IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-6, Issue-7, July 2018 E-ISSN: 2347-2693

Relevance Feature Search for Text Mining using FClustering Algorithm

R. R. Kamble [1*], D. V. Kodavade [2]

[1] Dept. of Computer Science and Engineering, D.K.T.E.'s TEI (An Autonomous Institute), Ichalkaranji, India.
[2] Dept. of Computer Science and Engineering, D.K.T.E.'s TEI (An Autonomous Institute), Ichalkaranji, India.

**Corresponding Author: rekhakamble2604@gmail.com*

Available online at: www.ijcseonline.org

Accepted: 20/July/2018, Published: 31/July/2018

***Abstract*—** The huge challenge in discovering relevance feature is to determine the quality of user searched documents. The user wants relevant features to search the text, document, image, etc. approximately. The techniques earlier used where term based and pattern based. Now days clustering methods like partition based, density based and hierarchical is used along with different feature selection method. The term-based approach is extracting terms from the training set for describing relevant features. Partitioned text mining solves the low-level support problem, but it suffers from a large number of noise patterns. Information content in documents is identified by frequent sequential patterns and sequential patterns in the text documents and the useful features for text mining are extracted from this. Extracted terms are classified into three type's positive terms, general terms and negative terms. In order to deploy advanced features on low-level features, this article finds positive and negative patterns in text documents.

***Keywords*—** Text mining, text feature extraction, text classification.

I. INTRODUCTION

In order to find useful information from many digital text documents, text mining techniques are used. Text categorization is designed to improve the quality of text representations and to develop high quality classifiers. By using a text mining model to retrieve information that meets the user's needs, this will be done at valid intervals.

Retrieving as many relevant documents as possible is the goal of traditional information retrieval. Unrelated ones are filtered out at the same time. In order to extract information from the dataset and transform it into an understandable structure, using a data mining process, the data mining process can be described as a pattern in the lookup data and defined as extracting hidden, previously unknown and useful information.

If the quality of the relevance of the discovery is to be ensured, the features in the text document describing the user's preferences are important. This is a very challenging task due to large-scale terminology and data models. In a terminology based approach, each term in a document is associated with a value called a weight.

Most existing text mining and classification methods use a term-based approach used in many previous classification methods and text mining. There are two challenging issues in using pattern mining techniques to find relevant features in related and unrelated documents. The first is the low support issue.

Given a topic, long patterns are usually more specific to the topic, but they usually appear in documents that are supported or less frequently. If the minimum support is reduced, many noisy patterns are found. The second problem is misunderstanding, which means that the results of the measures used in pattern mining are not suitable for using patterns to solve problems.

For example, the high frequency pattern can be a general pattern because it can often be used in related and unrelated documents. Therefore, the difficult question is how to use the discovered patterns to accurately weight useful features. Pattern classification mining (PTM) has been found to be a closed sequential pattern in text documents, where patterns are a set of terms that often appear in paragraphs.

Pattern mining based methods have been used for information filtering because data mining has developed techniques for removing redundancy and noise patterns. Pattern-based methods perform better in describing user preferences than term-based methods. Many of the more weighted terms are more general because they can often be used in related and unrelated documents.

The specificity score of a term is calculated based on its appearance in discovered positive and negative patterns.

The terms are classified into three categories: positive specific terms, general terms and negative specific terms based on their appearance in a training set. The terms frequently used in both relevant documents and irrelevant documents are general terms. The terms that are more frequently used in the relevant documents are classified into the positive specific terms. The terms that are more frequently used in the irrelevant documents are classified into the negative specific terms.

The Relevance Feature Discovery presents an innovative model by classifying terms into different categories and updating term weights and their distribution in patterns efficiently by improving the performance of text mining. The objective of Relevance Feature Discovery is to extract high-quality features that can represent what user needs. As compared to Term based Methods and Pattern based methods this system is better.

The remainder of this paper is organized as follows. In Section 2, we present the literature review about relevance feature discovery. Section 3 explain the methodology, Section 4 is about Result and discussion. Conclusion and Future work are drawn in Section 5.

II. RELATED WORK

G. Salton and C. Buckley [1] proposed a paper named Term Weighting Approaches in Automatic Text Retrieval. The system uses text indexing by appropriately using weighted single items, which facilitates the information retrieval process. A valid term weighting systems are used that can represent documents in the term vector space. The documents are ranked according to the weight of the term. But it suffers from polysemy and synonymy problems. In some literature, terms with higher tf*idf will be meaningless and it is difficult to select a limited number of features among a large number of words.

Y. Li et al. [2] proposed a method of mining ontology for automatically retrieving Web user information. The fundamental objective of this research is the automatic meaning discovery other than pattern discovery. Because it is difficult to automatically obtain web user profiles, it is difficult to get the correct information from a particular web user or a group of users from the web. This paper proposes a new way to solve this problem. It proposes a way to capture evolving patterns. In addition, it identifies the process of assessing relevance. This paper provides theoretical and experimental evaluations for this approach. The problem is that users cannot distinguish between relevant data and irrelevant data. Network users don't know how to represent interesting topics, and use phrase-based methods that have low frequency of occurrence.

Y. Li et al. [3] proposed a method to select negative documents that are closed to the extracted features in positive documents. It proposed the mining and revision algorithms which use twice for positive and negative documents. Features in positive documents are found by revision process in the training set which contains higher level positive patterns and low-level terms. After this top-K negative samples are selected from the training set in compliance with standard rules of the positive features. The feature discovery in the positive document is done using pattern mining technique and which is also used to discover negative patterns and terms from selected negative documents. The Revised weight function is obtained by the process of revised initial features. But the negative pattern can't greatly improve the accuracy is the shortcoming of the system.

N. Zhong et al. [4] proposed an effective pattern discovery technique to overcome the low frequency and misunderstanding problems in text mining. This technique uses pattern deployment and pattern evolution to optimize the patterns found in text documents. A D- pattern mining algorithm is proposed. The training process is described to find the set of D- patterns. Deploying process is focused that consists of term support evolution and D- pattern discovery. D- Pattern is composed by discovering all the positive documents. But how to effectively integrate patterns into related and unrelated documents is a disadvantage of this system.

Z. Zhao et al. [5] proposed Similarity Preserving Feature Selection - Nesterov's method. The proposed SPFS framework improves existing algorithms by overcoming their common disadvantages in dealing with feature redundancy. The feature selection is used to select a subset of the original features, which is done according to the selection criteria for selecting a small group of original features. By considering the original features and feature selection, the learning model is improved. This is a learning process. This is the main goal of the learning model in order to guide the search for relevant features. The disadvantage of this system is that it can identify the optimal feature set without any redundancy.

Li et al. [6] proposed FClustering and WFeature algorithm. The FClustering algorithm describes the process of feature clustering, and after classifying terms using the FClustering algorithm, the WFeature algorithm is used to calculate term weights. A model for relevance feature discovery is presented that finds positive and negative patterns in a text document that are described as higher level features and deployed on low level features. Pattern mining techniques raise two issues. The first is how to deal with low-frequency modes. The second is how to effectively use negative feedback to modify the extracted information filtering features. Another challenging issue in text mining is the long-term model. The proposed system will propose an innovative technique to overcome the above limitations and

problems in order to discover and classify low-level terms based on the appearance in advanced features and their specificity in the training set.

III. METHODOLOGY

Fig. 1. shows system architecture of Text Mining process.

There are three steps which consists of feature discovery and deploying, term classification and term weighting. It first finds positive and negative patterns and terms in the training set and by using the FClustering algorithm it classifies terms into three categories. Finally, term weighting is done by using the WFeature algorithm.

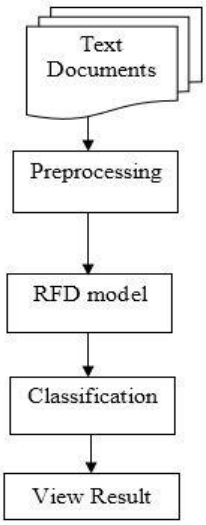

Fig. 1. System architecture of Text Mining

Description of each module:

The system consists of following modules:

1. Pre-processing
2. RFD model
3. Classification

I. Pre-processing

In this module, the preprocessing is applied for text documents. The stop words are removed from the documents and stemming process is used to reduce a word to its root form.

II. Relevance Feature Discovery (RFD)

The RFD model describes the relevant features in relation to three groups: positive specific terms, general terms and negative specific terms based on their appearances in a training set.

Specificity Function:-

In the RFD model, a term's specificity is defined according to its appearance in a given documents. The terms that are more frequently used in the relevant documents are classified into the positive specific category; the terms that are more frequently used in the irrelevant documents are classified into the negative specific category. Terms that are often used in related and unrelated documents are general terms.

Weighting Features:-

The calculation of the original RFD term weighting function includes deploying support(d_sup) and specificity(spe). The d_sup is the number of times a term appears in a document. The specificity measures the importance of terms in all files; we obtain this metric by dividing the total number of documents by the number of documents containing the term and then taking the logarithm of the quotient.

III. Term Classification

Algorithms:

- **FClustering Algorithm**

- It describes the process of feature clustering.

- Initialize the category of the document and perform clustering, i.e. positive specific(pos), negative specific(neg) and general(gen), sorting the cluster according to the specificity of the cluster.

- **WFeature Algorithm**

-- It calculates specificity(spe) and support(sup).

- It uses the FClustering algorithm to classify terms into three categories of positive specific terms, general specific terms and negative specific terms.

- It calculates term weights.

Algorithmic description of each method

1. Preprocessing:

The first module is preprocessing. In this module, the preprocessing is applied to text documents. The stop words are removed from the documents and stemming process is used to reduce a word to its root form.

The Porter Stemmer Algorithm [12]:

This is a simple rule-based stemming algorithm. The algorithm consists of seven sets of rules, applied in order.

Consonant is a letter other than A, E, I, O, U, and Y preceded by consonant.

Vowel is any other letter.
With this definition, all words are of the form:
$(C)(VC)^{m}(V)$
Where, C-string of one or more consonants (con+)
V-string of one or more vowels

The rules are of the form:
(condition) S1 -> S2
Where S1 and S2 are suffixes
m-The measure of the stem
*S-The stem ends with S
v-The stem contains a vowel
*d-The stem ends with a double consonant
*o-The stem ends in CVC (second C not W, X, or Y)
Step 1: removes final -es.
SSES -> SS
IES -> I
SS -> SS
S -> ϵ
Step 2a: gets rid of plurals and -ed or -ing.
(m>1) EED -> EE
(*V*) ED -> ϵ
(*V*) ING -> ϵ
Step 2b:
(These rules are ran if second or third rule in 2a apply)
AT-> ATE
BL -> BLE
(*d & ! (*L or *S or *Z)) -> single letter
(m=1 & *o) -> E
Steps 3 and 4: turns terminal y to i when there is another vowel in the stem.
Step 3: Y Elimination (*V*) Y -> I
Step 4: maps double suffices to single ones. so -ization (= -ize plus -ation) maps to -ize etc. note that the string before the suffix must give m() > 0.
(m>0) ATIONAL -> ATE
(m>0) IZATION -> IZE
(m>0) BILITI -> BLE
Steps 5 and 6:
Step 5: deals with -ic-, -full, -ness etc. similar strategy to step 4.
(m>0) ICATE -> IC
(m>0) FUL -> ϵ
(m>0) NESS -> ϵ
Step 6: takes off -ant, -ence etc., in context <c>vcvc<v>.
(m>0) ANCE -> ϵ
(m>0) ENT -> ϵ
(m>0) IVE -> ϵ
Step 7: removes a final -e if m() > 1.
Step 7a:
(m>1) E -> ϵ
(m=1 & !*o) NESS -> ϵ
Step 7b:
(m>1 & *d & *L) -> single letter

2. FClustering Algorithm [6]:
Algorithm FClustering describes the process of feature clustering, where DP^+ is the set of discovered patterns of D^+ and DP^- is the set of discovered patterns of D^-. First it initialize the three categories. All terms that are not the elements of positive patterns are assigned to category neg. For the remaining m terms, each is viewed as a single cluster in the beginning. It also sorts these clusters in C based on their min_{spe} values. Then it describes the iterative process of merging clusters until there are three clusters left. The merging process first decides the closest of two adjacent clusters, c_k and c_{k+1}. It also merges the two clusters into one, denoted as c_k , and deletes c_{k+1} from C. In the last step, it chooses the first cluster as pos, the second cluster as gen (if it exists) and the last cluster as a part of neg (if it exists).

3. WFeature Algorithm [6]:
Algorithm WFeature is applied to calculate term weights after terms are classified using Algorithm FClustering. It first calculates the sup function and spe function. It also uses Algorithm FClustering to classify the terms into the three categories of pos, gen and neg. Finally, it calculates the weights of terms using the w function.

IV. RESULTS AND DISCUSSION

The system experimented on Reuters-21578 dataset and calculated average precision of the top-20 documents, mean average precision (MAP).

MAP measures the precision at each relevant document first, and then obtains the average precision for all topics.

In this experiment, the two versions of the RFD model are developed. The first version is called RFD_1 which uses two empirical parameters (Q1 and Q2) to group the low-level terms into three groups. This model can achieve the satisfactory performance, but it has to manually decide the two parameters according to their real performance in testing sets. The second model is called RFD_2 which uses the proposed FClustering algorithm to automatically determine the three categories pos, gen and neg based on the training sets.

The RFD_1 and RFD_2 model is compared and the performance of RFD_2 model is approximate better than RFD_1. Table 1 shows the average results of precision and mean average precision on Reuters-21578, where %chg denotes the percentage change of RFD_2 over RFD_1.

As shown in Table 1, RFD_2 can produce the same performance as RFD_1. In addition, a small improvement was observed.

Table1. Comparison Results of RFD_1 and RFD_2 models on Reuters-21578

MODEL	TOP-20	MAP
RFD_1	0.889	0.834
RFD_2	0.895	0.837
%CHG	0.67%	0.35%

The following graph shows Comparison Results of RFD_1 and RFD_2 models on Reuters21578.

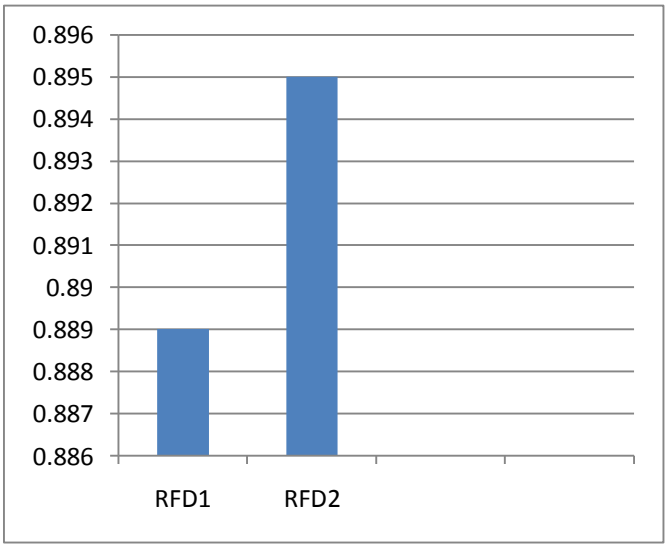

Fig2. Comparison Results of RFD_1 and RFD_2 models on Reuters-21578

Conclusion AND FUTURE SCOPE

The relevance feature discovery describes the relevant features in relation to three groups: positive specific terms, general terms and negative specific terms based on their appearance in a training set. The FClustering algorithm is used to classify the terms into the three categories pos, gen and neg. Finally, it calculates the weights of terms using the w function.

The text mining here is carried on text documents which are classified into a set of relevant and irrelevant documents to optimize the search efficiency. The similar implementation can be carried out on multimedia files. Also from the storage point of view instead of a relational database a non relational database can be used in order to store the documents. Example MongoDB or Cassandra. Hence the future scope includes incorporating similar mechanism for the multimedia files and using a NOSQL database for storage purpose.

REFERENCES

[1] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," in Inf. Process. Manage., vol. 24, no. 5,pp. 513–523, Aug. 1988.

[2] Y. Li and N. Zhong, "Mining ontology for automatically acquiring web user information needs," in IEEE Trans. Knowl. Data Eng., vol. 18, no. 4, pp. 554–568, Apr. 2006.

[3] Y. Li, A. Algarni, and N. Zhong, "Mining positive and negative patterns for relevance feature discovery," in Proc. ACM SIGKDD Knowl. Discovery Data Mining, 2010, pp. 753–762.

[4] N. Zhong, Y. Li, and S.-T. Wu, "Effective pattern discovery for text mining," in IEEE Trans. Knowl. Data Eng., vol. 24, no. 1, pp. 30–44, Jan. 2012.

[5] Z. Zhao, L. Wang, H. Liu, and J. Ye, "On similarity preserving feature selection," in IEEE Trans. Knowl. Data Eng., vol. 25, no. 3, pp. 619–632, Mar. 2013.

[6] Yuefng Li, Abdulmohsen Algarni, Mubarak Albathan, Yan shen, and moch Arif Bijaksana "Relevance feature discovery for text mining" IEEE transaction on knowledge and data engineering,vol.27,no.6, pp.1656-1669, june2015.

[7] N. Azam and J. Yao, "Comparison of term frequency and document frequency based feature selection metrics in text categorization" Expert Syst. Appl., vol. 39, no. 5, pp. 4760–4768,2012.

[8] X. Li and B. Liu, "Learning to classify texts using positive and unlabeled data," in Proc. 18th Int. Joint Conf. Artif. Intell., 2003,pp. 587–592.

[9] Y. Li, A. Algarni, S.-T. Wu, and Y. Xue, "Mining negative relevance feedback for information filtering," in Proc. Web Intell. Intell. Agent Technol., 2009, pp. 606–613.

[10] S. Purandare, "Relevance Feature Discovery In Text Documents", International Journal of Computer Engineering, Vol.3, Issue.6, pp.98-101, 2016.

[11] Sujamol.S, Ariya T K Identifying and Analyzing Efficient Pattern Discovering Techniques for Text Mining ", International Journal of Research in Computer and Communication Technology pp.102-105, 2014.

[12] M.F. Porter , "An algorithm for suffix stripping, Program", 14(3) pp 130−137, 1980.

Authors Profile

Ms. Rekha R. Kamble completed B.E. in Computer Science & Engineering from DKTE Society's Textile &Engineering Institute, Ichalkaranji, India in 2016. She is pursuing Master of Technology in Computer Science & Engineering from DKTE Society's Textile & Engineering Institute, Ichalkaranji (An Autonomous Institute), India.

Dr .D. V. Kodavade, the Head of Department of Computer Science & Engineering, at DKTE Society's Textile & Engineering Institute, Ichalkaranji (An Autonomous Institute),India. He is a member of the ACM, CSI, IEEE Computer Society. His current research interest includes Artificial Intelligence & Knowledge Based Systems, IoT, Neural Networks, Hybrid Intelligence.