Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 4, Issue. 6, June 2015, pg.405 – 408

RESEARCH ARTICLE

Improved Text Mining Approach for Conversion of Unstructured to Structured Text

Pranita Baitule[1], Prof. Vikrant Chole[2]

[1] Department of Computer Science and Engineering, GHRAET, Nagpur, India

[2]Department of Computer Science and Engineering, GHRAET, Nagpur, India

[1] pranitabaitule@gmail.com; [2] Vikrantchole@gmail.com

Abstract- Text mining refers to the process of deriving high-quality information from the text. As more knowledge and information becomes available through computers, critical capability of systems supporting knowledge management is classification of documents into categories that are meaningful to the user. We are implementing text mining method for automatically constructing and updating D-matrix by miming thousands repair verbatim collected during diagnosis period. Text mining algorithm is also use to process data on unstructured text to use this ontology to identify the primitive tool like parts, failure modes and dependencies from the unstructured text. We are also converting unstructured data to structured data. SVM Algorithm is used to classify data that improve the result.

Keywords-Text mining, data mining, information retrieval, fault diagnosis, unstructured data.

I. Introduction

Performance is retained by set of tasks when it interact a complex system with surrounding. A fault is treated as difference of the system from acceptable performance. The fault detection and diagnosis is performed to recognize and analyze the root cause to reduce the pause time of the system. Hundreds of thousands of repair verbatim are collected and disagree that there is an urgent need to mine data to improve fault diagnosis. In the process of FD its efficient utilization is restricted by the size of repair verbatim data. Normally the process of fault diagnosis starts by deriving the error codes from target system. Throughout fault diagnosis, several data are gathered such as error codes, scanned values of operating parameters cohort with faulty component system.

The collected data is transmitted to the database and the repair verbatim data gathered over a period of time can be mined for to develop the D-matrix diagnostic models. To perform the definite FDD these models can be used by field technicians and other stakeholders. The D-matrix catches component and system level dependencies between a single or a multiple failure modes with a single and multiple symptoms in a structured way.

Text mining includes an application of techniques like information retrieval, natural language processing, information extraction and data mining. Information Retrieval (IR) systems distinguish documents in collection which match a user's query. The most notable IR systems are search engines such as Google which allows recognition of set of documents that describe a set of key words. Text mining contains applying computationally-intensive algorithms to large document collections, IR can speed up discovery cycle considerably by reducing the number of documents found for analysis. consider if a researcher is interested in mining information only about protein interactions, he/she might restrict their analysis to documents that contain name of protein or some form of the verb 'to interact' or one of its synonyms. Through application of IR the vast accumulation of scientific research information can be reduced to smaller subset of related items. Natural Language Processing (NLP**)** is the analysis of human language in case computers can understand research terms in same way as humans do. Information

Extraction (IE) is the process of naturally obtaining structured data from an unstructured natural language document. This involves defining the general form of the information that researcher is interested in as one or more templates, which are then used to guide an abstraction process. IE systems depend heavily on the data generated by NLP systems. Data Mining (DM) is the process of identifying patterns in large sets of data. When used in text mining DM is applied to the facts generated by the information extraction phase.

II. RELATED WORK

Dnyanesh G.Rajpathak et.al [1] they proposed the fault diagnosis ontology. They have comparing concepts and relationships normally observed in fault diagnosis. Then they also proposed text mining algorithm. They showed how proposed method has been used to develop D-matrix using real-life data.

Vishwadeep Singh et.al [2] they have described text mining tech for automatically extracting association rule from collection of textual documents. The text mining approach includes the use of natural language processing for information extraction

Shaboo Wong et.al [3] has described automatic hierarchical domain ontology from semi structured data, from HTML and XML documents. The important process is domain term extraction, pruning, union and hierarchical structure.

III. Proposed System

In proposed work, we are using text mining algorithm for converting unstructured text to structured text for that we need to do some module work such as preprocessing, lexical analysis, fault analysis.

First of all we will do the preprocessing which include stemming and stopword. Secondly we will do the lexical analysis for domain identification that means which domain belongs to which dataset.

In lexical analysis we are using SVM i.e. support vector machine algorithm for classification here we are using two domain first one is automobile and second one is medical domain.

A] Preprocessing: Data are generally incomplete, lacking attributes value, lacking certain attributes of interest or containing only aggregate data, noisy data containing errors or outliers, inconsistent containing discrepancies in codes or name.

a] Stemming: A stemming is the process of linguistic normalization in which the variant forms of a word are reduced to common form. For example- connection, connective, connected, connecting whose stemmed form is connect. It is important to appreciate that we use stemming with the intention of improving performance of IR systems. It is not an exercising in etymology or grammar.

b] Stop word: Stop words are words which are filtered out before or after processing of natural language data. There is no single universal list of stop words used by all processing of natural language tools, and indeed not all tools even use such a list.

B] Lexical Analysis: Lexical analysis is the process of converting a sequence of characters into a sequence of tokens, i.e. meaningful character strings. A program or function that performs lexical analysis is called a lexical analyzer, lexer, tokenizer, or scanner, though scanner is also used for the first stage of a lexer. A lexer is generally combined with a parser, which together analyze the syntax of programming languages such as in compilers but also HTML parsers in web browsers.

A lexer is itself a kind of parser – the syntax of some programming languages is divided into two pieces: the lexical syntax (token structure) which is processed by the lexer and the phrase syntax, which is processed by the parser. The lexical syntax is usually a regular language, whose alphabet consists of the individual characters of the source code text.

Lexers and parsers are most often used for compilers, but can be used for other computer language tools such as pretty printers or linters. Lexing itself can be divided into two stages: the scanning, which segments the input sequence into groups and categorizes these into token classes; and the evaluating, which converts the raw input characters into a processed value.

a] SVM : Support vector machine are supervised learning models with associated learning algorithm that analyze data and recognize pattern, it is used for classification and regression analysis. Given a set of training examples each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other, making it a non-probabilistic binary linear classifier. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible.

A support vector machine constructs a hyper plane or set of hyper planes in high or infinite dimensional space which can be used for classification, regression, or other tasks. Probably a good separation is achieved by the hyper plane that has the largest distance to the nearest training data point of any class as in general the larger the margin the lower the generalization error of the classifier.

To keep the computational load reasonable the mappings used by SVM schemes are designed to ensure that dot products may be computed easily in terms of variables in the original space, by defining them in terms of a kernel function $k(x, y)$ selected to suit the problem. The

hyper planes in the higher dimensional space are defined as the set of points whose dot product with vector in that space is constant. The vectors defining the hyper planes can be chosen to be linear combinations with parameters α_i of images of feature vectors that occur in the data base. With this choice of a hyper plane, the points x in the feature space that are mapped into the hyper plane are defined by the relation:

$\sum_i \alpha_i k(x_i, x) = \text{constant.}$ if $k(x, y)$ becomes small as y grows further away from x, each term in sum measures the degree of closeness of the test point x to the corresponding data base point x_i. In this way sum of kernels above can be used to measure the relative closeness of each test point to the data points originating in one or the other of the sets to be discriminated. The fact that a set of points x mapped into any hyper plane can be quite involved as a result, allowing much more complex discrimination between sets which are not convex at all in the original space.

C] Fault Analysis: Any deviation of the system from its acceptable performance is called as fault. Fault analysis is used for detecting the fault. Here, we are showing the fault if there is a fault in data that means the domain related data concern with their own domain so there is no fault and if data or any abbreviations are not related to particular domain then there is a fault. Fault is indicating by 0 and 1 if fault occurs it will be indicated by 1 if not it will be shown by 0.

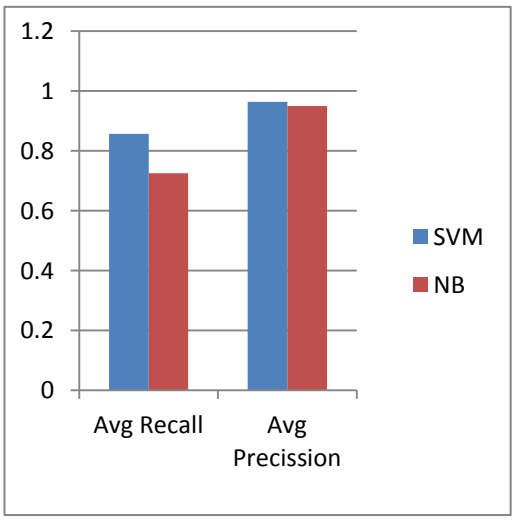
Fault detection and diagnosis is key component of many operations management automation systems. A fault is another word for a problem. A root cause fault is a fundamental, underlying problem that may lead to other problems and observable symptoms. A root cause is also generally associated with procedures for repair.

A fault or problem does not have to be the result of complete failure of piece of equipment or even involve specific hardware. For instance a problem might be defined as non-optimal operation or off-spec product. In a process root causes of non optimal operation might be hardware failures, but problems might also be by poor choice of operating targets, poor feedstock quality, poor controller tuning, and partial caused loss of catalyst activity by human error. A fault may be considered a binary variable or there may be numerical extent such as the amount of a leak or a measure of inefficiency.

IV. Experimental Result

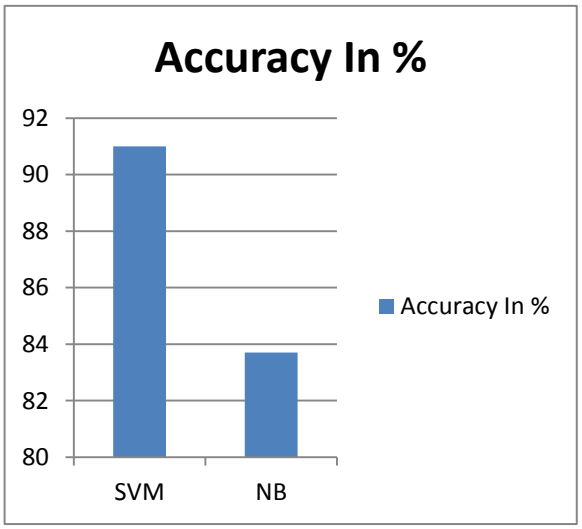
SVM:

	Recall	Precision
Dataset1	0.8	1
Dataset2	0.86	1
Dataset3	0.9	0.8181
Dataset4	0.722	1
Dataset5	1	1



NB:

	Recall	Precision
Dataset1	0.66	1
Dataset2	0.76	0.92
Dataset3	0.75	0.81
Dataset4	0.72	1
Dataset5	0.727	1



Above two table is for Support vector machine (SVM) and naïve bayes (NB) showing the dataset values and its recall and precision. These two table are for comparing SVM and NB for finding the accuracy. We can show from above graph SVM has the higher accuracy than NB.

V. Conclusion

In this paper we present improved version of our previous research by comparing experimental results on SVM and naive bayes classify. We are showing the conversion of unstructured to structured text by applying text mining algorithm. Fault detection is also applied to detect the fault.

References

[1] Dnyanesh G. Rajpathak, Satnam Singh, "An Ontology-Based Text Mining Method to Develop D-Matrix from Unstructured Text" IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS, VOL. 44, NO. 7, JULY 2014.

[2] Vishwadeepak Singh "Text Mining Approaches To Extracts Interesting Association Rules From Text Documents" IJCSI International Journal of Computer science tssues vol.9 issue 3, no.3 May2012.

[3] Shaobo Wang1, Yi Zeng1, and Ning Zhong1,2," Ontology Extraction and Integration from Semi-structured Data" International WIC Institute, Beijing University of Technology, Maebashi Institute of Technology, Japan.

[4] Ching Kang Cheng, Xiao Shan Pan Franz Kurfess, "Ontology-based Semantic Classification of Unstructured Documents" California Polytechnic State University San Luis Obispo, California 93407, USA.

[5] Dimitrios Skoutas, Alkis Simitsis, "Ontology-Based Conceptual Design of ETL Processes for Both Structured and Semi-Structured Data", International Journal on Semantic Web & Information Systems, Volume 3, Issue 4,2007.

[6] Jantima Polpinij ,"Ontology-based Knowledge Discovery from Unstructured Text" International Journal of Information Processing and Management(IJIPM) Volume4, Number4, June 2013.

[7] P. M. Frank, "Fault detection in dynamic systems using analytical and knowledge-based redundancy-a survey and some new results," Automatica, vol. 26, no. 3, pp. 459–474, 1990.

[8] J. Gertler, M. Costine, et. al,"Model based diagnosis for automotive engines—algorithm development and testing on a production vehicle," IEEE Trans. Control Syst. Technol., vol. 3, no. 1, pp. 61–69, Mar. 1995.

[9] J.Gertler et.al "A new structural framework for parity equation-based failure detection and isolation," Automatica, vol. 26, no.2, pp. 381–388, 1990.

[10] V. VenkataSubramanian et .al, "A review of process fault detection and diagnosis Part I: Quantitative modelbased methods," Comput. Chem. Eng., vol. 27, no. 3, pp. 293–311, 2003.

