A Novel Approach for Web Document Classification

Rajendra Kumar Roul
BITS, Pilani - K.K. Birla, Goa Campus
Zuarinagar, Goa - 403726, India
rkroul@goa.bits-pilani.ac.in

Abstract—The web is a huge repository of information and there is a need for categorizing web documents to facilitate the search and retrieval of documents. Web document classification plays an important role in information organization and retrieval.This paper presents a fuzzy set based approach for automatically classifying web documents into one of the classes represented by a set of training documents belonging to a number of classes. Using same word to represent more than one meaning and many words representing one meaning lead to ambiguity especially in web environment where numbers of users are very large. This problem is tackled using fuzzy association wherein each pair of words has a value associated with it. This helps in distinguishing it with other such pairs of words and thus helps in tackling ambiguities. The approach present in this paper does not require any parameter to be given by the user and hence is independent of any bias that may occur due to user input. It requires a training set on which the model is trained and then test set is given as input to be classified. We used Gensim package to implement the approach because of its simplicity and robust nature. The experimental results show that our approach efficiently classifies the web documents by tackling ambiguities among the words.

Keywords- Classification, Fuzzy Association, Fuzzy Related Term, Gensim, Information Retrieval.

I. Introduction

In present times, when we have billions of web documents on the internet, we cannot rely on the conventional methods of searching. With this much amount of information available it is not possible to take the full advantage of the World Wide Web without having a proper framework to search through the available data. As the documents available are not organized properly it takes an extra effort from the user to retrieve the search results. There are umpteen number of web documents related to a topic which may or may not be relevant to the search query. The existence of an abundance of information, in combination with the dynamic and heterogeneous nature of the web, makes information retrieval a difficult process for the average user. This has led to the need for the development of new techniques which can assist the users effectively to navigate, trace and organize the available web documents according to the best match of their needs. This requisite organization can be done in many ways. One of the techniques that can play an important role towards the achievement of this objective is documents classification. Classification is one of the main data analysis techniques and deals with the categorizing a new data entry into one of the categories based on the values of different attributes. Generally classification algorithms need to train a model based on the training set. Then after the training phase gets complete, one can subject the test set for evaluation through that trained model. This brings the classification process to an end. Document classification or text categorization (as used in information retrieval context) is the process of assigning a document to a predefined set of categories based on the document content. It can be applied as an information filtering tool and can also be used to improve the retrieval results from a query process. This enhances the performance and decreases the burden of search engine used for information retrieval. Classification basically divided into two categories. The first one is *Eager learners*, where given a set of training tuples, it will construct a classification model before receiving new (i.e test) tuples to classify. We can think of the learned model as being ready and eager to classify previously unseen tuples. Different classification techniques such as decision tree induction, bayesian classification, rule based classification, classification by back propagation, support vector machine, classification based on association rule mining etc are fall into this category. The other one of classification is *lazy learner*, in which the learner waits till the last minute before doing any model construction in order to classify a given tuple. In another way, given a training tuple, a lazy learner simply stores it(or does only a little minor processing) and waits until it gets a test tuple. Only when it sees the test tuples, it performs generalization (i.e classification) in order to classify the tuples based on its similarity to the stored training tuples. Classification techniques such as k-nearest neighbor, cased-based reasoning etc are belonging to this category.

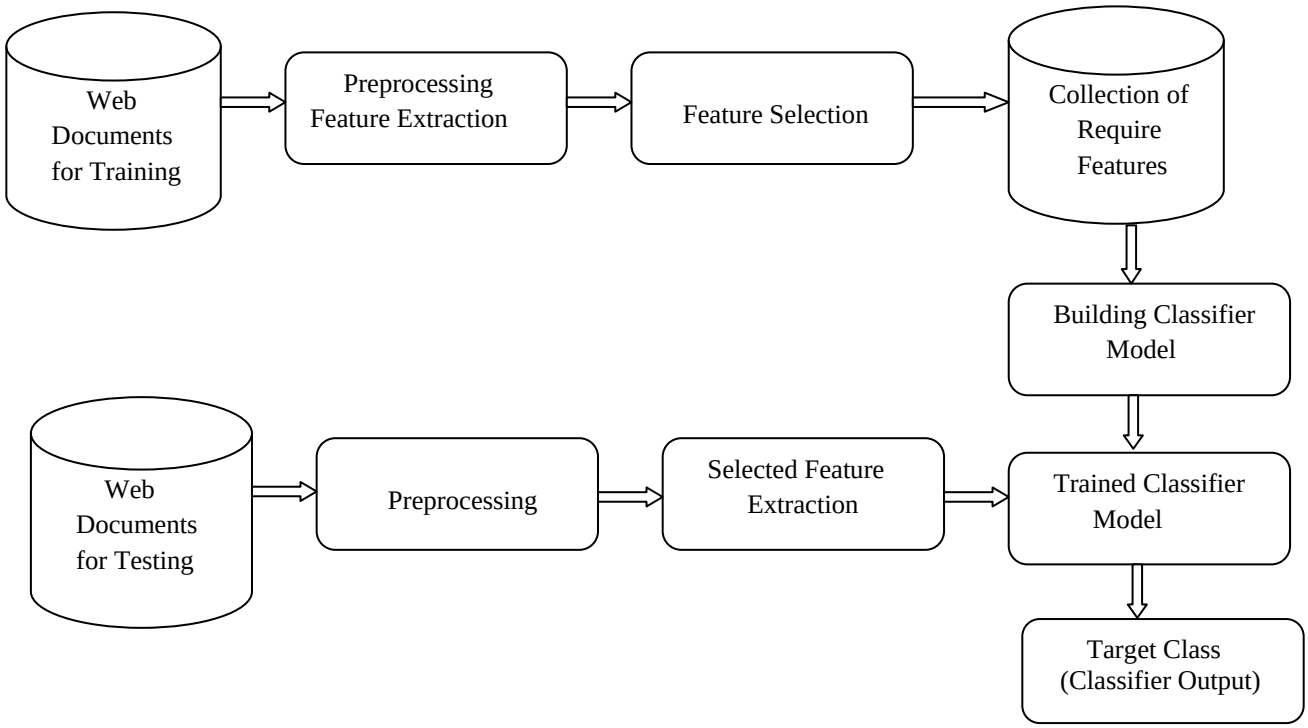

Figure 1. System Architecture.

Lazy learners can be computationally expensive when making classification as they require efficient storage techniques. It does less work when a training tuple is presented and more work when making a classification. Lazy learners, however, naturally support incremental learning. It is also known as instance based learners as it is storing the training tuples. Some other classification techniques are there, like genetic algorithms, rough set approach, fuzzy set approach etc. Basically classification of web queries helps us by refining the search results obtained using some search metric. It reduces the web navigation distance by understanding the usage behaviour of the user. Besides the text content of the web page, the images, video and other multimedia content and the structure of the document also provide a lot of information aiding in the classification of a document.

The classification techniques can be broadly categories as follows:

- Link and Content Analysis.
- A combination of document content and META tags.
- Manual classification by domain specific experts.
- META tags (which serve the purpose of document indexing).
- Clustering approaches.
- Solely on document content.

In this paper, we proposed an approach for automatically classifying the web documents into a set of categories using the *fuzzy association* concept. The fuzzy association uses the concept of the *Fuzzy Set* theory [9] to model the vagueness in the information retrieval process. The basic concept of fuzzy association involves the construction of a *pseudothesaurus* of keywords or index terms from a set of documents [10]. By constructing a *pseudothesaurus*, the relationship among different index terms or keywords in the documents is captured, i.e., each pair of words has an associated value to distinguish itself from other pairs of words. Therefore, the ambiguity in word usage is minimized.

The remainder of this paper is organized on the following lines: section 2 covers the related work based on different classification techniques used for web documents. Section 3 describes the materials and methods used in our approach. In section 4, we describe the proposed approach adopted for classification. The results and discussions are covered in section 5 and finally conclusions and future work are presented in section 6.

II. Related Work

Web document classification has been widely studied in the past few years. Much research work has been done in this area. Chakrabarti et al.[1] used predicted labels of neighboring documents to reinforce classification decisions for a given document. R.Ghani et al.[2] found that meta data is a useful source of information and the combination of meta data with text can result in better performance. A dynamic and

hierarchical classification system that is capable of adding new categories as required, organizing the web pages into a tree structure, and classifying web pages by searching through only one path of the tree structure is proposed in [3]. The test results show that the proposed single path search technique reduces the search complexity and increases the accuracy by 6% comparing to related algorithms. H.D Hussain et al.[4] presents a novel ontology based web page classification method for the knowledge grid environment, which utilizes generated metadata from web pages as the inter medium to classify the web pages by ontology concepts. Oh et al.[5] proposed a practical method for exploiting hypertext structure and hyperlink information . They modified the naive bayes algorithm to classify documents by using neighboring documents that were similar to the target document. Both the predicted labels and the text contents of the neighboring documents were used to assistant classification. The experimental results on an encyclopedia corpus that contains hyperlinks validate their algorithms. J. F¨urnkranz [6] also reported a significant improvement in classification accuracy when using the link based method as opposed to the full-text alone on 1,050 pages of the web KB corpus, although adding the entire text of "neighbor documents" seemed to harm the ability to classify pages [1]. A. Sun et al. [7] claimed that the combination of the plain text, the anchor text and the title can get a large improvement on F-measure compared with full-text. Liu et al. [8] present an Entity-Based web page classification algorithm, which can be embedded in search engines easily. In this algorithm, an entity system is built to classify web pages immediately before indexing jobs .Some research has been done to enhance categorization by summarization [11][14][15], but these works handle pure text categorization only. Most research efforts have assumed that the text components of web pages provide the primary information for web classification while the other non text components can be used to improve the classification performance [1, 5, 12, 13].Our approach automatically classify the web documents into a set of pre defined classes by the help of fuzzy association which use the concept of fuzzy set theory by handling ambiguities among the words.

III. Materials and Methods

A. Vector Space Model

In vector space model [16], each document is defined as a multidimensional vector of keywords in euclidean space whose axis correspond to the keyword i.e., each dimension corresponds to a separate keyword. The keywords are extracted from the document and weight associated with each keyword determines the importance of the keyword in the document. Thus, a document is represented as, $D_j = (w_{1j}, w_{2j}, w_{3j},....w_{nj})$ where, w_{ij} is the weight of term i in document j indicates the relevance and importance of the keyword.

B. Gensim

Gensim package [17] is a python library for vector space modeling, aims to process raw, unstructured digital texts (plain text). It can automatically extract semantic topics from documents, used basically for the Natural Language Processing (NLP) community. Its memory (RAM) independent feature with respect to the corpus size allows to process large web based corpora. In Gensim, one can easily plug-in his own input corpus and data stream. Other vector space algorithms can be trivially incorporated in it. In Gensim, many unsupervised algorithms are based on word co-occurrence patterns within a corpus of training documents. Once these statistical patterns are found, any plain text documents can be succinctly expressed in the new semantic representation and can be queried for the topical similarity against other documents and so on.

C. Fuzzy Association

Fuzzy association has been widely used in the area of information retrieval. Fuzzy set theory deals with the representation of classes whose boundaries are not well defined. The main aim is to associate a membership function with the elements of the class. This function takes values on the interval [*0, 1*] with *0* corresponding to no membership in the class and *1* corresponding to whole membership. Membership values between *0* and *1* indicate *marginal* elements of the class. Thus, membership in a fuzzy set is a notion intrinsically *gradual* instead of abrupt or *crisp*. To improves the retrieval results from traditional information retrieval (IR) system, fuzzy association IR captures the association between the keywords. By providing such association between the keywords, some additional documents that are not directly indexed by the keywords in the query can also be retrieved.

Before going into the details of the algorithm, we have discussed few important descriptions which form the base of the whole process.

Description 1:

A fuzzy association between two finite sets $A= \{a_1,\ldots,a_u\}$ and $B=\{b_1,\ldots,b_v\}$ is formally defined as a binary fuzzy relation $f: A \times B \rightarrow [0,1]$, where u and v are the numbers of elements in A and B, respectively. The construction of association between index terms or keywords is generally known as the generation of the fuzzy pseudothesaurus which can be summarized below by description 2 and description 3.

Description 2:

Given a set of index terms, T={t_1,…,t_u}, and a set of documents, D={d_1,…,d_v}, each t_i is represented by a fuzzy set h(t_i) of documents; h(t_i) = { F(t_i, d_j) | ∀d_i ∈ D }, where F(t_i, d_j) is the significance (or membership) degree of t_i in d_j.

Description 3:

The fuzzy related terms (RT) relation is based on the evaluation of the co-occurrences of t_i and t_j in the set D and can be defined as follows.

$$RT(t_i,t_j) = \frac{\sum_k \min\,(F(t_i,d_k),F(t_j,d_k))}{\sum_k \max\,(F(t_i,d_k),F(t_j,d_k))} \quad (1)$$

A simplification of the fuzzy *RT* relation based on the co-occurrence of keywords is given as follows.

$$r_{i,j} = \frac{n_{i,j}}{n_i+n_j-n_{i,j}} \quad (2)$$

where,

- $r_{i,j}$ represents the fuzzy *RT* relation between keywords *i* and *j*,
- $n_{i,j}$ is the number of documents containing both i^{th} and j^{th} keywords,
- n_i is the number of documents including the i^{th} keyword, and
- n_j is the number of documents including the j^{th} keyword

IV. Proposed Approach

Input:

1. Set of training documents TD = {TD_1, ... TD_m} each belongs to one of the predefined classes from C = {C_1, ... C_m}. Each TD_i consists of a set of documents belonging to class i.
2. Set of test documents T = {T_0, ... T_i}.

Output:

Each document belonging to the test set, T, is assigned to appropriate class.

Steps:

1. *Preprocessing training documents:*
   - Each training document, TD_i, is parsed to text.
   - Stop words are removed from each TD_i.
   - In each TD_i only the nouns are treated as keywords.
   - Stemming has been done on each TD_i using porter stemming algorithm [18].
2. *Extraction of keywords:*

   The most frequently occurred keywords from the training document sets(TD_i) based on each category are extracted and put into separate keyword sets, $K = \{K_1, K_2, \ldots, K_m\}$. From these *m* sets of keywords, we combined them into a set of all keywords, $A = \{k_1, k_2, \ldots, k_n\}$,where *n* is the total number of all distinct keywords representing the vector dimension. Since some keywords can appear in more than one category, we consider only one instance of these.
3. *Generation of keyword correlation matrix M:*

   After extraction of keywords, we generate the keyword correlation matrix *M* using the fuzzy *RT* relation equation (given in Eq. 2). The keyword correlation matrix is an $n \times n$ symmetric matrix whose element, r_{ij}, has the value on the interval [*0, 1*] with *0* indicates no relationship and *1* indicates full relationship between the keywords k_i and k_j. Therefore, r_{ij} is equal to 1, ∀i = *j*, since a keyword has the strongest relationship to itself.
4. *Representing categories:*

   Before being able to classify the query, we need to create a representative set of keywords for each of the classes present. The best way to represent each category is to select only the exclusive keywords, i.e., for category C_i, we consider the keywords in K_i which do not belong in another keyword sets K_j, where $j=1\ldots m$ and $j \neq i$. We refer to this as the *category keyword sets*, $CK = \{CK_1, CK_2, \ldots, CK_m\}$.
5. *Preprocessing test documents/queries:*

   Now the test set of documents, T, is preprocessed in the same way as mentioned in step 1 and the keywords are extracted.
6. *Classification of test documents:*

   Step 5 gives us the transformed representation of the test set T, which is now referred to as

$D = \{d_1, \ldots, d_p\}$ in its preprocessed form, where p is the total number of documents to be classified. After that, the membership degrees between each document to each of the category sets are calculated using the following equation:

$$\mu_{i,j} = avg_{\forall k_a \in d_i}\left[1 - \prod_{\forall k_b \in CK_j}(1 - r_{a,b})\right] \quad (3)$$

where,

- CK_j = keywords belonging only to a given class j,
- d_i = represents keywords belonging to document i.
- $\mu_{i,j}$ is the membership degree of d_i belonging to class j, and
- $r_{a,b}$ is the fuzzy relation between keyword $K_a \in d_i$ and keyword $K_b \in CK_j$.

7. A document d_i is classified to class C_j, for which the value obtained from the above step is maximum.

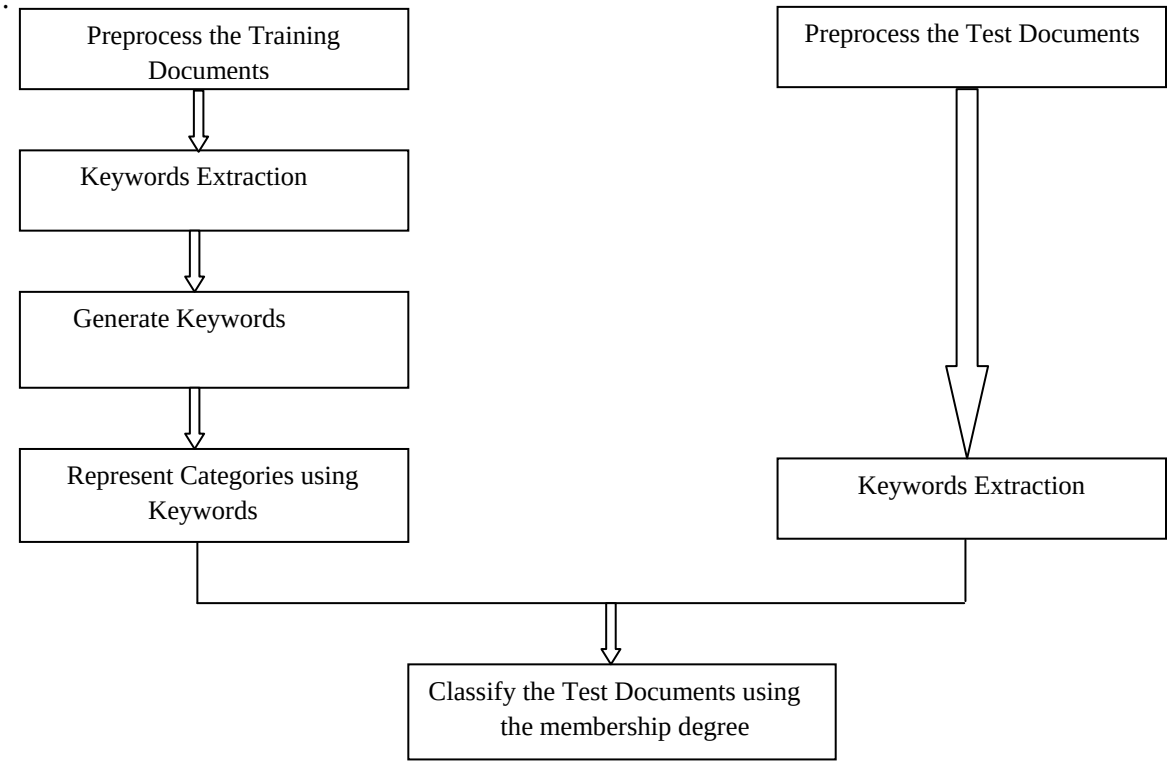

Figure 2. Classification of Test Documents using fuzzy set approach.

V. Results and Discussions

To illustrate our approach, we took eight training documents. After preprocessing, we extract noun as keywords which shown in table I with their corresponding predefined class name. Table II shows the entire keywords of the training set with their corresponding word_id. Similarly we preprocess all the test documents and extract keywords shown in table III. Fuzzy RT relation matrix, shown in table IV formed after applying Eq. 2 to the keywords of training documents. We separate each keyword which belongs to a particular class from all the keywords of training sets that reflect in table V. Finally table VI shows class membership of test documents where a test document belongs to that target class whose class membership value is maximum among all the target classes for that particular test document. Below we have shown the calculation of class membership of test documents. For calculating class membership, the average value (over all keywords in the test documents) of the variable μ(defined in Eq. 3) is taken for each class and the class for which it is highest is assigned to the test document.

According to Eq. 3, keywords for each of the classes from all keywords of training sets are as follows:

Class : 0 ['laptop', 'air', 'apple', 'macbook']

Class : 1 ['hollywood', 'california']

Class : 2 ['model', 'car', 'audi']

Class : 3 ['steak', 'oriental', 'beef', 'iron', 'mustard']

Class : 4 ['aguero', 'century', 'messi', 'partnership', 'sachin']

Each index of membership function gives the membership of test document to class corresponding to that index.

Class membership of test documents:

Test Doc # 1: apple, macbook, model, computer
Membership Function: [0.6666, 0, 0.3333, 0, 0]
Class Membership: computers

Test Doc # 2: dhoni, triple, century, partnership
Membership Function: [0, 0, 0, 0, 0.5]
Class Membership: sports

Test Doc # 3: oriental, wine,steak
Membership Function: [0, 0, 0, 0.6666, 0]
Class Membership: food and cooking

Test Doc # 4: oriental, car, model
Membership Function: [0.2222, 0, 0.6666, 0.3333, 0]
Class Membership: companies and industries

Test Doc # 5: hollywood, movie
Membership Function: [0, 0.5, 0, 0, 0]
Class Membership: entertainment

VI. Conclusions and Future Work

Fuzzy set approach for classification is very efficient way that it takes into consideration the ambiguity by capturing the relationship or associations among different index terms or keywords. It uses the concepts of fuzzy set theory. By constructing a *pseudothesaurus*, which is the basic concepts of fuzzy association, we captured the relationship among different index terms or keywords in the documents i.e., the result is, each pair of words has an associated value to distinguish itself from other pairs of words. Therefore, the ambiguity in word usage is minimized. Moreover, as in real life any object can have membership of more than one class. Our approach classifies each document into a particular class which gives us a broader understanding of the classes and documents which belong to them. We will further extend this work by ranking each class of documents using topic based modeling.

Acknowledgment

We are thankful to Prof.Bharat Deshpande, Head of the Computer Science Department of BITS, Pilani-K.K.Birla Goa Campus and our colleagues Aruna Govada and K.V. Santhilata for their useful discussions and valuable suggestions.

References

[1] S. Chakrabarti, B. Dom, and P. Indyk. Enhanced hypertext categorization using hyperlinks. In SIGMOD ’98: Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data, pages 307–318, New York, NY, USA, 1998.

[2] R. Ghani, S. Slattery, and Y. Yang. Hypertext categorization using hyperlink patterns and meta data. In Proceedings of the Eighteenth International Conference on Machine Learning (ICML-2001), pages 178– 185, San Francisco, CA, USA, 2001.Morgan Kaufmann Publishers Inc.

[3] Xiaogang Peng, Ben Choi (2002), “Automatic Web Page Classification in a Dynamic and Hierarchical Way”, In Proceedings of Second IEEE International Conference on Data Mining, Washington DC, IEEE Computer Society, pp.386-393.

[4] Hai Dong Hussain, F. K. Chang E (2009), “An Ontology based Web Page Classification Approach for the Knowledge Grid Environment”, In Proceedings of the 5th International Conference on Semantics, Knowledge and Grid, Oct 12, China : IEEE Computer Society, pp. 120 – 127.

[5] H.-J. Oh, S.-H. Myaeng, and M.-H. Lee. A practical hypertext categorization method using links and incrementally available class information. In SIGIR ’00: Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages Athens, Greece, 2000, pp. 264–271.

[6] J. F¨urnkranz. Exploiting structural information for text classification on the www. In IDA ’99: Proceedings of the 3rd Symposium on Intelligent Data Analysis, pages 487–498, 1999.

[7] A. Sun, E.-P. Lim, and W.-K. Ng. Web classification using support vector machine. In Proceedings of the 4th International Workshop on Web Information and Data Management (WIDM 2002), pages 96–99, New York, NY, USA, 2002. ACM Press.

[8] Yicen Liu, Mingrong Liu, Liang Xiang and Qing Yang, (2008), “Entity-Based Classification of Web Page in Search Engine”, ICADL, LNCS, Vol. 5362, pp. 411- 412.

[9] L.A. Zadeh, “Fuzzy Sets,” in D. Dubois, H. Prade, and R.R.Yager, editors, Readings in Fuzzy Sets for Intelligent Systems, Morgan Kaufmann Publishers, 1993.

[10] S. Miyamoto, T. Miyake, and K. Nakayama, “Generation of a Pseudothesaurus for Information Retrieval Based on Cooccurrences and Fuzzy Set Operations,” IEEE Transactions on Systems, Man, and Cybernetics, vol. 13, no.1, 1983, pp. 62-70.

[11] S.J. Ker and J.-N. Chen. A Text Categorization Based on Summarization Technique. In the 38th Annual Meeting of the Association for Computational Linguistics IR&NLP workshop, Hong Kong, October 3-8, 2000.

[12] M. Craven and S. Slattery. Relational learning with statistical predicate invention: Better models for hypertext. Machine Learning, 43(1-2):97-119, 2001.

[13] Y. Yang, S. Slattery, and R. Ghani. A study of approaches to hypertext categorization. Journal of Intelligent Information Systems, 18(2-3):219-241, 2002.
[14] Y.J Ko, J.W Park, J.Y. Seo. Automatic Text Categorization using the Importance of Sentences. Proc. of COLING 2002.
[15] A. Kolcz, V. Prabakarmurthi, J.K. Kalita. Summarization as feature selection for text categorization. Proc. of CIKM01, 2001.
[16] http://www.miislita.com/term-vector/termvector-3.html.
[17] http://www.nlp.fi.muni.cz/projekty/gensim/intro.html.
[18] http://tartarus.org/martin/PorterStemmer/def.txt.

Appendix

Table I. Training sets after preprocessing and their corresponding classes

Document_id	***Keywords***	***Class_id***	***Class_ name***
D1	apple, laptop, model	0	computer
D2	sachin, century, partnership	4	sports
D3	mustard, beef, steak	3	food and cooking
D4	hollywood, california	1	entertainment
D5	audi, car, model	2	companies and industries
D6	apple, macbook, air	0	computer
D7	oriental, iron, steak	3	food and cooking
D8	messi, aguero, partnership	4	sports

Table II. All keywords from Training sets

Word_id	***Keywords***
W0	apple
W1	model
W2	partnership
W3	steak
W4	aguero
W5	air
W6	audi
W7	beef
W8	california
W9	car
W10	century
W11	hollywood
W12	iron
W13	laptop
W14	macbook
W15	messi
W16	mustard
W17	oriental
W18	sachin

Table III. Test set after preprocessing

T_id	***Keywords***
T1	apple, macbook, model, computer
T2	dhoni, triple, century, partnership
T3	oriental, wine, steak
T4	oriental, car, model
T5	hollywood, movie

Table IV. Fuzzy RT relation matrix

	W0	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	W15	W16	W17	W18
W0	1	0.333	0	0	0	0.5	0	0	0	0	0	0	0	0.5	0.5	0	0	0	0
W1	0.333	1	0	0	0	0	0.5	0	0	0.5	0	0	0	0.5	0	0	0	0	0
W2	0	0	1	0	0.5	0	0	0	0	0	0.5	0	0	0	0	0.5	0	0	0.5
W3	0	0	0	1	0	0	0	0.5	0	0	0	0	0.5	0	0	0	0.5	0.5	0
W4	0	0	0.5	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0
W5	0.5	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0
W6	0	0.5	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0
W7	0	0	0	0.5	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0
W8	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0
W9	0	0.5	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0
W10	0	0	0.5	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
W11	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0
W12	0	0	0	0.5	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0
W13	0.5	0.5	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
W14	0.5	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0
W15	0	0	0.5	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0
W16	0	0	0	0.5	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0
W17	0	0	0	0.5	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0
W18	0	0	0.5	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1

Table V. Class and their corresponding keywords

Class_id	***Keywords***
Class_0	laptop, air, apple, macbook
Class_1	hollywood, california
Class_2	model, car, audi
Class_3	steak, oriental, beef, iron, mustard
Class_4	aguero, century, messi, partnership, sachin

Table VI. Class membership of Test Documents

T_id	***Class_0***	***Class_1***	***Class_2***	***Class_3***	***Class_4***	***Target class(maximum class membership value)***
T1	0.6666	0	0.3333	0	0	computer
T2	0	0	0	0	0.5	sports
T3	0	0	0	0.6666	0	food and cooking
T4	0.2222	0	0.6666	0.3333	0	companies and industries
T5	0	0.5	0	0	0	entertainment