected for the purpose of testing. These had been already divided into accept and reject categories by an expert. 2428 notes belonged to the accept category and 4547 notes to the reject category. Fig 4 shows typical service note comments considered diagnostically relevant.

```
1. TROUBLESHOOT FILAMENT FAULT REPLACED
   SHORTED TRANSISTORS ON DRIVER.
2. SHORTED X-FMR. WILL NEED TO REPLACE.
```

Fig 4. Examples of diagnostically relevant solutions identified by significance learning algorithm.

Some examples of service notes that were not diagnostically useful are given in Fig. 5.

```
1. PERFORMED PREVENTIVE MAINTENANCE WITH NO
PROBLEMS.
2. FOR BILLING REFERENCE ONLY
```

Fig 5. Examples of diagnostically irrelevant solutions identified by significance learning algorithm.

The user was allowed to specify what percentage of these would be used for training. The default value was 70% for training and 30% for testing. Training consisted of learning weights for N-grams culled from the service notes in the training set based upon the pre-assigned labels. A partial table of results is given below for reference. In general, the system was able to correctly sort between 80% and 90% of the service notes which was considered acceptable for the business application.

Trial	Size of N-gram	Overall accuracy %	% of accepted notes correctly classified	% of rejected notes correctly classified	% training set
1	3	84	90	80	70
2	4	87	94	83	70
3	5	81	88	77	70
4	4	88	95	84	80

Table 1. Service note classification performance

The above numbers are computed only from the service notes on which the system actually gave a decision and not those that were referred for human judgment.

Conclusions

Textual information is often the preferred format in which diagnostic information is captured due to its simplicity. This paper has presented a simple approach towards being able to 'mine' large amounts of text service information to extract small percentage of service notes with the most valuable diagnostic information. This reduced subset may now be forwarded for further cleaning and processing. An advantage of this approach is that minimal user interaction of a yes/no nature is all that is required by a person that is not necessarily require a domain expert. As time goes by, the database of labeled phrases is continually enhanced and the quality of automated

decisions progressively improves. Variable thresholds can be set to control the quality of information as well as the user interaction required. A high threshold for the diagnostic score generated by the system for new phrases will result in good quality information at the risk of excluding some valuable information, and vice versa. If the user does not wish to spend a large amount of time providing judgements, he/she may engage the automated decision process earlier and its performance will be correspondingly somewhat decreased in quality since the system has had less information to learn over. Since this approach is not domain specific, it can be quickly applied to service notes from any business to improve the quality of text-based diagnostic information available. A tool based upon this approach is currently being deployed within a GE service business.

References

Uthurusamy, R., Means, L.G., Godden, K.S. and Lytinen, S.L., "*Extracting Knowledge from Diagnostic Databases* " ,1993, IEEE Expert pp.27-37.

Creecy, R.H., Masand B.M., Smith S.J., Waltz D.L., "*Tradimg MIPS and Memory for Knowledge Engineering: Automatic classification of Census Returns on a Massively Parallel Supercomputer* ", Communications of the ACM, 1992, pp 48-64, V. 35 (8).

Varma, Anil, Wood,W.H. and Agogino, Alice M., " *A Machine Learning Approach to Automated Design Classification, Association and Retrieval*", Published in the Proceedings of the Fourth International Conference on Artificial Intelligence in Design, June 24-27, 1996, Stanford,CA

Liu, ZQ and Yan, F, "*Case-based diagnostic systems using fuzzy neural network*" , in IEEE International Conference on Neural Networks. 1995, IEEE: Perth, WA, Australia. pp. 3107-3112.