International Journal of Soft Computing 2 (2): 346-348, 2007

Association Rule Generation from Textual Document

S. Ghanshyam Thakur, Rekha Thakur and R.C. Jain
Department of Computer Application, Samrat Ashok Technological Institute, Vidisha (m.p.), India

Abstract: In this research study we have done the work for extracting the useful knowledge from textual documents. Text documents are important because there are lots of attractive information hide in documents. Text mining is the process of extracting interesting information and meaningful knowledge from huge 'chunk' of unstructured or semi structured text documents. We describe the concept of Binary Matrix Model (BMM) in this research study. We applied the data mining methods on this matrix model for classification of documents.

Key words: Text mining, association, classification

INTRODUCTION

Everyday a vast amount of documents, reports, e-mails and web pages are generated from different sources, such as enterprises, governments, organizations and individuals. The unstructured data is usually not stored on relational or transaction database systems, but on web servers, files servers, or even personal workstations. They pursue a systematic and automatic approach in organizing these documents without human intervention or preparation work.

Text mining (Joachims, 1998) can be also defined similar to data mining as the application of algorithms and methods from the field's machine learning and statistics to texts with the goal of finding useful patterns. For this purpose it is necessary to pre-process the texts accordingly. Many authors use information extraction methods, natural language processing or some simple pre-processing steps in order to extract data from texts. To the extracted data then data mining algorithms can be applied.

WORK ALREADY DONE

All the researchers worked in the area of classification (Liu *et al*., 1998; Koller and Sahami, 1997; Han and Fu, 1995; Agrawal and Srikant, 1994; 1995) and developed data mining algorithms for classification, association and association based classification. In order to achieve this, one frequently relies on the experience and results of research in information retrieval, natural language processing and information extraction.

MATERIALS AND METHODS

Data pre-processing: Example D = (d1,d2,d3,d4,d5,d6,d7, d8,d9,d10,d11,d12,d13,d14,d15). Each document has stop words, punctuation marks and special marks. In this section we remove the stop word and perform the word stemming. A text document is split into a stream of words by removing all punctuation marks and by replacing tabs and other non-text characters by single white spaces.

Remove stop words: Stop words are "The", "And", "A", "Is", "Am", "Are", "Was", "Were" etc.. We remove the stop words from the documents (d1,d2,d3,d4,d5,d6,d7, d8,d9,d10,d11,d12,d13,d14,d15). The idea of stop word filtering is to remove words that bear little or no content information, like articles, conjunctions, prepositions, etc.

Words stemming: Stemming is the process of suffix removal to generate word stems. Stemming methods try to build the basic forms of words, i.e. strip the plural 's' from nouns, the 'ing' from verbs, or other affixes. A stem is a natural group of words with very similar meaning. After the stemming process, every word is represented by its stem.

Algorithms
Input:

- Documents d_i
- List of stopword L

Output: A list T of valid stemmed text terms
T = NULL;

Corresponding Author: S. Ghanshyam Thakur, Department of Computer Application, Samrat Ashok Technological Institute, Vidisha (m.p.), India

While(!eof(d_i))
{
Extract the term w from the document d_i
If(w ∉ L)
t_{stem} = wordstem(w);
T = T ⋃ $t_{stem;}$
}

Term selection: After stop word removing and word stemming the document has number of unused terms. We categories the stem words into two categories:

- Max frequency stem words.
- Min frequency stem words.

For categorisation stem words we choose the "threshold value" if the frequency of a term is greater or equal to the threshold value then they are in max frequency stem words otherwise they are in min frequency stem words. We remove all the min frequency stem words from the documents.

Algorithms
Input:

- Documents d_i
- Threshold value L

Output: A list T of valid text term
T = NULL;
While (!eof(d_i))
{
Extract the term w from the document $d_{i;}$
N = termfrec(w) ;
If(N≥L)
T = T ⋃ $w_{;}$
}

BINARY MATRIX MODEL (BMM)

After term selection we have limited terms in each documents suppose we have n documents and maximum m stem words in a documents. The binary matrix M is represented as.

$M[d_i \times w_j] = 1$, if w_j ª d_i
= 0, otherwise
Where i = 1,2,3, ……, n
j = 1,2,3, ……, m

In binary matrix model each row represents as a vector this means that each documents can be represented as a vector. E.g. in the given model document d1 is represented as a vector document as follows:
d1→(1,0,0,0,1,1,0,1,0)

This model used as a classifier and can apply various methods for documents classification.

	Feature vectors								
Documents	1	2	3	4	5	6	7	8	9
d1	1	0	0	0	1	1	0	1	0
d2	0	1	0	1	0	0	0	1	0
d3	0	0	0	1	1	0	1	0	0
d4	0	1	1	0	0	0	0	0	0
d5	0	0	0	0	1	1	1	0	0
d6	0	1	1	1	0	0	0	0	0
d7	0	1	0	0	0	1	1	0	1
d8	0	0	0	0	1	0	0	0	0
d9	0	0	0	0	0	0	0	1	0
d10	0	0	1	0	1	0	1	0	0
d11	0	0	1	0	1	0	1	0	0
d12	0	0	0	0	1	1	0	1	0
d13	0	1	0	1	0	1	1	0	0
d14	1	0	1	0	1	0	1	0	0
d15	0	1	1	0	0	0	0	0	1

1 = Sound, 2 = Color, 3 = Image, 4 = Graphics, 5 = Picture, 6 = Compiler, 7 = Program, 8 = Assmbler, 9 = Interpreter

Example: T = (d1,d2,d3,d4,d5,d6,d7,d8,d9,d10,d11,d12,d13, d14,d15)
I = {Sound, color, image, graphics, Picture, compiler, program, assembler, interpretor}
Min sup = 15%

Apply a-priori on the above example:
K = 1;
C = {Sound, color, image, graphics, Picture, compiler, program, assembler, interpretor}
F1 = {color, image, graphics, Picture, compiler, program, assembler}
K = 2;
C2 = {(color, image), (color, graphics), (color, Picture), (color, compiler), (color, program), (color, assembler), (image, graphics), (image, Picture), (imag, compiler), (image, program), (image, assembler), (graphics, Picture), (graphics, compiler), (graphics, program), (graphics, assembler), (Picture, compiler), (Picture, program), (Picture, assembler), (compiler, program), (compiler, assembler), (program, assembler) }
F2 = {(color, image), (color, graphics), (image, Picture), (image, program), (Picture, compiler), (Picture, program), (compiler, program) }
K = 3;
C3 = {(color, image, graphics), (image, Picture, program), (Picture, compiler, program) }
F3 = {(image, Picture, program)}
F = f1\/f2\/f3

Association rules: 70% conf
R1: Image/\ Picture _ program
R2: Image/\ program_ Picture

CONCLUSION

In this research study we have derived association rules. On the basis of we can decide that whose documents are very important i.e. which are more associative.

REFERENCES

Agrawal, R. and R. Srikant, 1994. Fast Algorithm for Mining Association Rules. In J. B. Bocca, M. Jarke and C. Zaniolo, (Eds.), Proc. 20th Int. Conf. Very Large Data Bases, VLDB, Morgan Kaufmann, pp: 487-499.

Han, J. and Y. Fu, 1995. Discovery of multiple-level association rules from large databases. In Proc. 21st Int. Conf. Very Large Data Bases (VLDB), Zurich, Switzerland, pp: 420-431.

Joachims, T., 1998. Text categorization with support vector machines: learning with many relevant features. In Proc. 10th European Conf. Machine Learning (ECML), Chemnitz, Germany, pp: 137-142.

Koller, D. and M. Sahami, 1997. Hierarchically classifying documents using very few words. In Proc. 14th Int. Conf. Mach. Learn. (ICML), Nashville, TN, pp: 170-178.

Liu, B., W. Hsu and Y. Ma, 1998. Integrating classification and association rule mining. In Proc. 4th Int. Conf. Knowledge Discovery and Data Mining (KDD), New York, pp: 80-86.

Srikant, R. and R. Agrawal, 1995. Mining generalized association rules. In Proc. 21st Int. Conf. Very Large Data Bases (VLDB), Zurich, Switzerland, pp: 407-419.

Zhang, Y. and G. Ling, 2005. An improved TF-IDF approach for text classification. J. Zhejiang Uni. Sci., 6A: 49-55.