Performance Measurement of Using Jaccard Coefficient for Keywords Similarity

Su Mon Ko[1], Mie Mie Aung[2], Yin Yin Cho[3]
[1,2,3]*Information Technology Support and Maintenance Department, University of Computer Studies (Meiktila), Myanmar*

***Abstract*— Analysis of huge data requires extracting information for decision making. Presently, information retrieval (IR) is one of the retrieval systems which are searching for specific information. This allows users to specify the search criteria as well as specific single keywords, phrases and sentences to obtain the required results. Additionally, an index of search engines has to be updated on most recent information as it is constantly changed over time. Particularly, information retrieval results as documents are typically too extensive, which effect on accessibility of the required results for searchers. Consequently, a similarity measurement between keywords and index terms is essentially performed to facilitate searchers in accessing the required results promptly. Thus, this paper proposed the similarity measurement method between words by deploying Jaccard Coefficient. The performance of this proposed similarity measurement method was accomplished by employing precision and recall based on three types of query user entered.**

***Keywords*— Information Retrieval, Vector Space Model, Jaccard Coefficient Similarity**

I. INTRODUCTION

Keyword search is the simplest form of the most popular query method for search engine in information systems. It contains a single keyword or multiple keywords and a sort phrase. In a single keyword search, a particular word in the document will be displayed such as in a case of searching for sugar-producing crops. Keywords are specific words that can be sugar cane or we can query with the keyword in other forms to allow users to easily find the needed information quickly. The first significant issue that needs to consider is the technique used to measure the similarity between a user-specified key and the index finger to indicate directly to the required information.

II. RELATED WORKS

There are several studies that used Vector Space Model in information retrieval system to optimize the user query. E men Al Mashagba et al described various different similarity measures like dice, cosine, Jaccard etc in vector space model and compare each similarity measures using genetic algorithms approach based on different fitness functions, different mutations and different crossover to find the best solution of the given query. Gokul Patil and Amit Patil described web based text mining problem and step to solve that problem and filter out just those that have the desired meaning. J.Allaan, Jay Aslam et al. described various area of information retrieval system and also describes major challenges within each of those areas.

III. THEORETICAL BACKGROUNG

Text mining has become an important research area. It is the discovery by computer of new, previously unknown information, by automatically extracting information from different written resources. Text mining is similar to data mining, except that data mining tools are designed to handle structured data from databases, but text mining can work with unstructured or semi-structured data sets such as emails, full-text documents and HTML files etc. Starting with a collection of documents, a text mining tool would retrieve a particular document and pre-process it by checking format and character sets. Text mining is the data analysis of text resources so that new, previously unknown knowledge is discovered. It is an interdisciplinary field that borrows techniques from the general field of data mining and it, additionally, combines methodologies from various other areas such as information extraction (IE), information retrieval (IR), computational linguistics, categorization, topic tracking and concept linkage.

A. Information Retrieval (IR) System

Information retrieval (IR) is the study of helping users to find information that matches their information needs. IR studies the acquisition, organization, storage, retrieval, and distribution of information. An information retrieval system that is a software program that stores and manages information on documents, often textual documents but possibly multimedia. The system assists users in finding the information they need. It does not explicitly return information or answer questions. A perfect retrieval system would retrieve only the relevant documents and no irrelevant documents. Perfect retrieval systems do not exist and will not exist, because search statements are necessarily incomplete and relevance depends on the subjective opinion of the user.

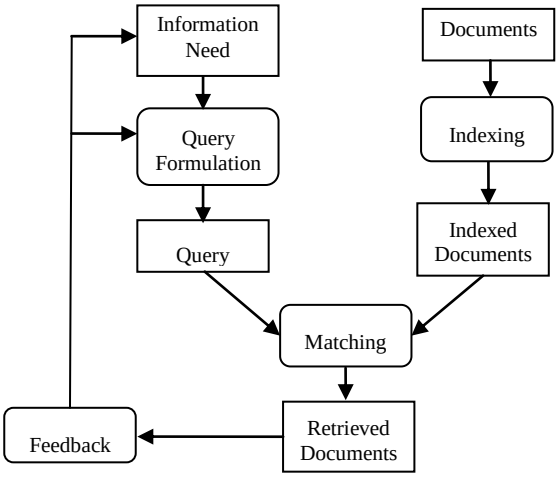

Fig 1. Information Retrieval Process

Information retrieval processes are shown in Fig 1. There are three basic processes an information retrieval system has to support: the representation of the content of the documents, the representation of the user's information need, and the comparison of the two representations. Representing the documents is usually called the indexing process. The process of representing user information need is often referred to as the query formulation process. The resulting representation is the query. The comparison of the query against the document representations is called the matching process. The matching process usually results in a ranked list of documents.

*1) Text Preprocessing: B*efore the documents in a collection are used for retrieval, some preprocessing tasks are usually performed. For traditional text documents (no HTML tags), the tasks are stopword removal, stemming, and handling of digits, hyphens, punctuations, and cases of letters. For Web pages, additional tasks such as HTML tag removal and identification of main content blocks also require careful considerations.

2) Tokenization: Tokenization is the process of breaking a stream of text up into words, phrases, symbols, or other meaningful elements called tokens. The list of tokens becomes input for further processing such as parsing or text mining. Given a character sequence and a defined document unit, tokenization is the task of chopping it up into pieces, called tokens, perhaps at the same time throwing away certain characters, such as punctuation. Example of tokenization is shown in Fig 2.

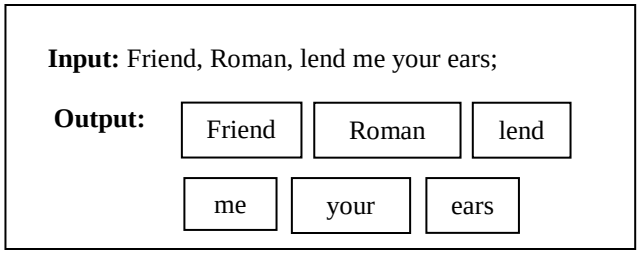

Fig 2. Example of Tokenization

3) Stopword Removal: Stop words, or stoplists are list of words that are filtered out prior to or after processing of text relying on their level of usefulness in a given context. Stopwords removal process improves information retrieval and searching by ignoring words that usually appear in every document and thus not helpful in distinguishing documents from each other. Additionally, the removal of stopwords reduces the index size (number of distinct words in the index) and therefore save space and time. Stopwords are frequently occurring and insignificant words in a language that help construct sentences but do not represent any content of the documents. Articles, prepositions and conjunctions and some pronouns are natural candidates. Common stopwords in English include: a, about, an, are, as, at, be, by, for, from, how, in, is, of, on, or, that, the, these, this, to, was, what, when, where, who, will, with. Such words should be removed before documents are indexed and stored. Stopwords in the query are also removed before retrieval is performed.

IV. Information Retrieval (IR) Models

An IR model governs how a document and a query are represented and how the relevance of a document to a user query is defined. There are four main IR models: Boolean model, vector space model, language model and probabilistic model.

A. Vector Space Model

A document vector captures the relative importance of the terms in a document. The representation of a set of documents as vectors in a common vector space is known as vector space model and is fundamental to a host of information retrieval operations ranging from scoring documents (d) on a query (q), document classification and clustering. A document is then represented as a vector of term weights. The number of dimensions in the vector space is equal to the number of terms used in the overall documents collection. The weight of a term in a document is calculated using a function of the form tf-idf, where tf (term frequency weight) is a function of the number of occurrences of the term within the document and idf (inverse document frequency weight) is an inverse function of the total number of documents that contain the term. A query in the vector space model is treated as if it were just another document allowing the same vector representation to be used for the queries as for documents. This representation naturally leads to the use of the vector inner product as the measure of similarity between the query and a document. This measure is typically normalized for vector length, such that the similarity is equal to the cosine of the angle between the two vectors. When document vectors reflect the frequencies with which terms appear, documents are considered similar if their term vectors are close together in the vector space. After all of the documents in the collection have been compared to the query, the system sorts the document by decreasing similarity measure and returns a ranked listing of documents as the result

of the query. Query and document representations in the vector space model are shown in Fig 3.

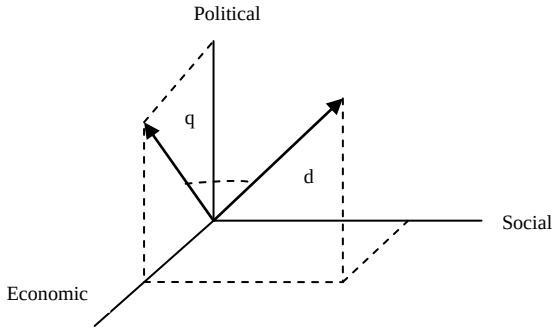

Fig 3. Query and Document Representation in the Vector Space Model

B. Jaccard Coefficient Similarity Method

Jaccard coefficient similarity method is one type of vector space model for information searching. Jaccard coefficient similarity is a parameter used to compare characteristic similarity between sets of information. This method can be applied in analyzing meaning of a word by deducing the word as a set of characters. This relates to the study of in which word searching procedure is created by matching a word that a user requires to words contained in the index. If the word input matches to the ones in the index, the input word will be the main in the search. However, if the word input doesn't match to the one in the index, the input word will be computed through keyword similarity measurement by Jaccard similarity function. The result is quite acceptable. In this system, the Jaccard coefficient similarity method is used to measure the similarity between the user query and each document.

$$sim(d_j, q) = \frac{\sum_{i=1}^{|v|} w_{ij} \times w_{iq}}{\sum_{i=1}^{|v|} (w_{ij})^2 + \sum_{i=1}^{|v|} (w_{iq})^2 - \left(\sum_{i=1}^{|v|} w_{ij} \times w_{iq} \right)} \quad (1)$$

Equation (1) describes the similarity measurement between document (*j*) and query (*q*). The number of vector in each document represents as (*v*) and the weight value of each vector in document (*j*) is w_{ij}. The weight value of each vector in query (*q*) is representing as w_{iq}.

1) Weight of the Term within Document: The weight equation for the term within document is as follows:

$$w_{ij} = tf_{ij} \times idf_i \quad (2)$$

$$tf_{ij} = \frac{f_{ij}}{\max\{f_{1j}, f_{2j}, ..., f_{|v|j}\}} \quad (3)$$

$$idf_i = \log \frac{N}{df_i} \quad (4)$$

Equation (2) describes the weight values of each term in documents. The weight of term (*i*) in document (*j*) is represented as w_{ij}. Equation (3) and (4) shows the term frequency and inverse document frequency of each term in documents to calculate the term weights.

2) Weight of the Term within Query: The weight equation for the term within query is as follows:

$$w_{iq} = \left[0.5 + \frac{0.5 f_{iq}}{\max\{f_{1q}, f_{2q}, ..., f_{|v|q}\}} \right] \times \log \frac{N}{df_i} \quad (5)$$

Equation (5) describes the weight values of each term in query. The weight of term (*i*) in query (*q*) is represented as w_{iq}.

IV. IMPLEMENTATION

A. The data relationship between the information

This research paper was classified into two parts: 1) the information prepared as text file documents which were contained the information user required and 2) the query that was tested in three groups (keywords, sentences and paragraphs) by user. These queries are also determined by the researchers. The information were taken for the history data of Myanmar Bagan Dynasty.

B. The calculation of search words to identify similarity

There are two steps that are pre-processing step and similarity calculation step. Stopwords are removed from both of the user queries and each document in pre-processing step. Stopwords are "on", "in", "of" and so on. In the similarity calculation step, Jaccard coefficient similarity method are used to calculate the similarity between the user query and each document. Before calculation similarity, the term frequency (TF) and inverse document frequency (IDF) are firstly calculated. After calculating similarity between user query and each document, Ranked similarity results are produced.

C. Performance Evaluation

There are used two performance measure methods. These are precision and recall. Precision is the ratio of the number of relevant records retrieved to the total number of irrelevant and relevant records retrieved. Recall is the ratio of the number of relevant records retrieved to the total number of relevant records in the database. The equations for precision and recall methods are as follows:

$$\text{Precision} = \frac{A}{A+C}$$

$$\text{Recall} = \frac{A}{A+B}$$

Where,

A = Number of relevant records retrieved.
B = Number of relevant records not retrieved.
C = Number of irrelevant records retrieved.

V. RESULTS

Table I presented the precision results of the accuracy testing of Jaccard similarity coefficient on three query types. Q1, Q2 and Q3 represent the queries which entered by keyword, sentence and paragraph. For Q1, the number of missing documents retrieved by the system becomes narrow. Therefore, the precision values of those queries are high when threadshold value is not set. For Q2 and Q3, the number of missing documents retrieved by the system is more than that of Q1 when threadshold value is not set. Therefore, the precision values of Q2 and Q3 are also lower than Q1. As soon as the threadshold value is increase, the precision results for three query types will also increase. Generally, threadshold value 0.3 produces the best accuracy result for this system.

TABLE I. THE PRECISION RESULTS OF THREE QUERY TYPES

	Threadshold Value (0)	Threadshold value (0.1)	Threadshold value (0.2)	Threadshold value (0.3)
Q1	38%	71.10%	76.60%	84.80%
Q2	19.10%	53.20%	67.40%	85.10%
Q3	11.40%	53.90%	87.90%	91.90%

Table II presented the recall results of the accuracy testing of Jaccard similarity coefficient on three query types. Q1, Q2 and Q3 represent the queries which entered by keyword, sentence and paragraph. For Q1, Q2 and Q3, there are no documents which are relevant to the query but the system cannot retrieve. Therefore, the recall values of those queries are 100% when threadshold value is not set. The number of documents which are relevant to the query but the system cannot retrieve are increased when threadshold value of the system is increased. Therefore, the recall values of Q1, Q2 and Q3 are slightly decreased. As soon as the threadshold value is increase, the recall results for three query types will slightly decrease.

TABLE II. THE RECALL RESULTS OF THREE QUERY TYPES

	Threadshold Value (0)	Threadshold value (0.1)	Threadshold value (0.2)	Threadshold value (0.3)
Q1	100%	80.90%	64.70%	53.90%
Q2	100%	60.40%	41.20%	37.40%
Q3	100%	45.60%	39.10%	36.10%

VI. CONCLUSION

This paper gives a brief overview of a Vector Space Information Retrieval model, VSM, with the TF/IDF weighting scheme and the Jaccard similarity measures. It produces all documents which are relevant to the user query when threadshold value is not set. Therefore, the threadshold value must be between 0 and 1. The result can produce the accurate information. The testing process is performed depend on three query types (such as keywords, sentences and paragraphs). According to the testing process, threadshold value 0.3 produces the best accuracy result. Average relevance of document can be increased by applying other methods. In this paper Jaccard Similarity Function is applied but this work can also be done by applying other similarity measure and compare the result with each other. In this work weighted vector is applied to retrieve the information of research papers but it can also be done with binary vector.

REFERENCES

[1]. Anonymous: the free encyclopaedia, Tokenization, June 17, 2014. http://en.wikipedia.org/wiki/Tokenization.
[2]. E man Al Mashagba , Feras Al Mashagba and Mohammad Othman Nassar, "Query optimization using genetic algorithm in the vector space model", *International Journal of Computer Science*, ISSN 0814-1694, vol. 8, no. 3, pp. 450-457, Sept. 2011.
[3]. G. Salton, A. Wong, C. S. Yang, "A Vector Space Model for Automatic Indexing", Communications of the ACM, vol.18, no.11, pp.613-620, 2014.
[4]. G. Vishal and S. L. Gurpreet, "A Survey of Text Mining Techniques and Applications", Journal of Emerging Technologies in Web Intelligence, vol. 1, no. 1, pp. 60-76, August, 2009.
[5]. Gokul Patil, Amit Patil, "Web information extraction and classification using vector space model algorithm", International Journal of Emerging Technology and Advanced Engineering, ISSN 2250-2459, vol. 1, no. 2, pp. 70-73, Dec. 2011.
[6]. J.Allaan, Jay Aslam et al. "Challenges in Information Retrieval and Language Modeling " , Report of a Workshop held at the Center for Intelligent Information Retrieval, University of Massachusetts Amherst, September 2002.
[7]. K. M. Risvik, "Scaling Internet Search Engines: Methods and Analysis", Department of Computer and Information Science, Norwegian University of Science and Technology, 2003.
[8]. M. W. Berry, "Automatic Discovery of Similar Words", in "Survey of Text Mining: Clustering, Classification and Retrieval", Springer Verlag, New York, pp. 24-43, 2004.
[9]. J.Allaan, Jay Aslam et al. "Challenges in Information Retrieval and Language Modeling " , Report of a Workshop held at the Center for Intelligent Information Retrieval, University of Massachusetts Amherst, September 2002.