International Journal of Advance Research in Science and Engineering
Volume No.07, Special Issue No.03, April 2018
www.ijarse.com
IJARSE
ISSN: 2319-8354

# Introduction to keyword weighting algorithm to show the distinction among keywords

**Dipti D. Mehare[1], Prof. A. V. Deorankar[2]**

*[1]P. G. Scholar, Department of Computer Science,*
*Government college of Engineering, Amravati, Maharashtra, (India)*
*[52]Associate Professor, Department of Computer Science,*
*Government college of Engineering, Amravati, Maharashtra, (India)*

## ABSTRACT

*On-line text documents rapidly increase in size with the growth of World Wide Web. To manage such a huge amount of data, several applications came into existence. These applications such as search engine and topic detection are based on feature extraction. It is extremely time consuming and difficult task to extract keyword and feature manually. So an automated process that extracts keywords or features is established. Which is TF-IDF, Term frequency-Inverse document frequency is widely used to express the documents feature weight. This paper proposes a new keyword weighting method to show distinction among keywords, which are extracted from method TF-IDF. We use the grammatical relations as standards to show the weight of each keyword, and this enables users to retrieve relevant documents from the cloud based on their own interests.*

***Keyword: Cloud computing, Feature extraction, weighting method.***

## I. INTRODUCTION

Cloud computing becomes increasingly popular, more people are outsourced their data to the cloud, due to its flexibility and unlimited resources. Also, it can reduce local data maintenance costs and offer a convenient communication channel to share resources among the data owner and legitimate data users. Hence, a large amount of data, ranging from emails to personal health records etc. is increasingly outsourced to public clouds. For privacy concerns, data owners must encrypt their sensitive data before outsourcing. To enable effective searches over encrypted data, the data owner first builds an encrypted index based on the extracted keywords from data files and the corresponding index-based keyword matching algorithm, and then outsources both the encrypted data and the index structure to the cloud server. To search over the encrypted files, the cloud server integrates the trapdoors of keywords with the index information and finally returns the target files to the data users.

In this paper, we take the relationship among query keywords into consideration and design a keyword weighting algorithm to show the importance of distinction of the keywords. Using the keyword weights, we can accurately and efficiently localize the central keyword that the user is interested in. Since we can choose the

central keyword (not all keywords) of the query to extend, our scheme can greatly reduce the trapdoor generation time. In this way, our scheme makes a good tradeoff between the functionality and the efficiency. We use the grammatical relations
as standards to show the weight of each keyword, and this enables users to retrieve relevant documents from the cloud based on their own interests.

## II. FUNDAMENTALS

Grammatical relation is a connection between two keywords in the grammatical tree. In this paper, we use the standford parser to show the grammatical relations. Stanford parser is a natural language parser that works out the grammatical structure of sentences. When some plaintexts are input, it can output part-of-speech tagged text, phrase structure trees, and grammatical relations. If the distance between the two keywords in the grammatical tree is closer, the relation between them is considered as more important. So, we can use the distance between the two keywords in the grammatical tree to quantify the importance of a grammatical relation. Because the grammatical relation is shared between the two keywords, its importance value should be divided into two keywords.

For each query Q = {w1;w2;……;wt} w1,w2….wt are the keywords, it is independent of other queries. The overall weight of the query is t, where t is the number of the query keywords contained in the query Q. For a keyword w, its weight is p*t, where p is the proportion of the importance of the keyword w.

Thus, the keyword weight of w is

$$KW(w) = \frac{(1+\sum R) \times t}{\sum_{i=1}^{t}(1+\sum R)}.$$

Where R is the preference information contained in grammatical relation A
is R(A)=1/ln(d).

## III. COMPUTATION OF KEYWORD WEIGHT

In this paper, we stanford parser is use to show the grammatical relations. By taking keywords as input, it can output the grammatical relations of the keywords and the syntax parse tree.

Consider the example **“blue trouser with pockets”**

The syntax parse tree for above example is given below,

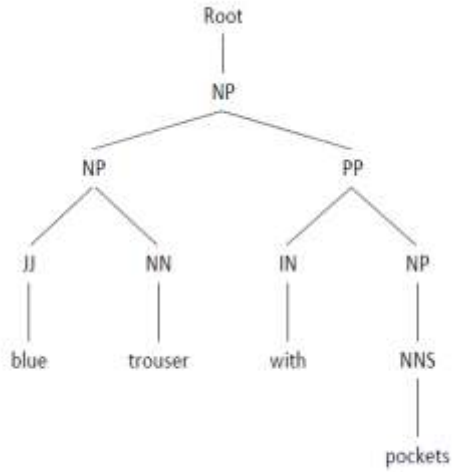


Fig. 2: A parse tree of stanford parser

The grammatical relation for above example is,

amod(trouser - 2; blue - 1)

root(ROOT - 0; trouser - 2)

case(pockets - 4;with - 3)

nmod(trouser - 2; pockets - 4);

where "amod" is the adjectival modifier, "root" is the root of the phrase structure trees, and "nmod" is the noun compound modifier. Fig. 2 shows the parse tree of "blue trouser with pockets", where "NP" is the noun phrase, "NN" is the noun, "JJ" means the adjective or numeral, ordinal and so on.

To evaluate the importance of the relation between two keywords, we use the distance d of two keywords in the parse tree. For example, the distance of "blue" and "trouser" is d = 4, so R(amod) = 1/ln(4) . Since the grammatical relation belongs to the two keywords, we assign R to them based on the following rules: R1 = d-d1/d*R ,R2 = 1-R1 = d-d2/d*I where d1 (resp. d2) is the distance between the first (resp. second) keyword in the grammatical relation and the ancestor of the two keywords. For the relation "root", we insert a random number ro (min(d) <=ro <=max(d)) to the R(root) and R(root) = 1/ln(ro) . Thus, the total importance is

$$4+\frac{1}{ln(4)}+\frac{1}{ln(4)}+\frac{1}{ln(7)}=5.96,$$ the importance of "trouser" is $$1+\frac{1}{ln(4)}+\frac{1}{2ln(4)}+\frac{4}{7ln(7)}=2.38,$$ and the keyword weight of "trouser" is $KW(trouser) = 1.60$ ($KW(blue) = 0.91$, $KW(with) = 0.67$, $KW(pockets) = 0.82$).

**Functionalities of the Keyword Weight:** The keyword weight has the following two functionalities:

1] Enable user to retrieve relevant documents based on his own interest: If a document has a keyword that is given more attention than those in other documents, it should have a higher priority in the returned Fid.

2] Identify the central keyword: Because the weight of each keyword has been calculated, the keyword with a larger weight is the one that a user is really interested in, and thus can be chosen as the central keyword.

**Limitations:**

1] If there exist some spelling mistakes in a query, the keyword weight will not be accurate. So the user needs to ensure that the input keyword is correct. That implies our scheme only does not support fuzzy keyword search which can tolerate input errors.

2] In some cases, the importance of each input keyword may be the same .In that case keyword weighting algorithum is not applicable.

## IV. CONCLUSIONS

In this paper, for the first time, we took the relationship among the query keywords into consideration and designed a keyword weighting algorithm based on the relations. Keyword weighting algorithm to show the importance of distinction of the keywords. Using the keyword weights, we can accurately and efficiently localize the central keyword that the user is interested in. Since we can choose the central keyword (not all keywords) of the query to extend, our scheme can greatly reduce the trapdoor generation time. In this way, our scheme makes a good tradeoff between the functionality and the efficiency.

## REFERENCES

[1]. Zhangjie Fu, Member, IEEE, Xinle Wu, Qian Wang, Member, Kui Ren," Enabling Central Keyword-based Semantic Extension Search over Encrypted Outsourced Data" IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY.

[2]. http://nlp.stanford.edu/software/lex-parser.shtml.

[3]. Christian Borgelt and Andreas N¨urnberger," Experiments in Term Weighting and Keyword Extraction in Document Clustering", Dept. of Knowledge Processing and Language Engineering Otto-von-Guericke-University of Magdeburg Universit¨atsplatz 2, D-39106 Magdeburg, Germany.

[4]. Rakhi Chakraborty," DOMAIN KEYWORD EXTRACTION TECHNIQUE: A NEW WEIGHTING METHOD BASED ON FREQUENCY ANALYSIS", Department of Computer Science & Engineering, Global Institute Of Management and Technology, Nadia, India.