IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology
Vol. 3, Issue 6, June 2016

Developing D-Matrix of Unstructured Text Using Ontology Based Text Mining

Padmini M. Magdum[1], Prof. V. S. Nandedkar [2]

ME Student, Dept of Computer Engineering, Padmabhooshan Vasantdada Patil Institute of Technology,

Pune, Maharashtra, India[1]

Associate Professor, Dept of Computer Engineering, Padmabhooshan Vasantdada Patil Institute of Technology.

Savitribai Phule Pune University, Maharashtra, India[2]

Abstract: Every complex system interacts with its environment to perform number of tasks within acceptable tolerances. Any change in performance or unacceptable result is treated as fault or error. The Fault Detection and Diagnosis (FDD) technique is used to determine such faults or errors, to find the root cause and to take necessary actions to prevent such error to occur in the system. This will protect the system from damage and will give maximum throughput. Fault Dependency D-matrix is one such approach to find faults at different levels which consists of dependencies observed and actual failure symptoms associated with the system. But in practical it's difficult to construct d-matrix . In this paper we define a text mining method based on ontology for automatically constructing D-matrix by mining thousands of unstructured data. In this method we first construct the fault diagnosis ontology consisting of concepts and relationships usually observed and examined in the fault diagnosis domain. Then we use the text mining algorithms that make use of the ontology concept to develop the D-matrix. Then we use the text mining algorithms that make use of the ontology concept to develop the D-matrix. Next Graph is created for every D Matrix & from all generated graphs similarity graph is created.

Keywords: Text mining, D-matrix, fault detection and diagnosis, text processing.

I. INTRODUCTION

A complex system interacts with its surrounding to execute a group of tasks by maintaining their performances within an appropriate vary of tolerances. Any variation of a system from its acceptable performance is treated as a fault. The fault detection and diagnosis (FDD) is performed to observe the faults and diagnose the root-causes to reduce the period of time of a system. Because of ever growing technological sophistication that's embedded within the vehicle systems, for example refined computer code embedded systems [1], diagnostic sensors, internet, etc. the method of FDD becomes a difficult activity within the event of component or system malfunction.

Text mining is gaining a heavy attention because of its ability to mechanically discover the data assets buried in unstructured text. Fault dependency (D)-matrix is a systematic analytic model to catch the different system-level fault diagnostic information consisting of conditions between recognizable symptoms and failure modes connected with a system. Constructing a D-matrix is a difficult task and time intensive assignment. An ontology based D-matrix depict an ontology based text mining strategy for consequently constructing and redesigning a D-matrix by mining countless repair verbatim (normally written in unstructured text) gathered during the diagnosis scenes. The system constructs the fault diagnosis ontology consisting of concepts and relationships commonly saw in the fault diagnosis space. Next, utilize the content mining calculations that make utilization of this ontology to distinguish the fundamental ancient rarities, for example, parts, symptoms, failure modes, and their conditions from the unstructured repair verbatim content. In our methodology, we first construct the D-Matrices from different datasets. Next, we generate graph model for each generated d-matrix and used only common patterns from generated graph and develop newgraph. And, after this procedure, we create D-matrix depending upon data given by the similarity graph. We elaborate our work in four sections. Section II describes the literature survey. Section III describes the proposed system and its structure. Section IV describes result analysis Finally we conclude with section V.

II. LITERATURE SURVEY

1] S. Strasser, J. Sheppard, M. Schuh, R. Angryk, and C. Izurieta, "Graph based ontology- guided data mining for d-matrix model maturation," in Proc. IEEE Aero. Conference, 2011.

In this paper domain ontologies are used as a way to join together different data sources and to find discrepancies between those different data sources. Maturation approach is proposed which uses the graph-theoretic representations of Timed Failure Propagation Graph (TFPG) models and diagnostic sessions based on recently standardized diagnostic ontologies to determine statistical discrepancies between that which is expected by the models and that which has been encountered in practice.

2] Stephen K. Reed, Adam Pease, "A framework for constructing cognition ontologies using WordNet, FrameNet, and SUMO", Science Direct Cognitive Systems Research 33 (2015) 122–144

In this paper framework for constructing cognition ontologies by using WordNet, FrameNet, and SUMO. WordNet is widely used across domains in the information sciences. FrameNet captures co-occurrence and structural relations among linguistic concepts. The frames provide organized packages of knowledge that represent how people perceive, remember, and reason about their experiences. SUMO is a formal ontology consisting of an upper ontology and numerous domain ontologies. The goal for building cognition ontologies was to formulate logical axioms that encode the definitions, empirical findings and theoretical statements that have widespread support from cognitive scientists.

[3] S. Singh, S. W. Holland, and P. Bandyopadhyay, "Trends in the development of system- level fault dependency matrices," in Proc. IEEE Aerosp. Conf ., 2010, pp. 1–9.

In this paper existing research work on developing D-matrices from disparate data sources and data formats is specified. An industrial perspective is offered to describe the pros and cons of various types of D-matrices along with the challenges faced while developing and applying them for vehicle health management. The paper provide a detailed review of various ways to prepare D-matrices for real world systems.

[4] W. Zhang, T. Yoshida, X. Tang, and Q. Wang, "Text clustering using frequent itemsets," Knowl.-Based Syst., vol. 23, no. 5, pp. 379–388, 2010.

In this paper the proposed algorithm has the input as similarity matrix and output a set of clusters as compared to other clustering algorithms that predefine the count of clusters. In this work, frequent items are generated using APRIORI approach by following a similar method.

[5] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. Kavuri, "A review of process fault detection and diagnosis Part I: Quantitative model based methods," Comput. Chem. Eng., vol. 27, no. 3, pp. 293–311, 2003.

In this paper, the discussion of fault diagnosis methods that are based on historic process knowledge is given. This comparative study reveals the relative strengths and weaknesses of the different approaches. One realizes that no single method has all the desirable features one would like a diagnostic system to possess.

III.PROPOSED SYSTEM

Problem Definition

Proposed system describes an ontology based text mining method for automatically constructing and updating a D-matrix by mining hundreds of thousands of repair verbatim (typically written in unstructured text) collected during the diagnosis episodes.

Objectives of the proposed system

- Design and Implement systematic analytic model to catch the different system-level fault diagnostic information consisting of conditions between recognizable symptoms and failure modes connected with a system.
- Graph model generation for each generated d-matrix on common patterns.
- Create D-matrix depending upon data given by the similarity graph.
- A Comprehensive D-Matrix development using text mining method which is based on ontology by which store unstructured information obtained during fault recognizing & fault solving practices.

Structure of the proposed system

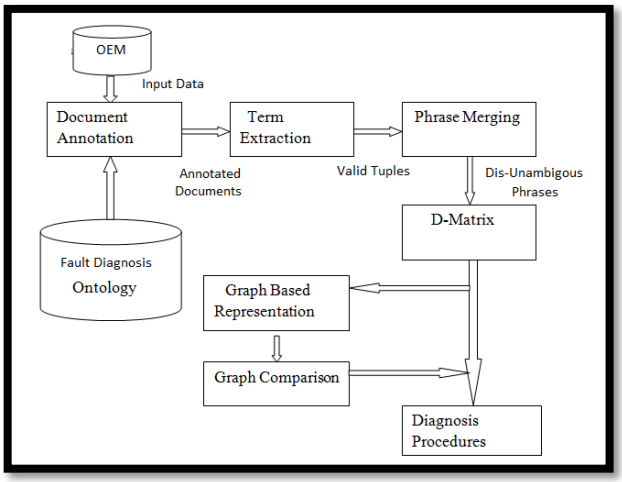

Fig. 1 System Architecture

The system consists of first the Document annotation: helps to filter out the information that is irrelevant for our analysis and it provides a specific context for the consistent and shared interpretation of the data. Initially, the following preprocessing steps—the sentence boundary detection (SBD), are used to split a repair verbatim into separate sentences, the stop words are deleted to remove the non-descriptive terms, and the lexical matching identifies the correct meaning of abbreviations. Subsequently the terms from the processed verbatim are matched using the instances in the fault diagnosis ontology. Second the Term Extractor: Having annotated the terms, the critical terms needed for the construction of a D-matrix, i.e., symptoms and failure modes are extracted by using the term extractor algorithm initially, the causal relationship between the relevant symptom-failure mode pairs is identified to make sure that only the correct pairs are extracted. Third the Phrase Merging: Due to the human intervention while capturing the repair verbatim data an inconsistency has been observed in the data in terms of the nomenclature used to record the failure modes. Hence, we check whether two different failure modes are the variations of essentially the same failure mode such that they can be merged before populating a D-matrix. The phrase merging algorithm is similar in spirit with the query expansion techniques, where each failure mode is treated

as a potential query and during the expansion stage additional information is collected in the form of attributes. Fourth the D-Matrix- D-matrix is developed from the repair verbatim data. After the creation of the D-Matrices from the different datasets, the graph is generated for each D-Matrix. Then, the graphs are combined such that only common patterns are merged from the generated heterogeneous D-Matrices to construct a single, generic D-Matrix.

Project Contribution

The existing system creates the D-Matrix for one dataset. It provides accurate d-matrix. So that, every time, the new d-matrix is created for the dataset. Even if the different datasets contains some similar data, the new D-Matrix is developed for each dataset. Our proposed system provides contribution to the existing system. In our system, multiple D-Matrices are created for different datasets. Then graph is created for each D matrix and then, the similarity between all graphs is found by using graph comparison algorithm.

Algorithm Used

Algorithm 1 - Term Extraction Input:
Output: The list T of candidate multi-word terms.
Step 1: Collect bigram frequencies for L in a proximity database DB.
Step 2: For all 4-grams w x y z in L, remove one count for x y in DB if
- mi(x, y) < mi(w, x) − k or
- mi(x, y) < mi(y, z) − k.
Step 3: For all entries (x, y) in DB, add (x, y) to a list T if:
- C(x, y) > minCount
-S(x, y) >minLogL

Algorithm 2 - **Multi-word term extraction algorithm.**
Input: A list T of two-word candidates for a corpus L in any language and a proximity
Database DB consisting of bigram frequencies for L.
Output: The list E of extracted multi-word terms.
Step 1: Accumulate features for candidate terms For each candidate c in T
For each w1 w2 …c …w2k-1 w2k in L
Add all possible substrings involving c in DB.
Step 2: Update the proximity database
Remove each entry in DB that has frequency < minFreq.
Step 3: Extend two-word candidates into an initially empty list E
For each candidate c in T
extend(c, E, DB) – see Figure 3
if most occurrences of c in the corpus have not been extended then add c to E.

Algorithm 3 Stop Word Removal
Input – set of sentences
Output – Sentences without stopwords

Step 1. Take the Input
Step 2. Declare the dictionary of stop words
Step 3. Split parameter into words
Step 4. Allocate new dictionary to store found words
Step 5. Store results in this String Builder
Step 6. Loop through all words 7. Convert to lowercase
Step 8. If this is a usable word, add it
Step 9. Return string with words removed
Step 10. Display query without stop words

Algorithm 4 - D-Matrix Generation
Input – Unstructured Information
Output – D –Matrix (tabular form)

Step 1 – Browsing information
Step 2 – Document Preprocessing
Sentence Boundary detection
Stop word removal
Steeming.
term disambiguation
Step 3 – Term Extraction
Calculate term frequencies.
Calculate probabilities
Step 4 – Check for the terms in ontology.
Step 5 - Insert values 1 in D Matrix for the terms that have highest score

Algorithm 5 - D- Matrix Graph Generation & Comparison
Step 1: D-matrix1 from datset1
Step 2: D-matrix2 from dataset2
Step 3: Graph1 from d-matrix1
Step 4: Graph2 from d-matrix2
Step 5: List of co-ordinate from graph1 and graph2 as Columns from d-matrix1 related to columns from d-matrix2 Rows from d-matrix1 related to rows from d-matrix2
Step 6: Union graph generation i.e. graph3

Mathematical Model

1. The annotated terms are extracted in the following combinations (Pi FMj) and [Sj (Pi - FMk)], where Pi is the position of a first part term and FMj are the failure modes.
2. All the failure modes appearing in a word window are selected to construct the tuples, such as [(Pi FM1), (Pi FM2), . . . ,(Pi FMk)].
3. The Symptom-Failure mode, [Sj (Pi - FMk)] tuples are constructed first by identifying the symptom, say Sj as the focal term and the parts co-occurring with Sj in the word window are selected.
4. Then, the word window of four terms is set on the either side of Pi, which is a member of (Sj Pi) and all the failure modes co-occurring with Pi are selected forming (Pi FMk).
5. Next, the tuples, (Sj Pi) and (Pi FMk) are merged using Pi as the common tuple member and the tuples (Sj Pi - FMk) are constructed.
6. Only the relevant tuples must be maintained while constructing a D-matrix corresponding to a specific system.
7. Hence, the weights are assigned to each tuple using, and the tuples with their weights above the specific threshold are considered as the valid members

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
Vol. 3, Issue 6, June 2016

$$(T_w)_{i,j} = T_{i,j} * idf_{Ti}$$

$$T_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

Where,
ni,j = The number of co-occurrences of a given tuple Ti, that appears in a repair verbatim Tj, the denominator is the sum of number of coccurrence of all tuples in Tj.

IV. RESULT ANALYSIS

Using multiple ontologies in information extraction has the potential to extract more information from text and thus leads to an improvement in performance measures. The main performance measures used in information extraction, which are precision, recall and F1-measure. Precision shows the fraction of correct extractions out of all the extractions made whereas recall shows the fraction of correct extractions made out of all possible correct extractions. With {Relevant} and {Retrieved} representing the set of all possible correct extractions and the set of extractions made respectively, precision and recall are defined as follows:

$$\text{Precision} = \frac{|Relevant| \cap |Retrieved|}{|Retrieved|}$$

$$\text{Recall} = \frac{|Relevant| \cap |Retrieved|}{|Relevant|}$$

The metrics of precision and recall are used in information retrieval as well, where {Relevant} and {Retrieved} denote the sets of all relevant documents, which should ideally be returned by an information retrieval systems in response to a query, and the set of retrieved documents respectively. It can be seen that a system can improve recall at the expense of precision by making many extractions. Similarly, precision can be improved at the expense of recall by making only few extractions that are highly likely to be accurate. Hence, a metric that takes both precision and recall into consideration is necessary to accurately evaluate the performance of an information extraction system. F1-score, which is the harmonic mean between precision and recall, is widely used for this purpose. It is defined as follows:

$$\text{F1-score} = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

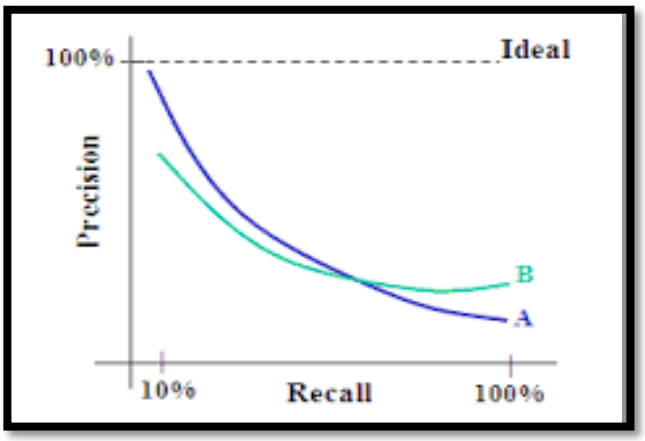

Fig 2: Precision Graph

V. CONCLUSION

A novel ontology-based text mining methodology is proposed to develop the D-matrix by automatically mining the unstructured text data collected during fault diagnosis. This framework helps the service technician to detect the faults related to complex system and diagnose it. This D-Matrix contains symptoms, failure modes and their causal relationship. Development of a graph from D-matrix gives better visualisation and analysis.

REFERENCES

[1] Dnyanesh G. Rajpathak, Satnam Singh, "An Ontology-Based Text Mining Method to Develop D-Matrix from Unstructured Text, IEEE Transactions on System, Man and Cybernetics System, Vol.44,No.7, July 2013.
[2] David W. Embley,Douglas M. Campbell, Randy D. Smith, "Ontology-Based Extraction and Structuring of Information from Data-Rich Unstructured Documents," Brigham Young University, Provo, Utah 84602, U.S.A.
[3] Ms. Madhuri M. Varma., Prof. Jyoti Nandimath, "An OntologyBased Text Mining Method To Construct D-Matrix For Fault Detection And Diagnosis Using Graph Comparison Algorithm," International Journal Of Innovative Research in Information Security (IJIRIS), Issue 2, Volume 5 (May 2015).
[4] Ayaz Ahmed Shariff K, Mohammed Ali Hussain , Sambath Kumar, "Leveraging Unstructured Data into Intelligent Information Analysis Evaluation," International Conference on Information and Network Technology IACSIT Press, Singapore vol.4 (2011)
[5] Tinal R. Thombare, Lalit Dole, "D-Matrix: Fault Diagnosis Framework," International Journal of Innovative Research in Computer and Communication Engineering, Vol. 3, Issue 3, March 2015.
[6] Raghu Anantharangachar, Srinivasan Ramani, S Rajagopalan,"Ontology Guided Information Extraction from Unstructured Text, International Journal of Web Semantic Technology (IJWesT) Vol.4, No.1, January 2013.
[7] E. Riloff., "Information extraction as a stepping stone toward story understanding". Understanding language understanding: computational models of reading, pages 435460, 1999.
[8] Kanagaraj.S and Dr.Sunitha Abburu,, "Converting Relational Database into Xml Document", IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 2, No 1, March 2012.
[9] Swati S Hinge, B. R. Nandwalkar. "A Survey on Mining of Unstructured Text for Development of D-Matrix", Int.J.Computer Technology Applications,Vol 5,Nov-Dec 2014.
[10] Vishal Gupta andGurpreet S. Lehal. " A Survey of Text Mining Techniques and Applications". Journal of Emerging Technologies In Web Intelligence, Vol. 1, No. 1, August 2009.

DOI 10.17148/IARJSET.2016.3642