2015

A Boolean based Question Answering System

Jiayi Wu

A Boolean based Question Answering System

By

Jiayi Wu

A Thesis
Submitted to the Faculty of Graduate Studies
through the School of **Computer Science**
in Partial Fulfillment of the Requirements for
the Degree of **Master of Science**
at the University of Windsor

Windsor, Ontario, Canada

2015

A Boolean based Question Answering System

by

Jiayi Wu

APPROVED BY:

_____

Dr. G. Lan
Odette School of Business

_____

Dr. D. Wu
School of Computer Science

_____

Dr. J. Morrissey, Advisor
School of Computer Science

January 23, 2015

DECLARATION OF ORIGINALITY

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

ABSTRACT

The search engine searches the information according to the key words and provides users with related links, which need users to review and find the direct information among a large number of webpages. To avoid this drawback and improve the search results from search engine, we implemented a Boolean based Question Answering System. This system used Boolean Retrieval Model to analyze and match the text information from corresponding webpages in the document indexing step when users ask a Boolean expression based question. To evaluate system and analyze Boolean Retrieval Model, we used the data set from TREC (Text Retrieval Conference) to finish our experiment. Different Boolean operators in the questions such as AND, OR has been evaluated separately which is clear to analyze the effectiveness for each of them. We also evaluate the overall performance for this system.

DEDICATION

I would like to dedicate the thesis to my parents and my family members who always support and encourage me.

ACKNOWLEDGEMENTS

My greatest gratitude goes to my supervisor Dr. Joan Morrissey. Thanks for her support, beneficial suggestions and encouragement. I am so glad to be her student in the past two year and it is one of most wonderful experiences in my life.

My sincere appreciation also goes to my thesis committee: Dr. Dan Wu and Dr. George Lan for their suggestions about my thesis.

Finally, I would like to give my thanks to who helped me. Thanks for helping me out of difficulties and supporting without a word of complaint.

TABLE OF CONTENTS

LIST OF APPENDICES

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Question Answering System and Boolean Retrieval Model

1.1 The development of Question Answering System

In recent years, the continuous research and development of the Internet boosts the need of network information resources. The network already became the information and data exchange platform. It is not easy for users to find useful information in such a large database. Thus, search engine was designed as the important tool for people to find the information they want. Using search engines to find information is the common method for most of the people in the world. The search engine works based on the key words provided by users and it returns millions of related webpages back to users (Gulli, A., and Signorini, A [2004]). However, among those webpages, some of the information is not useful, so users will waste their time reading lots of unrelated information and they need enough time to find out the direct information. There are many popular search engines being used today such as Google, Yahoo and Bing.

The traditional search engine exists some disadvantages. First of all, it returns back too much related information. Secondly, search engines still focus on key word retrieval which is the base step of understanding and analyzing human language. It did not apply any method to solve the semantic problem of human language, so it will not promote the improvement of the results being retrieved. The last but not least, it cannot provide users a convenient, easy way to find the accurate information.

According to these reasons, it is hard for users to find the direct and correct information quickly among huge network resources by only using the search engine.

Question Answering System was designed to solve this problem. It was considered as the advanced information retrieval system. Instead of only sending back related webpages, question answering system will give the accurate answer to the user. For instance, suppose users want to know what is TCP/IP. The question answering system will return the answers like: "(pronounced as separate letters) Short for Transmission Control Protocol/Internet Protocol, TCP/IP is the suite of communications protocols used to connect hosts on the Internet. TCP/IP uses several protocols, the two main ones being TCP and IP. TCP/IP is built into the UNIX operating system and is used by the Internet, making it the de facto standard for transmitting data over networks. Even network operating systems that have their own protocols, such as Netware, also support TCP/IP."

The rise of the research topic on question answering system was promoted by the need of achieving information in a quickly way. Recently, question answering system has attracted increasingly attention from the researchers in information retrieval and Natural Language Processing research field (Allan et al. [2002]). It can provide accurate and concise answers to the questions asked in natural language. The first Question Answering system is BASEBALL (Green, Wolf and Chomsky [1986]) that is used to answer questions about US League Baseball and the system can satisfy most

of the users at that time. It has a high structured database and the database stores the most common questions which were asked by users before. Although the traditional question answering system can provide users direct answers, the data source is still based on fixed document sets and it cannot satisfy all kinds of need from the users. However, there are abundant information resources on the Internet and it provides better data sets which are required by question answering system. Thus, the combination of the question answering system and the Internet is necessary. It promotes development on the web-based question answering system and this new type of question answering system came out. The brand new question answering system combines the techniques in different research areas that include Computational Linguistics, Information Science and artificial intelligence. More specific, techniques will be used in the system such as Natural Language Parsing, Question Classification, Named Entity Recognition etc. It can solve the questions asked in natural language instead of a list of key words. After searching and processing, it will provide users direct answers instead of related webpages. The first web-based question answering system is called START. It was created in MIT computer science department in 1993. Compared with the search engine, question answering system will save time for users searching information and satisfy users with the correct and high quality answers which they need.

1.2 The important steps of Question Answering System

The question answering system is complexly implemented, a typical question

answering system basically includes three parts: question analysis, information retrieval and answer extraction. In the question analysis step, question answering system will analysis the question being asked by users and achieve the significant part of it, then the system will prepare for the key words being used later. In the information retrieval step, question answering system always uses the popular the search engine such as Google to collect the resource. Apart from that, some researchers also develop their own techniques to achieve the specific information resource from the Internet. In the answer extraction step, question answer system will extract the text information from those webpages and organized those pieces information, then prepare the correct and concise answers for users. (Richardson, R. and Smeaton, A. F. [1995])

Specifically, in the question analysis step, there are three important parts that should be considered: question classification, key words extraction, and question expansion. Among those parts, key words extraction will use the techniques in information retrieval research area. There are three steps in this part: remove unimportant words, stemming and conflation. Also, different regulations and algorithms will be used to process this step efficiently. There are many different algorithms used in this step because different languages have different structures. In the stemming step, there are many famous algorithms used to process English such as Porter Stemming (Porter [1980]) and Lovins stemmer (Lovins [1968]). After stemming, the longest possible ending will be removed, in another word, the suffixes is stripped. Therefore, some

words that have same stems will not be used at the same time which means term used for index will decrease, so that it improves the retrieval efficiency. Some of the researchers in Computational Linguistics research area also mentioned Lemmatisation in this step that is used to return the words into their original format. But it not necessary used in this step, because the stemming step can almost solve the problems. For the other languages there are many algorithms being used which were built up by both language scientists and computer scientists. The question expansion is also a hot research topic in the question analysis step. It was proposed in 1970s and three methods were used on this topic: Global Analysis, Local Analysis and Local Context Analysis. Among those methods, Global Analysis will evaluate the relevance of the terms in the documents and realize the extension of the questions that are provided by users. The techniques such as keywords clustering, basic semantic similarity dictionary and snippets will be used in question extension step. After this step is applied in the question answering system, it returns better results, achieve good feedback from the users. However, these steps are impracticable when the applied system is too large.

The question answering system gains a promising expectation and it will play a significant role. The users of questions answering system are random, the users may only ask some simple questions. Some of the users may be a customer who wants to find the price and the features of a product. Some of the users may be a researcher who wants to collect the marketing, financial or commercial information. Some of the

users may be a technician who wants to find some technique related information that are very specific. Therefore, the answers are wide-ranging. The research has a spacious future development because of the different and numerous users.

1.3 The classification of Question Answering System

According to knowledge area and the methods used to process the question the as well as answers, the question answering system can simply classify as two types: close domain question answering system and open domain question answering system.

In the close domain question answering system, the system needs to answer the questions in a specific knowledge area such as: law, medical or mathematics, or it may answer some question based on a particular fact such as World Cup or The Guinness book of records. So, the close domain question answering system can be divided into those two classes. The close domain question answering system will firstly generate a best match according to the questions which were asked by users, the objects for the best match are provided by the default question set and those default questions already had existing answers. If the best match exists, the system will provide a correct answer. Specifically, to answer the questions under a particular domain, the system will detect with the specialized knowledge and then collect the information passages from different kinds of references. Finally, the combination of those passages will organize as the answers. The close domain question answering system has numerous of users and they need to use the system to find the information, which they want. The close

domain question answering system performs well in the specific knowledge area, because it is easy to forecast and achieve the answers according to the statistics method applied in the system. Also, it is easy to be implemented because the natural language system can quickly solve the question under a particular knowledge area.

For the open domain question answering system, it can almost all kinds of questions. In order to solve different types of question, open domain question answering system will use some syntax and semantic methods in natural language research area to process steps and find the direct answering from the web based document sets.

The difficulty for the implementation of open domain question answering system is dealing with the large amount and different types of questions. The questions may be in details such as the question collections were used by TREC and the questions may also refer to some complexity facts. Because the wide range of the question types, only giving classification towards the question is not enough. The retrieval results may be different for the same question and the degree of difficulty is different in the answer extraction step because the language structure is different in those documents, which are used to prepare answers. For these reasons above, the classification is not only applied in the question analysis step and answer extraction step but also has to be applied to the whole question answering system classify. The open domain question answering system can be classified as five simple types in details. For the first type, the question answering system that is focus on solving the fact based questions. This

system can extract passages from the document sets. It always retrieves the terms in the query one by one and then achieves the answers from the documents directly. For the second type, the question answering system that is focus on solving the question contains basic inference format. This system needs to retrieve back the answers in different documents' passages and then find the relationships between those answers according to basic inference format. Finally, combining those answers together. Under this procedure, the system need to applied techniques in Ontology and Pragmatics research area and the inference in answer extraction step will use these. Because the simple explanation is not enough, the inference will use the world knowledge and common sense. The third type is the question answering system that can combine and achieve the answers from different document sets. The character of this system is that it can collect parts of information in different documents and then organized as the answers. The format of the answers will decide the complexity of the question answering system. The fourth type of question answering system has the analogic reasoning ability. The answers prepared by this system will not definitely show in those documents but need to do analogy analysis with different answers and then distinguish the same points as well as the differences. When the system processes the analogy analysis step, it needs to divide the questions into small, which are used to collect passages of the answers. Finally, it will prepare the answers used analogism. The fifth type is the interactive question answering system. The questions provided by users are based on the interaction of computer and users but not only asked by the users.

1.4 Introduction of some well-developed Question Answering Systems

It has not been a long time since the Question Answering System was created. However, many researchers have designed some well-developed Question Answering Systems.

In 1993, the Question Answering System named START has been created in the Artificial Intelligence laboratory in MIT. This system is the first web-based Question Answering System. START can provide the accurate information to users and it can answer millions of English questions that include geographical questions, film related questions, celebrities' questions and some definition questions. This system can be seemed as a combination system because it still remains two knowledge libraries: "START KB" and "Internet Public Library". Thus, the system can use the data from these two libraries to find the candidate answers when users ask related questions. Otherwise, START will firstly analyze the questions, then find the candidate answer through the search engine and return the direct answers back to users. For example, when user asks: "Who was Bill Gates?", the system will return back: "Cofounder, Microsoft. Born William H. Gates on October 28, 1955, Seattle, Washington." The system will also return the link which is the original source contains the answer. If users want to find more information about the questions, they can simply click the links and view the information on the webpage.

University of Washington created a Question Answering System named MULDER. It

is the first automatic Question Answering System worked base on Internet. This system does not apply any knowledge library and it searches the answers only used the information from Internet. For each question, MULDER will send back a list of candidate answers instead of only one answer. The system will use statistic method to add weights for each answer and the weight is called “confidence level”. For example, when user asks: “Who was the first American in space?”, MULDER will return a list of answers. Among those answers, the answer “Alan Shepard” has 70% confidence level and “John Glenn” has 15% confidence level. At the same time, the system will show the summary of content for those answers and the links as well.

AskJeeves is a famous commercial Question Answering System. This system can answer the questions asked in natural language and return back the answers in paragraphs. There are also some related details of contents and links showing after the answers. AskJeeves also support multimedia format answers. For example, when user asks: “Who was Bill Gates?”, the system not only send back answers but also a photo of Bill Gates. As a commercial Question Answering System, AskJeeves supports different kinds of search such as achieve the information from the pictures and news. The questions processing step of AskJeeves works based on manual operation. AskJeeves has hundreds of employees who are focus on analyze the questions provides by users, then, set up the templates of the questions and save some common questions into the system. Although the templates of questions can be structured in details and it shows clear about the users requirements, the involvement is huge in

both create procedure and maintain procedure.

University of Michigan developed a Question Answering System called AnswerBus. This system is a Multilanguage Question Answering System, which can answer the question in different language such as Spanish, German, Portuguese, English and Italian. For each question, AnswerBus will return back five links and give back the possible answers in XML and TXT format. The Encarta created by Microsoft is an online cyclopedia search engine and it also support with different languages.

LAMP is a Question Answering System created in Singapore. It provides a list of question types that can be selected by users such as person, organization, location, date, time, money and percent. This system only returns back the answers and does not provides the related links.

Webclopedia created in University of Southern California and the Language Computer system created by a company in USA have an excellent performance and outstanding evaluation results in the competition of TREC.

1.5 The future works in Question Answering Systems research area

The research of Question Answering System has made lots of achievements by the involvement of lots of researchers. However, there still exists some problem that need the researchers to resolve. Most of the Question Answering System is a small

application system and the range of solutions provides by these system is limited. Some of the systems even need manual operation. From the current research results, the Question Answering System can be improved in the following aspects.

Firstly, the improvement in the question processing step. This step should involve more techniques in Natural Language Processing to solve the syntax and semantic problems have generated. Secondly, the answer extraction technique have to be completed which aim to extract the important information and provide users with accuracy answers. Thirdly, the typing method for the question could be different. The question could be inputted with voice or graph.

1.6 Information Retrieval Model review

In recent years, with the changing of information and knowledge environment, the information retrieval raises higher requirements and the study of information retrieval model has also been ongoing.

The Boolean model represents the query and document in a simple and understandable way, their similarity based on whether or not they are meeting the Boolean expression. Vector space model represents the document and the query which in the form of vector, estimating the similarity between the query and document by computing the similarity of the vector, and document set results have to be sorted by similarity after

they returned by querying. Probability model is based on the principle of probability sort, which takes into account the intrinsic link between the entry and documentation, and conducting information retrieval statistics based on the probability of similarity between the entry and documentation. Whether classic Boolean model, vector space model, probabilistic model, or the language model which emerged with the changing of environment and technological developments, ontology-based information retrieval model, they all enriched the content of the information retrieval model at different levels (Jin, Hauptmann and Zhai [2002]).

1.7 Boolean model

Boolean retrieval mode1 based on Boolean expression and whether the similarity of the query and document meets the Boolean expression. The Boolean expression should be the queries connected by Boolean operator such as AND, OR. Thus, if a set document collection D = (D_1, D_2, D_3, ⋯, D_i) and the terms in a specific document D_i are T_i = (T_1 , T_2 , $T_3, \dots, T_n$). When the users give a question as Q =T'_1 AND T'_2 AND ... AND T'_i and all of those terms belongs to the T_i, the document D_i should be retrieved back as a result. When the users give a question as Q =T'_1 OR T'_2 OR ... OR T'_i and one of those terms belongs to T_i, the document D_i should be retrieved back as a result (Wang [1989]).

However, due to the results of the Boolean model disorder and the similarity of the document cannot be determined, in recent years, the research of Boolean model is less,

the existing research is mainly manifested in the improvement of the Boolean model, or further optimizing the extended Boolean model (Akutsu, Miyano and Kuhara [1999]).

1.7.1 Evaluation of traditional Boolean query

Patro and Malhotra estimated the success from characteristics of the Boolean web search query (Patro and Malhotra [2005]). The authors state that whether the query will successfully satisfy the users' requirements are unknown. Similarly, whether the Boolean query is efficient or not in finding the location of the best documents is unknown. The authors state that this paper relates characteristics of the search results which are meet the user's requirement. A program which compares the performance of humans and the search queries synthesized performed better than humans and gives an objective judgment of the search queries. The author states that using the precision and recall to measure the quality of queries is the common method. When search query, using the number of relevant documents first returned 20 links as the precision measurement. The authors uses the same range as precision to identify recall. The authors collect data by using volunteers. Volunteers are students who have different topics and some sample relevant documents. After they use the same search engine to search their topic, they revised their query. Then compare the volunteer's query and the synthesized query. The authors use figures to show the relationship between recall and precision of the volunteer queries; the relationship between recall and precision after the queries' quality changed; the precision as function of terms in query and

number of attempts to improve query and relationship between number of terms and precision. A good query returns higher values for precision and good recall which can be used as evaluation results of the method applied to the Boolean model. A good predictor of a successful Boolean web search query is four terms and above average recall. The authors claim that they found the characteristics of good queries that may give good sample of the query successfully achieved by users and their requirements.

1.7.2 Fuzzy set method

Bookstein proposes the fuzzy request which is an approach to weight Boolean search query (Bookstein [1980]). The author states that using Boolean expressions permits one to represent accurately the logical relationships among concepts involved in an information need, but it has some loss in flexibility. When a user is able to express a concept in a Boolean expression and its logical relationship to other concept, the user is not able to express how important that concept is to him relative to the other concepts represented in the query. The same situation will occur in documents indexing. Therefore, it is more desirable to have an approach in which one provides boolean queries with independently assigning weights to each term in the query to indicate how important that term is. This paper refers to previous work by Zadeh (Zadeh [1965]). Zadeh has worked on developing the concept of a fuzzy set to satisfy the need for a set that permits partial membership. However, it is hard to determine whether a given document should be indexed by a specific term sometime and thereby be included in the set. So, for fuzzy sets only indicate the extent to which it is in a set.

The purpose of this paper is to propose a method for resulting information system merges some of the Boolean and weighted systems being accomplished by relying on a generalization of the traditional algebra of sets and by defining a weighting scheme for requests that is consistent with this algebra. In this paper, the author gives a general idea of fuzzy set which is a new extended expression and the manipulations on fuzzy sets can be defined in terms of the membership functions: inclusion, union, intersection and complementation. Then, the author analyses how the fuzzy set works with the queries in which terms are weighted. The specific queries being analyzed include four forms of Boolean expressions: the single index terms, the terms are connecting by AND, the terms are connecting by OR and the terms are connecting by NOT. The author states that allow transforming queries into more convenient forms is one of characteristics of Boolean retrieval systems. In this paper, the author gives some rules that permit one to change a fuzzy query into a different but equivalent one and some relationships follow immediately from the properties of fuzzy sets in general. The rules include: commutativity, associativity, distributivity, duality, idempotency, weight distributivity, involution and weight manipulation. Bookstein gives some example graphs of analysis of the four forms of Boolean expression and use tables simply show the analysis results of them. Bookstein states that the weighted assignment of index terms can be modeled in terms of fuzzy sets so as to accomplish this goal in all four different Boolean expressions. Bookstein claims that it is possible to assign weights to the new expression when it can be transformed into an equivalent Boolean expression.

Kraft and Buell applied fuzzy sets into the generalized Boolean retrieval systems. The authors state that the problem for traditional Boolean retrieval model can be seen as decision theory problem (Kraft and Buell [1983]). The basic assumptions of the model determine documents as relevant or totally non-relevant. It is not possible to consider a document partially relevant and this situation may miss some important relevant documents. According to this problem, there are some new approaches has been generalized to allow for weights to be attached to individual terms, in either the document indexing or the query representation, or both. This paper refers to previous work by Salton and Wu (Salton and Wu [1980]). Salton has worked on vector model which includes the weight on index terms but queries are not Boolean and are usually discrete. This method does not allow for Boolean logic in the query structure that is lacking in general. In this model, differences are precisely what the fuzzy threshold approach emphasizes, while still allowing for Boolean logic in the query. The authors state that the concept of a fuzzy subset can be applied to the document retrieval situation. The membership function can describe terms with a weight in the interval [0, 1]. The traditional Boolean indexing has the weights with zero or one. If weights are in the open interval (0, 1), then, the fuzzy indexing will be used. The probabilistic approach used in the fuzzy retrieval is concerned with estimating the probability of relevance of a given document to a query and it preserves the Boolean lattice properties. The fuzzy subset retrieval allow user to generalize from single-term queries. So, Boolean connectives are similar with the vector lattice. The generalization of traditional Boolean query processing is more complex than merely fuzzifying the

indexing. The query representation can also be weighted. There are four types of generalizations: Boolean indexing and Boolean queries with non-Boolean retrieval status values, fuzzy indexing and Boolean queries with retrieval status values computed using fuzzy subset rules, Boolean indexing and fuzzy queries with retrieval status values and fuzzy indexing and fuzzy queries with the retrieval status value being calculated by some general function. The last generalization has problem with generating a "weight" for a term and document evaluation in terms of its relevance. A new and alternative function for document relevance evaluation has been mentioned. The authors use a threshold approach, rather than a weight. This can solve the last generalization problem. A new function form conducted and several criteria which specify necessary conditions for a proper document evaluation mechanism have been mentioned. The function form implies that one is given some partial credit (F/a) for the membership function's coming close but not exceeding the threshold. This credit is weighted by an increasing function of the threshold. This implies that as the threshold increases, a given percentage of partial is given more weight. The authors claim that fuzzy subset theory applies to document retrieval systems allowing non-Boolean index weights to be attached to the document and non-Boolean weights or thresholds to be attached to the individual terms in the query representation. This is a generalization of document and query representation and processing.

Bordogna, Carrara and Pasi proposed that query term weights as constrains in the fuzzy information retrieval (Bordogna, et al. [1991]). The authors state that the

Boolean model is widely used in many traditional systems and it suitable for a flexible query formulation; however, it also has some disadvantages. The Boolean model retrieves information items only into two classes: relevant and irrelevant. Therefore the model does not allows the ranking property of documents in descending order of estimated by a query and does not provide a method to solve with the imprecision in the query formulation. The author refers to previous work by Bookstein, Kantor, Buell, Kraft and Buell (Bookstein [1980], Kantor [1981], Buell [1982], Kraft and Buell [1983]). The authors state that Bookstein's work on the fuzzy set model as the first kind of semantics defines query weights as measures of the relative relevance of each term with respect to other terms. This model defines the relevance semantics in all aspects but the AND operator is associated with the lowest weighted term. This situation generates contradiction with the semantics of relevance. The authors state that Kantor's works is another difficulty that the notion of complementation when using relevance weights in the fuzzy set context. The authors state that Buell's worked on another approach with threshold semantics for query weights also have a problem: the meaning of threshold weights is not clear when apply to Boolean expressions rather than to single terms and this is a problem of semantics. The authors present an extended Boolean model formally described by means of the fuzzy set theory and the problem of query weighting in the existing models. In the author's model, a query term weight has the meaning of an ideal document-term relation value which be considered as a constraint on the stored document representations. So, the system will retrieve first documents whose index term weight is close to w (where w is a weight to

a term t in a query). This interpretation of query term weight as clear requirements of ideal index term weights permits interpretation of a query as description of one or more ideal documents for the user. The Retrieval Status Value (RSV) is expressed as the degree to which all constrains has been satisfied by each document representation in stored collection. The authors use fuzzy set theory to define the generalization of the Boolean model and the constraint is expressed by the formalism of fuzzy restrictions. RSV evaluation mechanism is defined and analyzed with respect to the Cater and Kraft wish-list. The author use a function E* to evaluate the matching of a query against a collection of documents. The authors state that function E* satisfies these criteria for an RSV mechanism: separability, Boolean restriction, Self-consistency, Term similarity, Weights of zero, Query weight volume, Binary Boolean operations and Unary Boolean operation. The authors claim that an analytical approach to the interpretation of weighted Boolean query has been presented in their paper. A query becomes a means of describing classes of ideal documents and expressing relativity criteria in order to distinguish query term weights from query weights.

1.7.3 On extending the vector space model for Boolean queries

Wong, Ziarko and Raghavan introduced the idea of extending the vector space model applies for Boolean queries (Wong et al. [1986]). The authors state that the Boolean retrieval systems do not apply the incorporating term correlations into the retrieval process. In another words, the weighted queries and documents is a problem for

Boolean retrieval systems. The authors refer to previous work by Buell (Buell [1981]). The authors state that Buell's work on the strict Boolean retrieval systems has the problem that there is no provision for weights of importance to the terms both in the queries and documents. The representation is binary that lacks various index terms and the output is not ranked. In most cases, the AND connectives tend to be too restrictive. The authors introduce a new information retrieval model, named Generalized Vector Space Model (GVSM). The new model solves the weights problem in the Boolean model and the queries used in this model are seemed as an extended Boolean expressions. In GVSM, a query is simply defined as a weighted vector sum of term vectors. However, this form of query does not help the user to explain clearly structure as can be done in Boolean systems. Therefore, the most common language used to express query structure involves Boolean logic and the query is considered to be a list of index term and weight pairs. This is the first important variation queries that are called the basic GVSM. The other one is the called unified GSVM which involves a scheme for expressing weighted Boolean queries as vectors. Queries are specified as a weighted Boolean expression in which are connected by AND, OR and NOT operators. The authors compare the unified model with the p-norm model which was applied for extended Boolean retrieval model. The authors use MEDLARS and CISI collections for experimental evaluation, because other collections used for information retrieval do not provide Boolean queries. The standard recall and precision measures are used for comparing the performance of different models. After comparing these models, the authors find that both models

handle weighted Boolean queries, both reduce to VSM and strict Boolean retrieval models under certain conditions, p-norm model involves the parameter p, which has to be experimentally determined and the extended query language satisfies more algebraic properties under unified GVSM. The unified GVSM is closer to the strict Boolean model than it is to VSM which means the roles of Boolean operators are rather strictly retained. However, p-norm model achieves more softening of the operators. The authors claim that it would be advantageous to provide a prescription to handle Boolean queries in the GVSM environment. The important part of basic GVSM is generalizing VSM to incorporate term correlations and document representation reduces to the vector sum of terms when terms are assumed to be orthogonal. The unified GVSM reduces to the strict Boolean retrieval model when each document is represented by its dominant atomic vector.

1.7.4 A logical formulation of weighted Boolean models

Pasi proposes a logical formulation of weighted Boolean models (Pasi [1999]). The author states that in order to model IR in the logical framework there is a need for a more general formal discipline. It is necessary to analyze the role of logic in IR by defining a model of information retrieval based on modal logics, which provides a general framework to define pre-existing IR models. The query evaluation process of the Boolean model and of weighted Boolean models will be exploited by analyzing the role of logic as a formal basis. The analysis will give better understanding of some query weight semantics. The author gives a formulation of the Boolean model that

expresses the evaluation structure of the Boolean query. Fuzzy implications can be employed to generalize the logical interpretation of the Boolean model and it is necessary to describe the extension of the Boolean model. An extended Boolean model gives a weighted indexing function that evaluating a query term and the index term weight and it is interpreted as the degree of relevance of document with respect to query term. To enrich the expressiveness of the Boolean query language, numeric query weights have been introduced as an extension of the basic selection criteria, which become then pairs term-weight. The author uses QT (t, q) as the terms appearing in a given query q and IT (t, d) as the index terms belonging to the representation of document d. So, the logical interpretation of a Boolean query is the formal expression of the constraint imposed by a query term: $QT(t, q) \rightarrow IT(t, d)$. "The expression representing the query evaluation structure of the Boolean query q = (t_1 AND t_2 OR (NOT t_3)) is the following: $(QT(t_1, q) \rightarrow IT(t_1, d) \wedge QT(t_2, q) \rightarrow IT(t_2, d)) \vee (\neg QT(t_3, q) \rightarrow IT(t_3, d))$." There is an extension of the logical interpretation of the Boolean model to weighted Boolean models. The IMP and QW is the importance of the index terms in document and in query representations. "A query of the type $q = < t_1, w_1 >$ AND $< t_2, w_2 >$ OR $< t_3, w_3 >$, has an evaluation which is formally expressed by the following logical expression: $((QW(t_1, q) \rightarrow IMP(t_1, d)) \wedge (QW(t_2, q) \rightarrow IMP(t_2, d))) \vee (QW(t_3, q) \rightarrow IMP(t_3, d))$." In the interpretation, the weighted query is a part of the formulation. The constant symbols are terms in the formulation and the logical connectives correspond to the Boolean connectives. The choice of the implication operator is important in the formalization, as it is strictly

connected with the semantics of the query term-weight. The author claims that the approach is based on the following considerations: terms are the most essential elements which evaluate the relevance in a query; the natural logical interpretation of the Boolean model is important and the degree of relevance related to the truth of the given interpretations. The author also claims that the most important part of this approach is to make the bottom-up structure of the query evaluation procedure clear and the implication connective is employed to express limitation controlled by a query term on the document representations.

1.7.5 Evaluation of traditional Boolean query

Patro and Malhotra estimated the success from characteristics of the Boolean web search query (Patro and Malhotra [2005]). The authors state that whether the query will successfully satisfy the users' requirements are unknown. Similarly, whether the Boolean query is efficient or not in finding the location of the best documents is unknown. The authors state that this paper relates characteristics of the search results which are meet the user's requirement. A program which compares the performance of humans and the search queries synthesized performed better than humans and gives an objective judgment of the search queries. The author states that use the precision and recall to measure the quality of queries is the common method. When search query, use the number of relevant documents first returned 20 links as the precision measurement. The authors use the same range as precision to identify recall. The authors collect data by using volunteers. Volunteers are students have different topic

area and some sample relevant documents. After they use the same search engine to search their topic, they revise their query, then compare the volunteer's query and the synthesized query. The authors use figures to show the relationship between recall and precision of the volunteer queries; the relationship between recall and precision after the queries' quality changed; the precision as function of terms in query and number of attempts to improve query and relationship between number of terms and precision. A good query returns higher values for precision and good recall which can be used as evaluation results of the method applied to the Boolean model. A good predictor of a successful Boolean web search query is four terms and above average recall. The authors claim that they found the characteristics of good queries that may give good sample of the query successfully achieved by users and their requirements.

1.7.6 Evaluation of Extended Boolean Operators

Lee, J. H., Kim, W.Y., Kim, M. H. and Lee, Y. J. evaluated the Boolean operators in the Extended Boolean Retrieval Framework (Lee et al. [1993]). The authors state that the Boolean retrieval systems have been the commonly used information retrieval system because of the efficient retrieval results and easy query structure. But the ranking is not supported and similarity coefficients cannot be calculated between queries and documents in traditional Boolean retrieval systems. The fuzzy set model and the extended Boolean model have been suggested providing the ranking function to the traditional Boolean system and they are logical extensions of Boolean model because they reduce to Boolean model when document term weights are restricted to

zero or one. But still two problems exist: incorrect ranked output in some case and complex computation on Boolean operators. The authors refer to previous work by Booksteinand Salton (Bookstein, [1980] and Salton [1989]). The authors state that Bookstein's work on the fuzzy set model generates incorrectly ranked output in certain cases because the MIN and MAX operators have properties adverse to retrieval effectiveness. T-Operators in the fuzzy set theory have the Single Operand Dependency Problem and Negative Compensation Problem. Salton's work on the extended Boolean model has solve the problem of the former by apply the E_{AND} and E_{OR} operators; however, it suffers from the complex computation. The authors give the T-operators and the corresponding operator graphs, and also, the averaging operators and corresponding operator graphs. Then using the graphs to describe that one pair of the Averaging operators of Fuzzy Sets Model and the operators of Extended Boolean Model overcomes the single operand dependency and negative compensation problems. These operators are defined as positively compensatory operators. The authors use two different document collections covering items the ISI collection and the CACM collection. They compare the precision of each operator such as T-operator and average operator to find the effectiveness of them. The authors use the document term weights to evaluate rank documents in Extended Boolean Retrieval Framework. "The weight of document is normalized as follow: W_{ik} = (TF_{ik}/maximum TF in document i)* (IDF_k maximum/IDF in document i) where Inverse Document Frequency define as IDF and Term Frequency define as TF." The authors state that positively compensatory operators provide higher retrieval

effectiveness than the others. One pair of the Averaging operators of Fuzzy Sets Model achieves similar retrieval effectiveness and higher retrieval efficiency in comparison with the operators of Extended Boolean Model. The authors claim that the extended Boolean model has overcome the single operand dependency problem of the fuzzy set model by developing the operators for the evaluation of the AND and OR operations.

1.7.7 Effectiveness of Extended Boolean Model

Lee analyzed the extended Boolean models (Lee [1995]). The author states that each document is indexed with a set of keywords or terms, and each query contains terms connected with the Boolean operators AND, OR and NOT in the traditional Boolean model. However, the traditional model does not provide a document ranking function because it cannot compute similarity coefficients between query and documents and this function reflects the relevance between query and documents that has the same objective with term weight. The author refers to previous work by Sachs, Radecki and Buell (Sachs [1976], Radecki [1979] and Buell [1980]). The author states that MIN and MAX operators have been developed to support ranking function in the past for Boolean retrieval system. However, they do not correspond well with human behavior for the calculation of query-document similarities and lead the fuzzy set model to generate incorrect ranked output in some cases. The author analyzes the behavioral aspects of different operators for AND and OR operations and the four important properties of retrieval effectiveness: single operand dependency, negative

compensation, double operand dependency and unequal importance. The author describes them through examples and the four important could decrease retrieval effectiveness in some circumstance. The author also suggests that the properties of positive compensation retrieval and equal importance may help retrieval effectiveness. The author defines an operator class called n-ary soft Boolean operators that is suitable for achieving high retrieval effectiveness. The author evaluates the effectiveness of sixteen different operators that can evaluate AND and OR operations in extended Boolean models. The author uses a new large data collection to evaluate the effectiveness of the different operators in extended Boolean models. The large data collection includes one of the TREC sub-collections which is called WSJD2. The author also uses document term weights to calculate document values and two famous weighting schemes have been used which are Fox-weights and INQUERY-weights. The author states that the experiment's results suggest that the single operand dependency problem may be more adverse to retrieval effectiveness than the negative compensation problem. The double operand dependency problem as well as the single operand dependency problem may be more adverse to retrieval effectiveness than the negative compensation problem. INQUERY-weights give better retrieval effectiveness to the fuzzy set and Waller-Kraft models than Fox-weights. Network Boolean has the positively compensatory property in some operand values. The effectiveness of the p-norm model is slightly better than the vector space model for the well-formulated Boolean query. The author claims that this paper does not present optimal operators, but the properties being analyzed can be used as a high base line to

approach optimal operators.

Pohl, Moffat and Zobel also analyzed extended Boolean models (Pohl et al. [2012]). Boolean queries have the disadvantage of being harder to formulate than ranked queries. They have the drawback of generating answer lists of unpredictable length and changes in the query that appear to be small might result in disproportionately large changes in the size of the result set. The queries of Extended Boolean Retrieval (EBR) models, such as the p-norm model, queries are slow to evaluate, because of their complex scoring functions and none of the computational optimizations available for ranked keyword retrieval have been applied to EBR. The authors describe a scoring method for EBR models and adopts ideas from the max-score and wand algorithms and generalize them to be applicable in the context of models with hierarchical query specifications and monotonic score aggregation functions. The authors also present the p-norm EBR model as an instance of such models and that performance gains can be attained that are similar to the ones available when evaluating ranked queries. Term-independent bounds are proposed in this paper, which complement the bounds obtained from max-score. It can be employed in the wand algorithm, also reducing the number of score evaluations. The authors evaluated the efficiency of their methods on a large collection of biomedical literature using queries and results derived from real searches. Three query sets were used against this collection: 50 simple, short PUBMED queries consisting of a Boolean conjunction only, 50 structured queries containing both conjunctive and disjunctive operators at

least once in each query sampled from the same query log, 15 complex queries. Properties of these queries were summarized. The authors counted the number of scored documents with scores below the entry threshold and above the entry threshold. The authors state that the proposed scoring method is often faster and it significantly reduces the number of candidate documents scored, postings processed, and execution times, for all query sets. Term-independent bounds method for short-circuiting candidate document scoring reduce the number of score calculations, especially on simpler queries and when combined with max-score. The query execution times of Boolean execution will be faster, however, it must be remembered that the result of a Boolean query is of indeterminate size. The authors claim that optimization techniques developed for ranked keyword retrieval can be modified for EBR and this leads to considerable speedups. Term-independent bounds provide added benefit when complex scoring functions are used and it will as a mean for short-circuit score calculations.

CHAPTER 2

Techniques used in Boolean based Question Answering System

2.1 Introduction

With the rapid development of the Internet, using quick and direct methods to achieve information has become one of the hottest discussion topics in the information technology research area. Currently, the search engine is an important tool using by billions of users and it has already made great achievements. The search engine searches the information according to the key words and provides users with related webpage lists, so that users has to review and find the direct information among a large number of links.

As a result, this drawback motivated the researchers to design a better searching tool which can satisfy users with direct answers not only related information. The Question Answering System was created as the solution to this problem and it can seem as an "Advanced search engine". A tradition Question Answering System is complex to be implemented because it needs the techniques from both Natural Language Processing research area and Information Retrieval research area. Question Answering System provides users with paragraphs of direct answers, not the list of documents or webpages. For example, when users ask a question "What is the location and population of Windsor?" on the search engine such as Google, it will return back three

millions of results and users will read those webpages one by one to find actual location and population of Windsor. But with asking the same question on the Question Answering System, it will give back only paragraph of direct answers that contain the details of the location and population of Windsor.

With the research involved in this topic, Question Answering System becomes more flexible and practical. From the close domain question answering system that contains high structured database to the web-based open domain question answering system, the improvement is obvious to researchers as well as users. Because the researchers focused on pioneering innovative, effective methods to develop the question answering system and they always try to use techniques to make the system understand the difficult questions properly, the Question Answering System can deal with different types of questions now such as factoid questions, definition questions and cross-lingual questions.

As an inseparable part of Question Answering System, Natural Language Processing technique will provide an appropriate analysis method, which includes both semantic and syntactic understanding of natural language. After applied this module, the methods and algorithms will help the Question Answering System perfectly deal with different class of questions. It is well known that Google search engine performs excellent searching results and user experience. It is one of the most popular search engines and has a large amount of users. As a well-developed search engine, Google

returned high quality related webpage back to users by using their own PageRank technique. It also consist an efficient Natural Language Processing module which can analyzes the questions provided by users. As, our experiment is mainly focus on the Information Retrieval research area, thus, applying Google search engine to preprocess the questions is the best way for our system to prepare proper candidate answers to users.

Information retrieval model is one of the important parts of information retrieval, which is an important module of Question Answering System (Kobayashi and Takeda [2000]). These models use mathematical methods, language structures and algorithm tools to process retrieval steps efficiently. The main elements of information retrieval step are query and documents, and their degree of matching which can be defined as relevance. The relevance of the query term and the document term is an approach to evaluate the effectiveness of the information retrieval process.

Boolean model is one of the common models that we used in information retrieval. It is based on Set theory (Salton, G., and McGill, M.J. [1984]) and Boolean algebra. Westlaw is the largest commercial Boolean system used today. The Boolean model represents the query and document in a simple and understandable way, their similarity based on whether or not they are meeting the Boolean expression (Patro and Malhotra [2005]). A Boolean expression should be a query consisting of terms/words that connect by Boolean operators: AND, OR, NOT.

Boolean model is based on logical judgment, which defines a set of a binary variable to represent document. For query “A AND B”, retrieval document should include both A and B; for query “A OR B”, retrieval document should include either A or B; for query “A NOT B”, retrieval document should include only A, B must not be included in it. If there are more than one different Boolean operators in the query, the priority order should be: NOT, AND, OR. For those queries tjat have more than one of the same Boolean operator, the priority order should be: from the first left operator to the right one. For example, with the query “Toronto AND bookstores NOT Indigo”, users will find the information about Toronto bookstores that are not Indigo. Boolean retrieval model is simple and effective. It can reduced the range of retrieval and return better results when the AND operator being used as well as using OR operator to improve recall which means users can retrieve more relevant documents. For example: use University of Windsor AND School of Computer Science instead of only University of Windsor; use breaststroke OR backstroke instead of only swimming.

To prepare for the experiment, firstly, we have designed a Boolean based Question Answering System which can allow users to ask normal questions as well as the questions contains with Boolean queries. The system is an English Question Answering System that can answer the questions asked in English and it provides users a friendly user interface. Thus, users can figure out how to use it quickly and it shows the answers clearly. Once the users submit their questions to our system, the

system will first save the question in a local document and then send those questions to Google search engine. After Google's analysis, the system will achieve the retrieval results from Google and save it in the local document. Then, the system will extract the top ranked links (URLs) and snippets that are used for query expansion. The snippet results from the Google search results will provide rich related terms and then add those terms to the original question to extend the question. In this way, the system can improve the retrieval results because the range of existing relevant documents has been extended.

After extracting the URLs from the results, the system will use a web crawler to download the corresponding webpages and save them into the local folder according to their order. The system will analyze those webpages and prepare the candidate documents, which will be used as indexing documents later. To analyzing those webpages, the system has to extract the text information from each of them. Then, the system will separate the text information into paragraphs with the given structures and save the results into ordered documents in another local folder.

Before the system indexes those documents, there is an important step applied in the system that is the so-called "question processing step". This step is a vital part in information retrieval research area as well as Question Answering System. This step guarantees the system uses correct query terms from the question to flit and index candidate documents, and thus efficiency will be improved.

The system will use a Stop list to remove high frequency words and low frequency words, which are non-important words from the questions asked by users. The high frequency words such as “a” or “the” may occurs in almost all documents, therefore, these index terms do not help the system to distinguish the candidate documents while process indexing step. For the very low frequency words, they may not always occurs in the candidate documents and some of them even only occurs once among the candidate documents. So, these words were seems as irrelevant terms and the system will remove these terms from the index terms. These unimportant words such as “on”, “under” and “that”, these words will also considered as the terms which decrease index efficiency. Although these words are presented a high frequency in the documents, they do not involve any meaning in the questions. Thus, these types of words or terms should be removed during indexing, and then the system will use significant terms to retrieve the candidate documents efficiently. Especially, the technique performances excellent while the system deal with large amount of data.

Another technique that will be applied in the system during indexing is Stemming. With this method, the system will remove the suffixes that are the ending of words. Because the words shared with same stems but different suffixes may have same meaning in English, thus, to avoid the missing of indexing, the system will distinguish those words and index those words as the same one. For example, “do”, “did”, “done” have the same meaning but different formats in different tense, so the three words will

be indexed as the same word. These algorithms above provided by famous researchers in Information Retrieval area and the name of the algorithm is called Porter Stemming. (C.J. van Rijsbergen, Robertson and Porter [1979]).

After the question processing step, the system can index the terms left in the queries which are the extremely relevant terms. Therefore, when the user submits a question like "What countries speak both French and English?", the system will process it and convert it as "country speak French and English", then use this query to index and retrieve the relevant candidate documents.

In the document indexing step, the system will retrieve the questions which contains the Boolean expression using Boolean retrieval model. The system will use the Query Parser to identify the Boolean operators in the query. Then, the system will use the prepared index terms to index the candidate documents collection.

There is an algorithm used to finish the indexing procedure in this step and it will be described as follow: If the document contains the prepared index terms in the query, we would say the query make a "TRUE" statement about the document and all documents being given a "TRUE" statement will be retrieved. That is to say, for the term set S= (T_1, T_2, …, T_n), we give them logic operator AND, OR and NOT that structured as different expressions. If the value of the expression is true, the document's retrieval value is one, which means relevant. For example, query "coffee

AND Tim Hortons" would be true for a document indexed by "coffee, Tim Hortons", but not for document indexed by "coffee, Starbucks". There is also an inverse method. For each term we store a posting list of documents that contains them and give documents order number. Then, we will compare the lists from front posting according to different Boolean operators. For example, merge lists when processing with the AND operator. If the number of documents are equal, put it into the result list, otherwise, advance the smaller one; merge list but avoid duplicates in the result list when processing with OR. If the number of documents are equal, put one of them into the result list, otherwise put all of them into the list. Then move forward all lists together and put all the documents' numbers into the result list, which is at the end of the smallest list. Never put a document number from the term after the NOT operator (here call it term one). Put the equal document number into remove list and forward all list, if not equal move forward the list of term one. When this list is larger, move other lists. Put all other list' document number into results list and subtract the document number from remove list when moving to the end of the list term one.

After this step, our system will retrieve and match the relevant answers to the given questions. The system does not apply any ranking facility to the Boolean query because the weight of the query should not be used in the traditional Boolean retrieve model and there are still some other models such as fuzzy set model which try to solve the ranking problem in Boolean model. The candidate answers will rank as ordered list only according to the search results from Google. Because the candidate answers

from the documents are extracted from the top rank webpages from Google and Google used PageRank which is an excellent webpages ranking algorithm and can be presented perfect ordered webpages with the given searching query, thus, the system will use the same order number of the documents as the ranking results for the candidate answers.

To avoid some disadvantages of Boolean retrieval model and provide a better user experience, the system applied a question reformation module. For example, it may not retrieve anything back when the request range is too small; many terms connect by AND operator. In the other hand, it may retrieve too many things back when the request rang is too wide: many terms connect by OR operator. Thus, to build a module is necessary and this module will allow users change their question until them satisfied with the results. In this step, users can change their question according to their own opinion and the system will send the new questions to Google again to get back new results and then process the whole procedures again to provide new answers to users.

To sum up, the system is focus on using information technology and Google search engine to allow users retrieve Boolean query that is more effective and precise. In the following sections, the related works and the details for implementation of the Boolean based Question Answering System will be discussed. At the end, some experiments results for testing the designed system will be showed in details and there

is a conclusion section for the total research as well.

2.2 Related works

As an important part of our research, an original Question Answering System that is worked base on the Vector Space Model will be introduced. In addition, the system applied Google search engine as one of the retrieval step, therefore, the analysis of PageRank algorithm for Google search engine will be discussed as a part of related works as well.

2.2.1 The original Question Answering System

Many researchers have involved in the Question Answering System from 1960s. Currently, there are lots of different types of Question Answering System using different models and algorithms. Among those systems, a Vector Space Model based Question Answering System provides the basic ideas for our designed system (Zhao [2012]). This original Question Answering System only worked for the normal questions provide and it cannot answer the Boolean expression based questions because the Vector Space Model cannot provide the solution for these types of questions. Therefore, the different between our designed system and the related system are the queries and candidate documents being matched in different methods in the document indexing step and using different ranking methods.

The vector space model is currently the most widely used information retrieval model

because it has a wide range of adaptability and vitality. Vector space written permission of model represents queries and documents in the form of vector, to determine the similarity of the query and document by calculating the similarity of the vector, and the document results are sorted by similarity after return. In recent years, the studies of the vector space model are growing vigorously, mainly concentrated on the extending study of the vector space model and the application of vector space model. The vector space model is only a theoretical framework. According to the need, different weights evaluation functions and the similarity calculation method can be used such as Term Frequency (TF) and Inverse Document Frequency (IDF) (Manning, Raghavan and Schütze [2008]).

Vector space model takes it for granted that document consists of relatively independent term group (T_1, T_2, ..., T_n), and each term T will be endowed with certain value according to its importance in the document; therefore, the document represents a point in an N-dimensional space constituted of each term (Castells, Fernandez and Vallet [2007]). In addition, all documents and users' search are capable to map into this text vector space, and the degree of similarity between users' search and retrieved document can be measured with the angle among vectors. This model takes traits of contents of texts into account; besides, the measurement of similarity between documents is considerably simple. Consequently, some retrieval systems have adopted this retrieval model, and, achieved good affects.

The similarity is the only way for all kinds of retrieval model to define the relevance of the queries and candidate documents and the indexing procedure will use it to judge which document should be retrieve back.

In the related original Question Answering System, one of the important similarity calculation components is Term Frequency (TF). It represents the times of the index terms occur in each candidate document. In other words, the system will record the number of the appearance for the index terms. For example, if the index query contains a term “university”, the system will calculate the term frequency in each document. If the first document returns five as the results and the second document return two, the system will retrieve the first document because it has high term frequency, which means more relevant. It is a reasonable method because a high frequency word in the document may be one of the most important word in the candidate document set which system being retrieved back, such as the word “cuisine“ will occur frequently in a collection of documents about cooking.

The term frequency gives a high weight to a term that occurs frequently in a document but that it is not good at distinguishing between relevant and non-relevant documents. Thus, another important component being used which is called the Inverse Document Frequency (IDF). This weight is defined as the inverse proportion to the number of documents contains a term in the total document collection. The formula is as follow (Manning, Raghavan and Schütze [2008]):

$$W(i) = \log \frac{D(N)}{D(n_i)}$$

In the formula, $W(i)$ is the weight of the term i, the D (N) is the total number of documents in the set, $D(n_i)$ is the number of documents in which term at least occurs once.

Therefore, once the system has retrieved back the candidate document collection, the number of documents which contain a certain term in the query will be counted. In another word, this formula defines the occurrence of a certain term in a document collection. For instance, if there are 100 documents in the collection which being retrieve back by the system from Google search engine and there are 50 documents contains the index term in the query, the weight value for IDF will be 0.30103. More importantly, compare with the Term Frequency method, the Inverse Document Frequency (IDF) will calculate the term frequency in an inverse way: when a certain term occurs in the document collection frequently, it will share a lower IDF value. This method is still not very good at distinguishing between relevant and non-relevant documents because it only calculated the global frequency of the term.

Consequently, it is necessary to bring the complete formula to calculate the weight, which donates both local frequency of a term in a document and global frequency of a term in a specific document collection. Thus, TD×IDF is the best method to be taken into account and the formula is as follows (Manning, Raghavan and Schütze [2008]):

$$W_{ik} = TF_{ik} \cdot \log \frac{N}{n_k}$$

In the formula, the TF_{ik} represents the term frequency of occurrence of term T_k in the document D_i, N represents the total number of documents in the retrieved back document collection, n_k represents term T_k's frequency in the documents which it occurs.

Thus, the new formula involves a new method to calculate the importance of an index term as well as a new way to distinguish the relevance between different index terms and candidate documents. The weight W_{ik} describe the ability of the term to distinguish the content of the document. When a term has a high frequency in the document collection, the ability of that term to distinguish the content of the document will be poor. On the other hand, when it has a high frequency in a specific document, indicating that it has the stronger ability to distinguish the content of the document. Then, the stronger ability term can be selected as the represent term (Hallinan [1993], Weibull [1951], Chau and Chen [2003]). This represent term could be the "best" term to distinguish the relevant documents and non-relevant documents which involves as a complete formula to calculate the similarity between a query and a document. For example, if there are 100 documents in the collection which are being retrieved back by the system from Google search engine and there are 50 documents contains the index term in the query, the index term occurs in the first document for five times and it occurs in the second document for two time, so, the certain term TD×IDF weight for the first document will be 1.505 and the weight for the second document will be 0.602. Therefore, the same term will share different weight values in each different document,

which means this method represent a particular term's weight in each different document.

After the TD×IDF weight being assigned to each term，the system will store the terms in each documents as well as the weights in a location. The Vector Space Model based Question Answering System will use a matrix to save these information above. The matrix builds as follow(Manning, Raghavan and Schütze [2008]):

$$\begin{bmatrix} & Term_1 & Term_2 & \dots & Term_n \\ Doc_1 & WT_{11} & WT_{21} & \dots & WT_{n1} \\ Doc_2 & WT_{12} & WT_{22} & \dots & WT_{n2} \\ \vdots & \vdots & \vdots & & \vdots \\ Doc_m & WT_{1m} & WT_{2m} & \dots & WT_{nm} \end{bmatrix}$$

In the matrix, the columns are the documents' number from the collection and the rows are the terms from the document collection. Suppose there are n terms and m documents being retrieved back, the TD×IDF weight for term one in document one will be saved in the matrix as WT_{11}. After set up the whole matrix and save all the TD×IDF weight information, the system will use it to retrieve document correctly.

The Vector Space Model also gives another similarity calculation method that will be used as the ranking facility in the original system. In the Vector Space Model, the target document and user's query are represented by terms, which apply term weights. The degree of similarity between the document terms and the query terms is measured by the angle between them. The similarity formulation as follows (Manning, Raghavan and Schütze [2008], Na, Kang and Roh [2007]):

$$\mathrm{Sim}(\mathrm{Q},\mathrm{D}) = \cos(\mathrm{Q},\mathrm{D}) = \frac{\sum_{k=1}^{n} W_{qk} \cdot W_{dk}}{\sqrt{\sum_{k=1}^{n} W_{qk}^2} \cdot \sqrt{\sum_{k=1}^{n} W_{dk}^2}}$$

In the formula, the W_{qk} represents the weight of the terms in the query and W_{dk} represents the weight of the terms in the query. To be more specific, the Q and q represents the terms in the query and k is the order number of these terms. The D and d represents the terms in the documents and k is the order number of these terms. Accordingly, this formula will calculate the similarity of the total number of n terms in the query for each document. Each term has difference frequency in different documents. As the system has already store the TD×IDF weight information before, therefore, W_{dk} is a constant number. Suppose the terms in the query share the same weight that means every W_{qk} equals to one in the formula, the similarity value will be calculated. For example, when the user gives a question "What is the capital city of China", the system will firstly process the question and then the question becomes "capital city China". In this query, each term shares one as the TD×IDF weight. After the system prepare the document collection and the matrix that attached the TD×IDF weight for each term in collection, suppose the system find the first document contain the terms "Beijing", "capital", "city", "China" and the TD×IDF weight for each term are: seven, six, seven, six. Thus, the similarity of the document and query would be about 0.99 after calculation that means the document and the question are very relevant, so that the first document will share a high ranking order in the system.

Accordingly, the method attached each document achieves different similarity based

on the questions provided by users and help the system distinguish the relevance of each document, so that the documents in the collection will have an order ranking number based on the similarity value being calculated. Moreover, the ranking facility will help the user find their answers quickly because the top ranked answer will be more relevant comparing those after them. Similarly, there are some other methods being applied to calculate the Term Frequency. For instance, using statistical method can collect the frequency of the terms and then the properties can be extracted from the statistical results (Miller, Leek and Schwartz [1999], Lafferty and Zhai [2001], Ponte and Croft [1998]).

2.2.2 Google's ranking algorithm performance

Google search engine has applied many different kinds of algorithms that give the ranking order of webpages according to the relevance, such as Hilltop, PageRank, ExpertRank, HITS and TrustRank. Among them, PageRank is the most famous and useful algorithm. It has become the core and proprietary technique of Google. Many researchers use it to analyze the relationship of links and webpages.

PageRank is used to evaluate the importance and relevance of some specific webpages comparing to other webpages being send back at the same time by the search engine. The importance and relevance is the factor to evaluate the optimization of search engine. Larry Page and Sergey Brin created PageRank algorithm in 1998 in Stanford University. It realizes the ranking function by the importance of links and it confirms

the importance degree for the webpage with the relationship of hyperlinks. Basically, this algorithm will define the votes of each webpages. For example, the link from webpage A to webpage B will be defined as the vote webpage A gives to webpage B. Google will give an new ranking order according to the vote webpages (even the votes from subpages) and the target being voted. Thus, a webpage with a high ranking order will impact and even improve the low ranking order webpage.

To be more specific, the PageRank algorithm will let the links give the votes to each other. The number of votes for a webpage depends on the importance of the links that connect and point to this webpage. When the webpage has one hyperlink, there is one vote of this webpage. If many links point to the webpage, it will have a high importance as well as large PageRank value. On the other hand, the webpage does not have any in-links that means it does not have any importance and the PageRank value is zero. The recursive algorithm calculates the PageRank value of a webpage according to the importance of connecting links.

If a webpage A is linked by a webpage B, it means webpage B recommends webpage A. The webpage B will distribute the PageRank value to the webpages that it linked such as webpage A. Therefore, Webpage A would get accumulate PageRank value due to different links' contribution and webpage A has high PageRank value means it is very important. Besides, webpage B will get high PageRank value from other when B is very important, then, webpage B will improve webpage A' PageRank value and

webpage A become more important. The formula to calculate the PageRank value is as follows (Page, Brin, Motwani and Winograd, T. [1999]; Xing and Ghorbani [2004]):

$$\mathrm{PageRank(A)} = (1 - \mathrm{D}) + \mathrm{D} \sum_{i=1}^{N} \frac{\mathrm{PageRank}(\mathrm{P}_i)}{\mathrm{O}(\mathrm{P}_i)}$$

In this formula, PageRank (A) represents the PageRank value of webpage A. The character D represents the Damping Factor. Because some of the webpages do not has any incoming links or outgoing links, so the PageRank value cannot be calculated. To avoid this problem (LinkSink problem), the Damping Factor always defines as 0.85. The $\mathrm{PageRank}(\mathrm{P}_i)$ represents the PageRank value of the webpage P_i. $\mathrm{O}(\mathrm{P}_i)$ represents the number for outgoing links of the webpage P_i. The initialize PageRank value are the same. As the subpage may be important, the iterated operation is required. After repeatedly iterated operations, the PageRank value will reach stable.

PageRank is a static algorithm and the PageRank value can be calculate without the working Internet. Thus, it decreases the ranking time when users search the queries and it also decrease the response time for searching. However, there are two drawbacks of this algorithm. Firstly, PageRank algorithm does not work well with the new adding webpages, because those new webpages lack the outgoing and incoming links. Thus, the PageRank value is small. Secondly, PageRank algorithm only use the number of links and the importance to calculate the ranking order, so it may ignore the relevance of the topic for the webpages to some extent. Accordingly, some webpages with unrelated topic may achieve large PageRank value and the accuracy of the search

results will be impacted. In order to solve these drawbacks, some algorithms being created only focus on the content of the webpages.

For example, suppose the webpages 1, 2, 3, 4 structured as follow (Rogers [2002]):

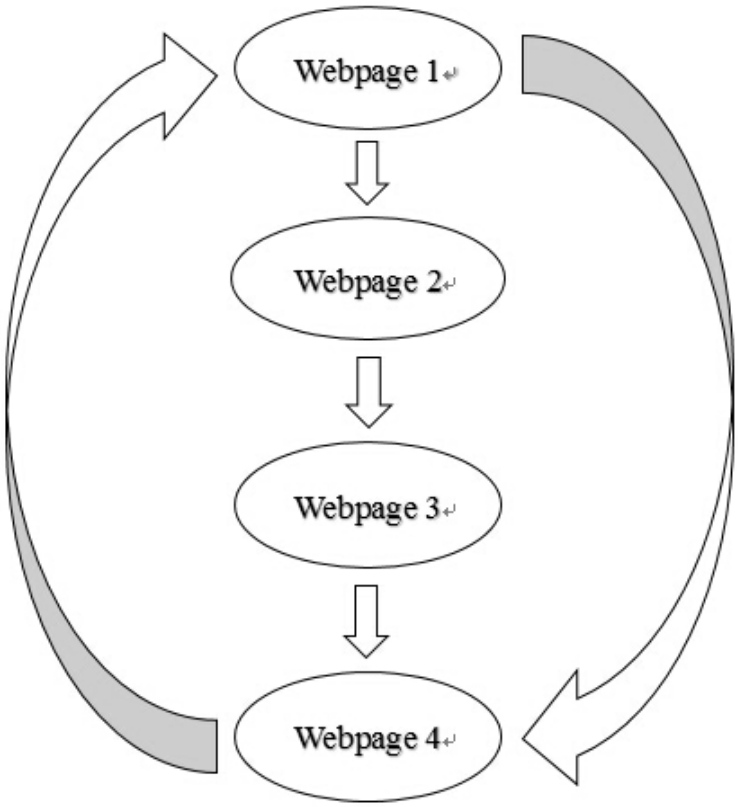

Figure 1

In Figure 1, there are four webpages and we assume the Damping Factor is 0.85. Thus, the PageRank value for the four webpages will be (Rogers [2002]):

PageRank (1) = 0．85 + 0．85 PageRank (4)

PageRank (2) = 0．85 + 0．85(PageRank (1) ／ 2)

PageRank (3) = 0．85 + 0．85(PageRank (2) ／ 2)

PageRank (4) = 0．85 + 0．85(PageRank (1) ／ 2 + PageRank (3))

From the calculation formula above, we firstly give an initial value to each page. Then

use iterated operation to calculate and update the PageRank value until the value is a constant value. Finally, we will find the webpage, which achieve the highest PageRank value. Most of the researcher found that the webpage has more incoming links will be ranked top.

The PageRank value only releases few times per year, thus, it is not a precise value and Google also tries to find more accurate value to calculate the ranking order. PageRank value is one of the factors that Google used to calculate the search results and structure the ranking order of webpages. It is possible for the webpage owning a low PageRank value that ranked before the webpage owning a high PageRank value when some particular search query is given. PageRank value may be irrelevant for the topic of the specific webpages as we talked before. Google also used the relevance of the links to calculate the ranking order. Accordingly, the PageRank value should not be the only factor that being considered for the search engines. From this point of view, we would found that constructing the incoming links would increase the PageRank value. Thus, the irrelevant links being constructed would change the ranking order, as Google do not test the relevance of the topics.

Actually, the Page Rank value still has some drawbacks but there are some solutions appearing to work out these problems. For example, the algorithm is good at calculating the old webpages and those webpages will achieve a high possibility to own a high PageRank value compared to the new webpages. However, it is not always

accurate because some of the new webpages do not connect with the high PageRank value webpage also have the possibility to contain some important information. So, there are some new methods appeared such as Accelerated ranking algorithm which focused on resolving this drawback. This algorithm adds the slope of the curve fitting in the PageRank value. Although the PageRank algorithm has some drawbacks, it is still a believable factor and being accepted by most of the researchers and users. The users tend to believe the webpage is famous and with high comments. This is what the PageRank algorithm aims to do. Google accepts the concept, so that Google will decrease or recalculate the PageRank value to punish the irrelevant and useless webpages.

The Topic-Sensitive PageRank algorithm is focused on solving the topic relevance problem, which is another drawback of PageRank algorithm (Haveliwala [2002]). Basically, this algorithm is the advanced version of PageRank algorithm. It will predefine some topic based units such as sports, entertainment or technique. Each of the topics will be organized as a vector. The algorithm will calculate and predict the possible interesting topic for users and give back the ranking order according to the user needs. There are two main procedures for this algorithm. The first step is calculating the vector sets and the second step is confirming the search topic which involves calculate the relevance and similarity. The PageRank algorithm works based on the "Random walk Model" which means the search engine will select a random link for users when they want to look for a new webpage rather the webpage they are

stay. The Topic-Sensitive PageRank algorithm introduces a hypothesis, which is practical. Normally, the users have their own interesting area and each webpage is related to some of the topics as well. After users reviewed a webpage, they may tend to skip to the webpage share the related topic with the webpage they have reviewed. In another word, the algorithm will combine the relevance of users' interests, the topics of the webpages and the new connected webpages' topics. So, it is more suitable and acceptable for the users when they search information. PageRank algorithm will consider the importance of all webpages and give each webpage a unique PageRank value. However, the Topic-Sensitive PageRank algorithm will contains different topics. Consequently, each topic will have a PageRank value, which means each webpage will achieve different PageRank value based on different topics. PageRank algorithm can be used independently and it is one of the factors for calculate the ranking order because it is irrelevant with search query. On the contrary, the Topic-Sensitive PageRank algorithm's calculation results are relevant with the search query and it can be used independently as a similarity calculation formula. After receive the search queries from the users, the algorithm will use the classifier to match the predefined topic and the results of matching will be used in the ranking calculation formula. Google search engine already applied this algorithm in the Personalizing Search module.

As we have mentioned before, the PageRank algorithm is not the only method applied in Google search engine and there are some others efficient algorithms. HillTop

algorithm is one of the famous ranking algorithms used by Google. During 1999 to 2000, an engineer of Google named Bharat created it (Bharat and Mihaila [2000]). The basic idea of this algorithm is same with PageRank algorithm. It also used the number of income links and outgoing links to calculate the ranking order. However, it involve the topic of the webpage as the calculate factor. Different from the Topic-Sensitive PageRank algorithm, the HillTop algorithm defines the webpages that will be effect by the topic as the "Expert webpages". The HillTop algorithm avoids the drawback of PageRank algorithm that webpages cannot achieve a high ranking order by adding many useless links. The HillTop algorithm not only provides a method to define the relevance between two webpages but also being used as a technique to distinguish the similar webpages in Google.

To sum up, Google search engine has applied many different and useful algorithms that provide high reliability ranking order for users. Therefore, users can find the information quickly by view the top links, which is sent back by Google search engine.

2.3 System Structure

The Flowchart is the best method to manage the idea during the preparation period of a designed system or program. It also presents and describes the whole ideas of the system because it contains all the modules being used in the system. To general show how the Boolean base Question Answering System worked, the flowchart will

introduce the different modules and how does they works. Also, it explains how the answers are made when users ask a random question on the system. Every strategy will be introduced shortly; therefore it will summarize the details that we will talk about in the following sections.

After the analysis of the related system provide by Zhao (Zhao [2012]), we designed a Boolean based Question Answering System. The complete structure of the Boolean based Question Answering System is shown as a flowchart.

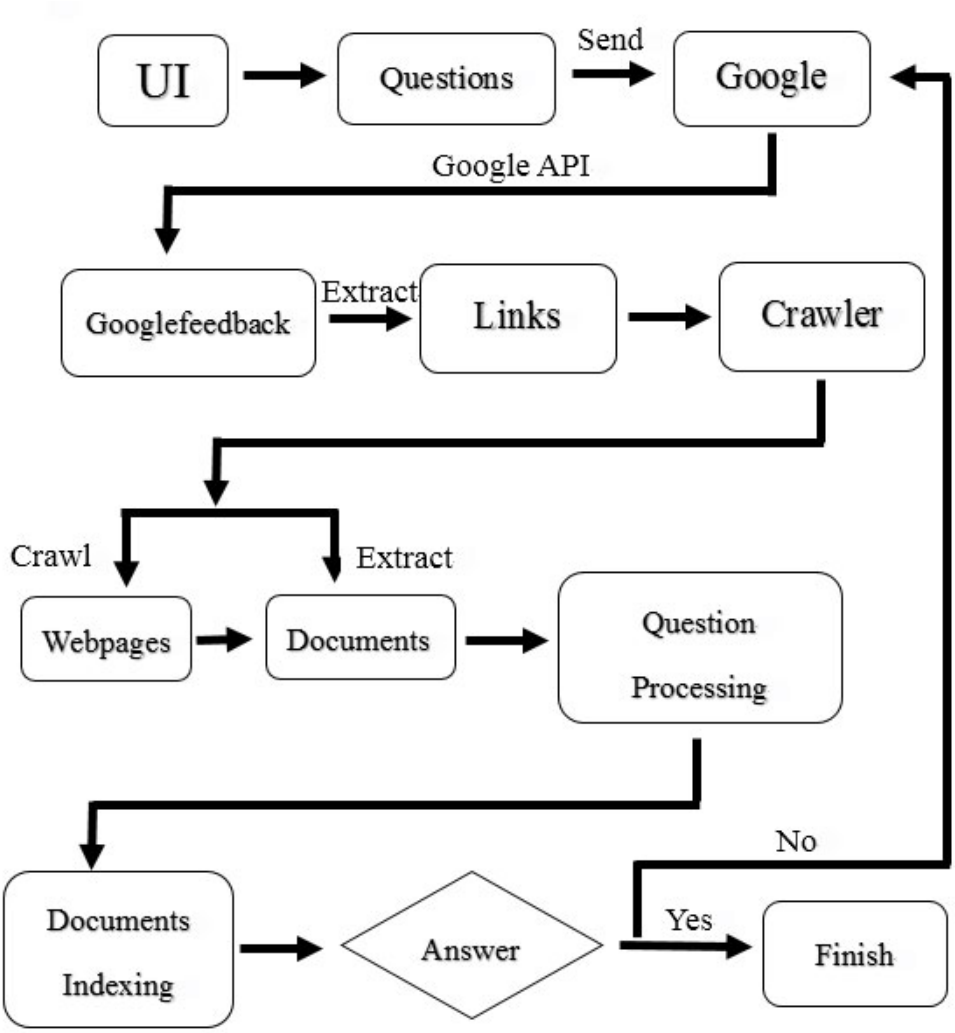

Figure 2

In Figure 2, all the important modules being applied in the designed system and the results for each step that is shown clearly and separately. The arrows define the directions of the procedures. It is easy to understand how the final answers are being achieved with step-by-step procedures.

Initially, the designed system provided a friend user interface, which is easy for users ask question. The users can find the question section as well as the answer section quickly. Once the users type their questions in the question section, the system will recognize the questions being provided and save those questions in to a local document, which will be process later. At the same time, the system will also sent the question to the search engine to get the direct information.

For all kinds of Question Answering System, the Natural Language Processing is necessary and this module cannot be ignored. Thus, the designed system will firstly send the questions to Google, which contains a sophisticated Natural Language Processing module. So, the questions will be process through this module. Then, the search engine will treat the question as a normal query and will send back millions of relevant links. However, the system will not use all of those links because only top ranked links are useful. As we talk in the last section, Google use efficient algorithms such as PageRank, Hilltop and HITS to retrieve the information and rank the webpages corresponding links in descending order. Therefore, the top ranked links are more useful and the designed system will receive that important information. To be

more specific, the designed system will firstly save the information from Google into the local document and there is a lot of information contained in it such as “totalResults”, “context” or “link”. Secondly, the designed system extracts the information followed by link, then, collects and saves the links information into a local document. After that, the system will use the web crawler to crawl all corresponding webpages based on those links and save the webpages as html format in the local folder. The web crawler will analyze the webpages about their text information and cut this information into pieces. Thus, this information will be saved as different passages as different documents. The order of those documents will be the same with the webpages and they will have a more specific sub-number based on their location in the webpages. The procedures above are used as our valid data, which is used to prepare the candidate answers. This is one of the important step used in web based Question Answering System.

As the candidate data are available, the designed system will continue to process another important step that is the question processing step. To prepare for the indexing, the system will firstly covert the question being saved into the query which will used be to index document. Because there are many useless information and terms in the question provide by users, detecting the important terms or words is a necessary step, especially the Boolean expression and Boolean operator here will be used to flit the candidate documents. The designed system will set up a list to remove non-important terms. This method has been applied in different kinds of information retrieval system

before by many researchers and they provided different kinds of efficient algorithms, which can be used in this step. The designed system used one of the famous methods called Porter Stemming (Porter [1980]) to achieve the query, which will be used as the indexing query.

Apart from the question processing step, there is another important step the system will contain that is called document indexing step. As the system has stored the document text information along with the ordered document ID, thus, the index procedure is easy to be applied. The system will first build the Boolean query, which is from the original question. There are three types of Boolean operators AND, OR and NOT. Accordingly, the terms connected by the Boolean operators AND means that both of the terms should be included in the candidate document. The system will use add method in Java to add the Boolean Clause and define both of the two terms as must occur. The terms connect by Boolean operators OR means that at least one of the terms should be included in the candidate document. The system will use add method in Java to add the Boolean Clause and define one of the two terms as must occur and another as should occur. The term connected by Boolean operators NOT means that the terms before NOT operator must be included in the candidate document. On the contrary, the term after NOT operator must not be included in the candidate document. The system will use add method in Java to add the Boolean Clause and define the one of the terms before NOT operator as must occur and the term after NOT operator as must not occur. This is the method to build the independent Boolean query or Boolean

expression. The system will also use the Query Parser to create the Boolean query. This method accepts different text formats. Therefore, the document collection can be indexed after the system builds the complete Query Parser; this step will let the system to flit and retrieve the candidate document back which will be used as the candidate answer showing as a list in the answer section.

After the document indexing step, the designed system gives back the candidate answers as different paragraphs in the answer frame of the user interface. All the answers shows as a list and each have a ranking number. The top ranked answers will mostly be the correct answer, which users really want. If the users are satisfied with the answers which the system has provided, they will finish the search by click the exit button. However, if the users have reviewed all the answers being provided by the system and they are not satisfied with answers, there is another option button provided by the designed system, which is the question reformulation step. It can be seen from figure 1, there are two optional directions in the answer module that means the system allow the users to choose the direction according to their opinion. Thus, if they choose to continue the reformulation step, the designed system will provide the user the addition step. In this step, the users can re-type their questions and the new question will have different format and contains different terms as well as different Boolean expressions. Consequently, the designed system will prepare totally new answers to users. To be more specific, the system will send the questions to Google search engine again and then process the same steps again until the new answers being given out.

This step allows users to ask questions continuously until they find the correct answer. It is necessary to be implemented because it enhances the user experience and the system is becoming more reliable and practicable.

The structure showing above gives out the basic and general design ideas and modules in the system. The following sections will introduce about the details of each important procedures and the corresponding results in the designed system.

2.4 Searching and document collecting

As the first core procedure of the Boolean based Question Answering System, the system will directly send the questions from and to the Google search engine, which is one of the most popular and best performance search engine in the world. This step is necessary and reasons are listed as follow.

First of all, the Boolean based Question Answering System was designed as a web-based Question Answering System, which used to solve the questions with unlimited knowledge domain. In other words, this system is an open domain Question Answering System that can deal with different kinds of questions from different users. Thus, in order to apply an unlimited knowledge answer set, the system has to change the traditional fixed database, which cannot satisfy all kinds of need from the users. There are abundant information resources on the internet and it provide better data sets and the combination of the question answering system and the internet is the best way to prepare the candidate answers to users. Therefore, we chose Google search

engine as the tool to set up the “database” which provide numerous of documents and resources.

Besides, the Natural Language Processing module is one of the inseparable parts of the Question Answering. However, our Boolean based Question Answering System focuses on the information retrieval research area and to evaluate the Boolean retrieval model. Thus, in order to avoid the impact of this module, the system used the Google search engine, which contains a sophisticated Natural Language Processing module. Since the questions asked by users send to Google directly and Google search engine prepare the search results, it will perfectly solve the problem from NLP module.

Last but not least, the Google search engine provides API that is easy to be implemented and connected with the designed system. Since the system sends the information to Google search engine, the search engine will use the Internet to prepare the information and send back millions of ranked links. The designed system will first get access to the search engine through the API and achieve the search results as well. Google search engine performs a quick executing time and the search results include high quality ranked links. As we talked before, Google search engine has efficient algorithms, which guarantee the search results. Thus, it is the best way for the designed system to prepare the document collection with Google search engine’s help.

Therefore, since the users ask the Boolean expression based questions, the designed

system will firstly send the questions to search engine. At the same time, the questions will be saved in the local document, which will use the question process step. After the designed system connect with search engine and achieve back the search results from Google, it will directly save the information into a local document. In this document, there is a lot of information such as the title of each links which is the summary of the content of the corresponding webpage, the executing time for the search as well as the links which will be used later.

Since the system has the information from Google, it will firstly extract the link information from the feedback document and then save those links into another local document. To be more specific, as we have talked above, according to Google's excellent algorithm, the top ranked links will contain the possible information which the users want, so that the system will only extract the top 10 links from the search results and the top 10 linksoccur in the first search results page from Google. Many reports, which provided by the marketing and advertisement researchers show that the first page is the most important page for the users. The researchers use CTR which stands for Click-through rate to evaluate how important of the website for users. The links that are more popular will get higher CTR compare with low ratio. Some of the researchers applied it on Google and found interesting results that around 40% Click-through rate holds by the first results from Google as well as the first page occupy a large percent compare with the second page. From this point of view, we would find that the users are more like to visit the first page, especially the first search

results from Google. In other words, the first page search results from Google are very important and estimable for users. Besides, in order to save the search time and the store space, removing the useless information and achieving the possible information is very important. Thus, the designed system only extracts the top ten links to avoid the unnecessary information being processed later.

Secondly, the designed system will use a web crawler to download the corresponding webpage to get the text information. Since the designed system is focused on the information retrieval research area, if the system only shows the Google feedback to the users that means there is not any contribution and improvement for the research. Thus, the document collection is only used to prepare the answer collection. This document collection will be processed in the core procedures of the designed system such as the question processing and document indexing step. This step will use the information retrieval techniques to prepare more specific answers to users. This is the difference between Google's work and the Boolean based Question Answering System's work. The designed system is used to improve the search results from Google; it not only shows the answers instead of links but also filters out some useless information, which retrieved from Google.

For extracting the text information, the designed system will not use the original format from the webpages. As we know, the body of the webpage is organized as text format, however, not all of the information is related to the question provide by users.

Thus, if we separate the text information sentence by sentence, it may contain little information because it is too short. As a result, the system misses some important information in the following retrieve step. If we index all the information, it will decrease the efficiency because it contains some useless information such as advertisement text information. The designed system will cut each webpage into short passages, so that it will guarantee the system would not miss any important information in the following retrieve step and it also would be more convenient for users to figure out the answer because the short format.

Finally, the designed system will give each document an ordered number, which is described as document ID. This step is also important for the designed system because there is no specific ranking facility applied in the system. As we have mentioned in chapter one, the traditional Boolean retrieve model does not allow applying the weight, thus, to implement the ranking facility is impossible. However, there are some researchers trying to perfect the disadvantages of traditional Boolean Model such as the Fuzzy Set Model and Extended Boolean Model. These models provide different methods to add weight to the traditional Boolean Model and they provide the possibility to apply the ranking facility. For Fuzzy Set Model, it still has some problems because it will result in a wrong ranking order when the system requires solving the complex Boolean expression. For the Extended Boolean Model, it works like the combination of Vector Space Model and Boolean Model. Although these Models can provide the ranking facility, the traditional Boolean Model is more

efficient in the information retrieval procedure.

Thanks to the algorithm provided by Google, we can achieve the ranking number of the candidate documents, so that it solves the drawback for the traditional Boolean Model. Because the system only downloads the top ten webpages from Google, thus, there is only one ID for each webpage, which involve two numbers from one to ten. This is the ordered number for the webpages. Then, the system will use the number to define the short passages, which extract from the text information of the webpages. Because the information can be cut into thousands of pieces, the system will give a sub-ID to each document, which involves four more numbers. All those documents will be saved in the local directory.

There is one more important factor that should be taken into account: the storage space. With the consideration of the practicability, the system should not keep the retrieved information and documents because it uses the Internet resources, which are dynamic and changes every day. The retrieved results will not be used as new questions from users being asked on the system. The system does not provide the fixed database for users to search for answers because it a web-based Question Answering System. Thus, the designed system also provides a Folder cleaning module, which is used to clean the search results being used before. In other words, the system will clean the information in the local document as well as the folders every time such as the webpages and the document collection. This module keeps the system clean and

removes the redundant information.

2.5 Question processing and document indexing strategy

Since the system has prepared the document collection, the main procedures about information retrieval for the Boolean based Question Answering System will start. Before the documents being indexed, the system will process a module called question processing which is used to convert the questions asked by users into queries. Then, the system will use the queries identify and select the documents which are used as the candidate answer. The definition of queries and questions are different. The queries can be seemed as the key terms, which are contained by the question and the question processing module is used to remove the useless terms in the questions. The designed system applies the algorithms from C.J. van Rijsbergen and M.F. Porter. These two researchers provide the most famous algorithms being used today, which works perfect in English information retrieval system as well as the Question Answering System research area.

There are two steps in the question processing module. In the first step, the system will use a Stop list to move the high frequency, low frequency and unimportant words or terms in the questions.

As we know there are different parts of speech, which describe the characteristic or property of a word in English such as noun, verb or adjectives. Some of the words

contains the key words in a sentence such as noun. On the contrary, there are some words do not have any actual meaning but still be a part of the sentence such as articles, prepositions and conjunctions. These types of words are called empty words in English, which are indivisible parts of some specific sentences. The words such as "a" or "the" are included in those types. From the statistic reports, linguists also found that those words occur frequent than others in the sentence human use every day. The researchers in the linguistics research area and computers science research area find a list of words of those high frequency words. In the designed system, the high frequency words may occurs in almost all documents. So, if the questions are not being prepared in the question processing step, the high frequency words will be used as the index terms in the query. Consequently, all candidate documents may be retrieved back because the terms in the query are matched with the terms in the documents. Therefore, these index terms do not help the system to distinguish the candidate documents while process indexing step.

Conversely, for the very low frequency words, they may not always occurs in the candidate documents and some of them even only occur once among the candidate documents as well as the questions provided by users. These kinds of words always occur in some specific knowledge based documents. Accordingly, these words were seems as irrelevant terms and the system will remove these terms automatically from the index terms because these terms will retrieve nothing back from the document collection and they do not have any distributions for the retrieval procedures.

If these words being considered as the index terms, the index efficiency will decrease. Although these words are presented at a high frequency or low frequency in the documents, they do not involve any meaning in the questions as well as the retrieval steps. Therefore, these words or terms should be removed during indexing, and then the system will use significant terms to retrieve the candidate documents efficiently. This technique will help the system to save the storage space because the system does not have to keep and index the non-important terms. So, the following retrieval procedures will process more quickly due to the reduced data set and the efficiency of the Boolean based Question Answering System have been improved. The step has an excellent performance while the system face with large amount of data. A full list of English stop words provided by C.J. van Rijsbergen (C.J. van Rijsbergen [1979]).

The second step is called stemming. The same meaning index terms or the missing of index terms in the candidate document collection will decrease the efficiency of the designed system. In English, it happens frequently that some of the words share the common root. For instance, “drive”, “driving” and “driven” have the same meaning but different formats in different tense. If one of the documents in the candidate collection contains the word “drive”, but the query only contains “driving”, this document will not be retrieved back because the query terms are not matching with the terms in the document. Consequently, the system will miss some relevant document in the candidate collection. It may decrease the recall value, which stands

for the proportion of the retrieve relevant document in the existing relevant documents. Thus, to avoid these drawbacks, the designed system applied the stemming step in the question processing module.

There are many algorithms applied in this step such as Porter Stemming (Porter [1980]) and Lovins stemmer (Lovins [1968]). The designed system will use the most famous algorithm called Porter Stemming (Porter [1980]) to process the questions provided by users. This algorithm's performance is excellent. It is not only used in information retrieval research area and Question Answering research area, but also many other research areas in computer science. The researchers has applied it in their research area such as data mining research area or artificial intelligent research area. This algorithm provided by the researcher named M. F. Porter (Porter [1980]).

To be more specific, this algorithm is used to remove useless suffixes from the index terms being processed before and conflate different conflate terms into a single term. There are five steps and lists of suffixes used in the in the algorithm.

The major distribution for the algorithm is to reduce size and complexity of the data. Therefore, it makes the system worked fast and the method is very easy to be understood. On the contrary there are also some drawback factor should be taken in to consideration. From Porter Stemming algorithm, a word can be made up of both vowel and consonant. So, the researcher donates A as the consonant and B as the

vowel. Accordingly, a normalization formula of an English word is: [A] $(BA)^m$ [B] (Porter [1980]). There are five steps to remove the suffixes of a word. For each step, the suffixes of the words will convert into new suffixes. It may exist a condition before the step such as the power m in the formula may bigger than one or zero. In another word, if the condition is stratified, the word can be removed some letters from the original suffixes. The lists of suffixes in each step are providing in Porter Stemming algorithm (Porter [1980]). In Figure 3, the example is showing about how Porter Stemming algorithm works with a specific word.

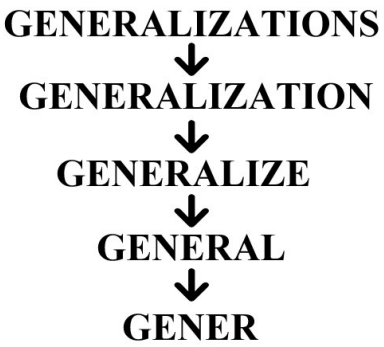

Figure 3

However, there still exist some defects for this method as well as all kinds of suffix stripping algorithms. For all kinds of stemming algorithm, it cannot be guaranteed that the method will reduce the vocabulary. Most of the time, the word shared the same root will have same meaning in English such as “car” and “cars”. Sometimes the words shared the same root may not holds same meaning such as “converse” and “conversely”. It is rare but still occurs in English. Thus, these kinds of words should not remove the suffixes. If these kinds of suffixes are being removed, it may completely change the original meaning of the term. So, the algorithm should be improved and suffix lists should be updated when some situations occur. Although

there are some defects, this algorithm still works well with removing around 30% duplicate meaning words.

For some of the information retrieval systems as well as Question Answering Systems, there is one more step applied called lemmatization. This step is different from the stemming. This method can convert the words into their original format. For example, the word “wrote” can be changed into “write”. This method can solve some of the problem in stemming step but the suffix stripping algorithm already performs very well. Therefore, it seems an optional step in the designed system and we do not implement it as a necessary processing step.

After the question processing system, the index terms in the query for matching documents is ready. As the document collection has been prepared in order before, the indexing step is starting.

The Boolean based Question Answering System will use the traditional Boolean retrieval Model to execute the index step. This method is easy to be described. The similarity is used to help the designed system to judge whether the document should be retrieved back or not. It is based on whether the documents are meeting the Boolean expression in the query. As a result, the system will first build the Boolean query, which extracts from the original index terms. A Boolean expression is the terms connected by Boolean operators. There are many types of Boolean operators and the most common used are AND, OR and NOT. Because the designed system is a

Question Answering System and the questions are provides by the normal users, thus, the Boolean operators in the query and the document may not have those complex Boolean operators such as NEAR. These operators are only used in some of the information retrieval systems and the skilled programmers or technician will use them. So, the designed system only provides the technique to solve the types of Boolean operators, which we have talked above. These operators may organize in different format in the question and may be very complex structure. The designed system will use two methods to build the query parser, which may use as the filter for the documents collection.

To prepare for the query parser, the system will use different statements to treat the different types of Boolean operators. Accordingly, the index terms connect by Boolean operators AND means that both of the terms should be included in the candidate document collection. The system will use add method in Java to add the Boolean Clause and state that both of the two terms must occur. Therefore, the index terms in the clause have to appear in the candidate document collection. The terms connect by Boolean operators OR means that at least one of the terms should be included in the candidate document. The system will use add method in Java to add the Boolean Clause and define one of the two terms as must occur and another as should occur. Therefore, one of the index terms in the clause has to be appeared in the candidate document collection and the other is optional appeared. The system will exchange the choice for the terms. The terms connect by Boolean operators NOT

means that the terms before NOT operator must be included in the candidate document, on the contrary the term after NOT operator must not be included in the candidate document. The system will use add method in Java to add the Boolean Clause and define the one of the terms before NOT operator as must occur and the term after NOT operator as must not occur. Therefore, the index terms before NOT operator in the clause has to be appeared in the candidate document collection and the other term will be never appeared in the document. This method builds the independent Boolean query or Boolean expression.

The system will also use the Query Parser to create the Boolean query. This method accepts different text formats. Therefore, the document collection can be indexed after the system build the complete Query Parser, this step will let the system to flit and retrieve the candidate document back which will be used as the candidate answer showing as a list in the answer section.

As the system has stored the document text information along with the ordered document numbers, thus, the index procedure is easy to be applied. After index step, the candidate answers for the users are prepared. Users can find the answers provided by the designed system directly in the answer frame. Each paragraph is generating in separate lines and in order so that users can easily view the answers from the top to the bottom. The details of implementation will be introduced later in this paper.

2.6 Question reformulation

To get a friendly user experience, provide better answers and improve the practicability, the designed system provides an additional procedure, which is called question reformulation step. Since these kinds of applications program may have some defects, programmers usually provide users a reformulation step to avoid some drawbacks come from the original steps. Thus, in order to satisfy the need from users, we apply this step.

As the system has provided all the answers being retrieved back, the users should find the correct answers from the first beginning to the end. If the users find the answers that they want, they can quit the application program by simply press the button “yes”. However, it could happen that the users still cannot give the answers that they expected after they have review all the answers provided by the designed system, they can press the button “no” which means they are not satisfied with the results. The system will enter in the question reformulation step.

At this time, the designed system will clear the question as well as the results, which has been retrieved before. When the webpages in the folder and the text information in the documents that includes the Google results document and the answer document have being removed, the empty documents and the folders in local are ready to receive the new information and results. Therefore, the system will allow the users to provide the new questions. The users would think about and organize a new question or they

can just simply remove some terms from the original question which they used before. As the designed system is provided to all kinds of users and they may not has the ability to organize acceptable questions such as children. Accordingly, the questions that contains too less terms or too many terms will generate bad retrieve results or even not accept by the designed system. More important, the Boolean based Question System is different than the normal Question Answering System. Whenever the users can provide a bad query, the system will not provide a reasonable answer to the user. Thus, in order to solve this problem, the system allows users to modify the questions according to their own opinions because the normal users are not sophisticated as the technicians who always use the system and has lots of experiences.

Basically, there are two types of question that can be solved by the designed system. The first and significant type of question that this paper mainly focuses on is the Boolean expression based questions. The designed system will provide the opportunity that changes the whole question or change part of it in this procedure. Because the traditional Boolean retrieved Model do not allow the weight adding in, so that the system cannot provide any better solution with it. As the different retrieval Model has mentioned in chapter one, there may be another way to improve the traditional Boolean retrieval system, but it is not focus on the original Boolean retrieval and it may stray off the main point. Thus, the designed system does not use any of those models in the experiment. The system will clean the old questions has saved in the document before and allow users to type a completely new question in the

new frame.

However, the system will accept the normal questions, which may not include the Boolean expression. In the related work section, there is an original Question Answering System has been mentioned and it used different retrieval model but only deal with the normal questions. For the normal types of question it is able to add weight on each terms in the question because the retrieval model is allowed. For instance, there is a question “Which countries speak both French and English”. Then, the system will provide the first solution to help users reformulate their question because for each term “French” and “English” are important. The system cannot give a different weight for them otherwise the system will generate a wrong answer or even miss retrieve some important information. When the users want to retrieve the question “Which countries speak French”. This type is the normal question which being defined early in this paragraph. At this time, the terms in the query can be added with weight and they can add an actual number from one to ten as the weight. For example, in this sentence, users may think the term “French” is the most important word in it. Thus, they can add ten as the weight for this word and add other number less than ten for other words. For this step, the system will not the clean the question being used last time, so that the user would see their original question during adding the weight and the new frame will not cove the question frame. It will be a convenient and easy way for users to figure out what happened.

After users give out a new Boolean expression base question, the systems will process the same steps what has been discussed before. In the other word, the system will firstly submit the question to Google search engine, and then generate the new Google results because the questions are totally new. Thus, the system will get different webpages as well as different document collection. Also, the system will provide the new answer list for the users.

For the normal questions, the users will find the system use the same questions to search their answers. However, as the system has added weight for the question, the retrieve results would be different. Similarly, the system will process the step, but the Google results here would be the same as before. So, the webpages and the candidate documents would be the same as before. But the retrieve results would be different since the users add the weight to their questions and the ranking order would be changed as well.

To make the whole procedure more complete and to generate better answers for users, the system will provide two options for the users. So, if the users need to find the questions containing the Boolean expression, they can just choose the first step. For the users to provide with the normal questions, the system will also allow them to process the first step and then process the second. Because the second step only can a little bit retrieve results, so that combine these two steps would be better. However, the users can only chose the second step if they really want to skip it.

At this point, the designed system would provide users more opportunities to find the question without exiting the program. The users can modify their questions several times by themselves until they find the answer, which satisfies them. According to the designed idea of the system, the users will get different answers as well as the different ranking order each time when they modify the questions. As we provide the system to the normal users and not only the professional technicians, the question reformulation step is necessary and practical when users cannot find the answer the first time. It performs well when users give a Boolean based question. Usually, the users will not need to use this step because the system can already achieve good retrieval results, which will be present in the evaluation section. Thus, the question reformulation step is the backup part of the designed system and system does not set any limiting times for the users to modify the questions. This step would not occupy the storage space since the system cleans the local folder and document each time.

Once the users find the answer in the answer box that they want, they can press the "yes" and the system will not switch to the question reformulation step and then the whole procedures of the system are finished.

CHAPTER 3

Implementation

This paper aims to improve the information retrieval strategy and analyze the Boolean retrieval Model. Thus, the Boolean based Question Answering System is implemented. The experiment will use the designed system to generate the objective results and analyze the data sets. The Boolean based Question Answering System is a web-based question answering system and Google search engine is an important tool for the system to find the candidate document collection. The users can ask Boolean expression based questions such as “Where is the location and population of Toronto ” and “What imaginary line is halfway between the North and South Poles”. The designed system will process the retrieval steps by using Boolean retrieval Model. Therefore, the system can improve the Internet retrieval resource eventually. Apart from the research purpose, normal users would get direct and better results using the Boolean based Question Answering System compare with only using search engines. In another work, the contribution and achievement of this research is to improve the search engine’s retrieve results and promote a better way for users to find information.

Before starting the experiment, we have to implement the Boolean based Question Answering System. We use Java to finish the program coding part and we chose NetBeans 8.0.1 as the Integrated Development Environment (IDE) (Boudreau [2002]).

NetBeans as one of the common and fully featured IDE not only support for Java but also PHP, Ruby, JavaScript, Groovy, Grails and C++ as well. It improves continuously to promote different versions and it also support for different operating systems. It is an open frame and provides an open source and extensible platform for web, enterprise, and desktop applications. These are the reason why lots of program developers chose NetBeans as the IDE when they design the projects. The application modular is consist by many Modular Software Components and these components are the Java Archive File which includes a group of Java classes. The designed system will present the procedures in different classes in the function package, which is clear and easy to test our different components step by step and figure out the error quickly. NetBeans also provide a Graphic User Interface (GUI) modeling utilities, which can simplify the development of User Interface. Thus, we also use NetBeans to design the User Interface of the Boolean based Question Answering System.

To successfully complete the designed system, we also apply different Java libraries such Jsoup. Jsoup[1] is a Java library that used as a HTML parser. It can parse the HTML from a URL and extract the text data from it. There are many elements in the HTML document such as content. Jsoup provides the selectors to analyze these information, remove the useless information and revise the information. Jsoup provides a strong functional API and it has scalability of solutions. The selectors in Jsoup can support different requirements for parse the HTML.

[1] http://jsoup.org/

Another technique tool has been used in the designed system is Lucene (McCandless, Hatcher and Gospodnetic [2010]). Lucene is a text retrieval engine package but not complete text retrieval engine because it provides the frame of retrieval procedures, complete search method, complete index method and parts of analysis method. Lucene provides a convenient API for the programmers, so it is easy to use the open source libraries in it. Not only have the programmers used it to structure the text retrieval application but also some commercial software such as the Web Sphere of IBM (Iyengar, Jessani, and Chilanti [2007]. Lucene is built by Java and has become a sophisticated open source project. It is very popular and provides free libraries that being used in information retrieval research area. There are some advantages of Lucene. Firstly, the format of the index files can be support separately from the application platform. Inother words, the index files can be shared by different operating systems or platforms. Secondly, it contains the original index method such as the inverted index and add the partition index which can index some small files. Thus, it optimizes the index procedure. Thirdly, it provides a perfect object oriented frame, which is easy for programmers to implement and add new functions. Fourthly, it has the text analysis interface which does not has any programing language and file format limitations. The indexer builds the index file by accepting tokens and users only has to realize the interface when they create new file format. Lastly, Lucene supports the different search queries such as Boolean search and Fuzzy search. So, our designed system is able to import Lucene's libraries. Basically, there are seven

packages that need to be imported: analysis，document，index，queryParser，search，store，util. Our designed system imports some of them to build our classes. Lucene also provide the algorithms we have to use in the system, so that we can change some of the coding and override the algorithms in the system. The good compatibility of Lucene also allows us to implement in the operating system as well as the IDE.

In the Boolean based Question Answering System, there are eleven classes in the function part. The different classes stand for different procedures and modules in the designed system. Among those classes, the formalQueryBuilder class is used to formalize the questions such as remove the space in questions. To formalize the questions before send questions to Google search engine is necessary because the Google's API would not recognize the normal questions with some undefined characters in it. The GoogleConnector class and GoogleRsults class is used to get the Google search results and save it into the local documents. The crawler class is used to crawl the links and corresponding information from each of the webpage, then save the information in to local documents as the candidate retrieval documents. The PorterStemAnalyzer class is used to extract and build the queries from the question provides by users. In this class, the Porter Stemming algorithm has being applied and the Stoplist also include in it. Thus, the unimportant words in the questions will be removed and terms in the queries will convert into a stander format, which will be used as the index terms in the following steps. The Searcher class, TextExtractor class and TextIndexer class has been applied the libraries in Lucene. These two classes

realize the index procedures by building the index directories and indexing files in the system.

For the User Interface part, the main page is showing in Figure 4.

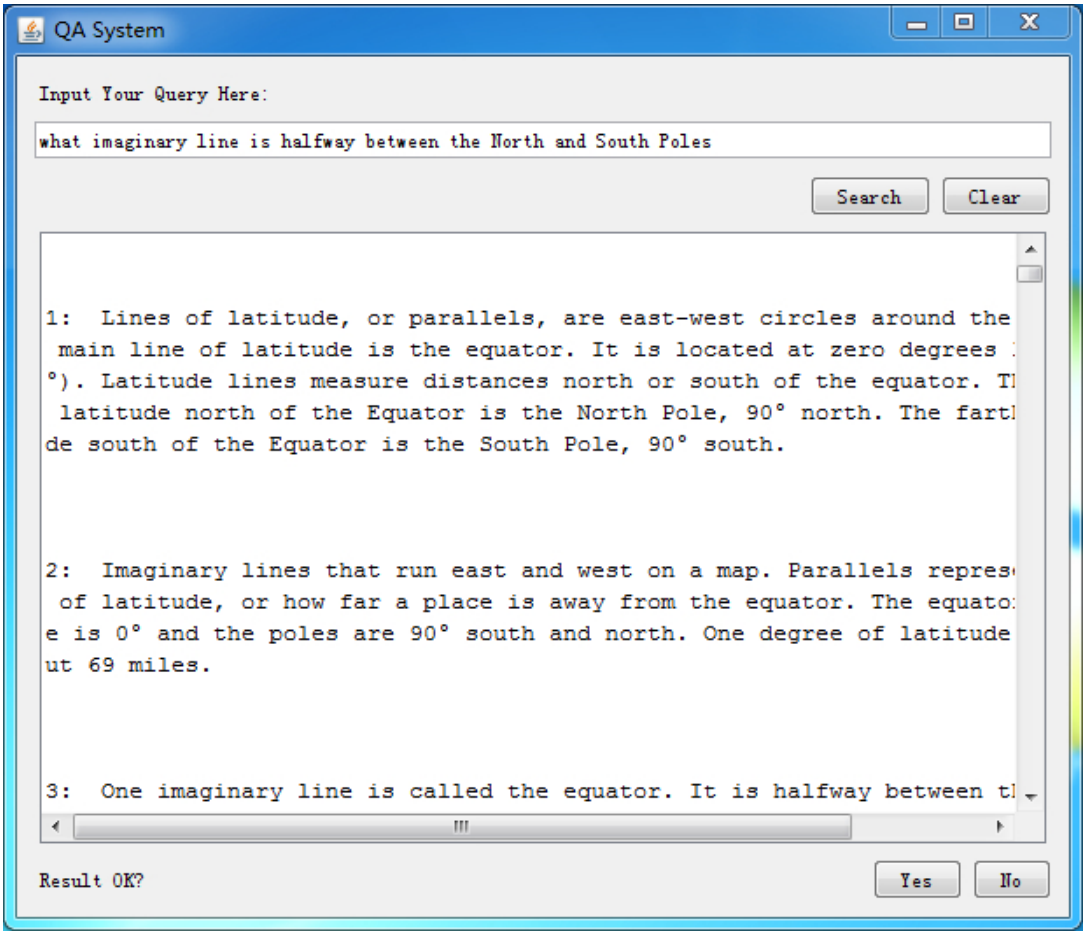

Figure 4

As it is shown in Figure 4, users can type their questions in the text box following the instruction. After they press search button the system will send the formalize questions to Google search engine. During the search procedures, if users find the question that they typed is not correct, the system will allow the users to change their question. They can simply press the stop button in the new frame, which is shown in Figure 5.

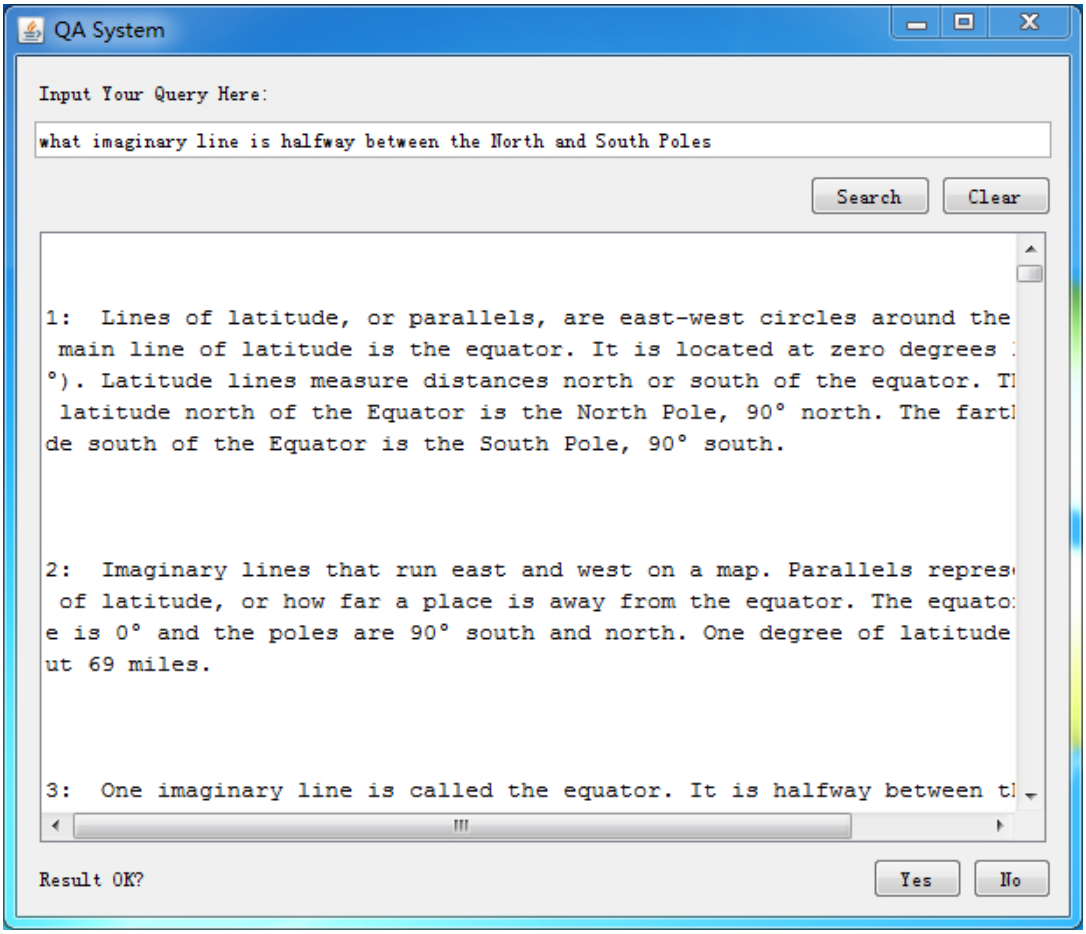

Figure 5

When the system gives back the answers, the users can find the answers in the answer box. The answers are organizing in an order and users can Scroll up and down to view all the answers being sending back. Whenever users find the answers they are satisfied with, they can press the “Yes” button at the end of the user interface. That means the users can quit the system. It is shown in Figure 6.

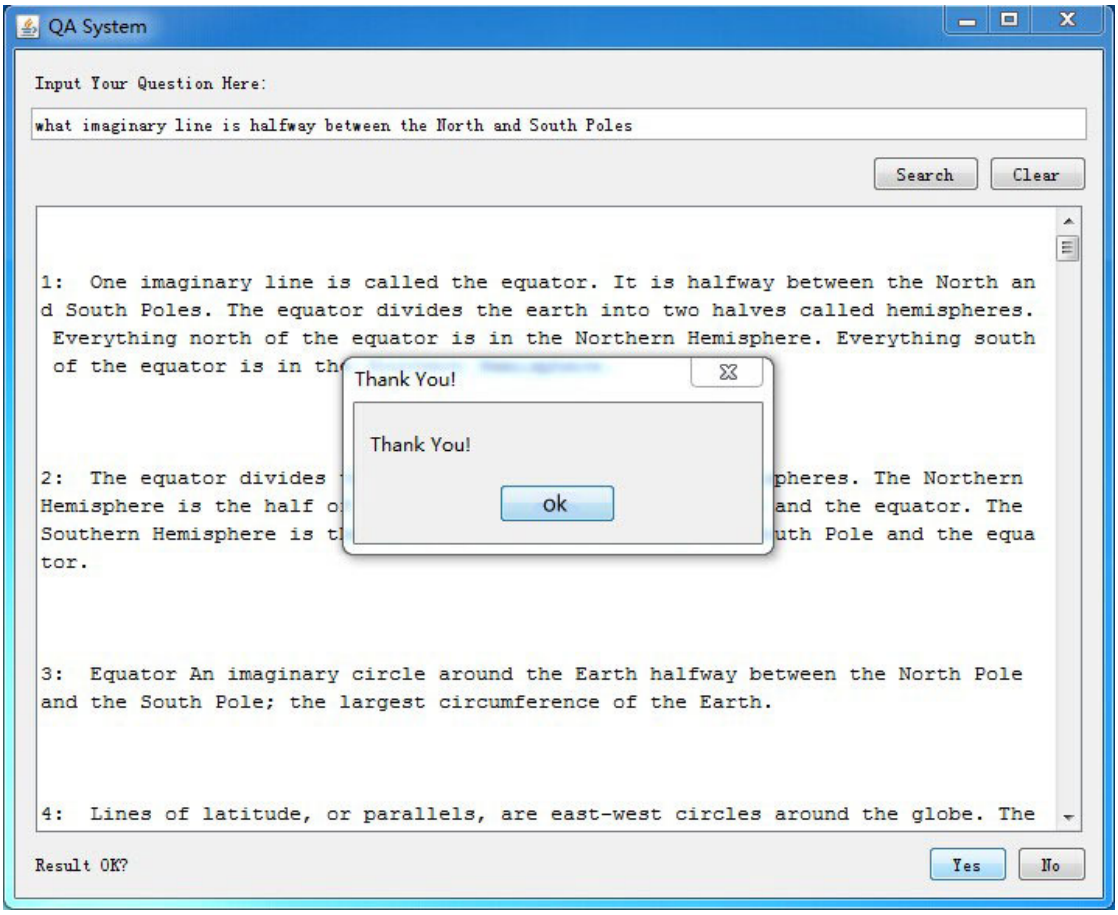

Figure 6

However, if users are not satisfied with the answers that provided by the designed system, they can press “No” button and the system will turn into the question reformulation module. As we have mentioned before, users can revise and add weights into the terms of the normal questions. The procedures are shown in Figure 7. The users can exit the system whenever they find the satisfied answers and it does not have the time limitation for their search action.

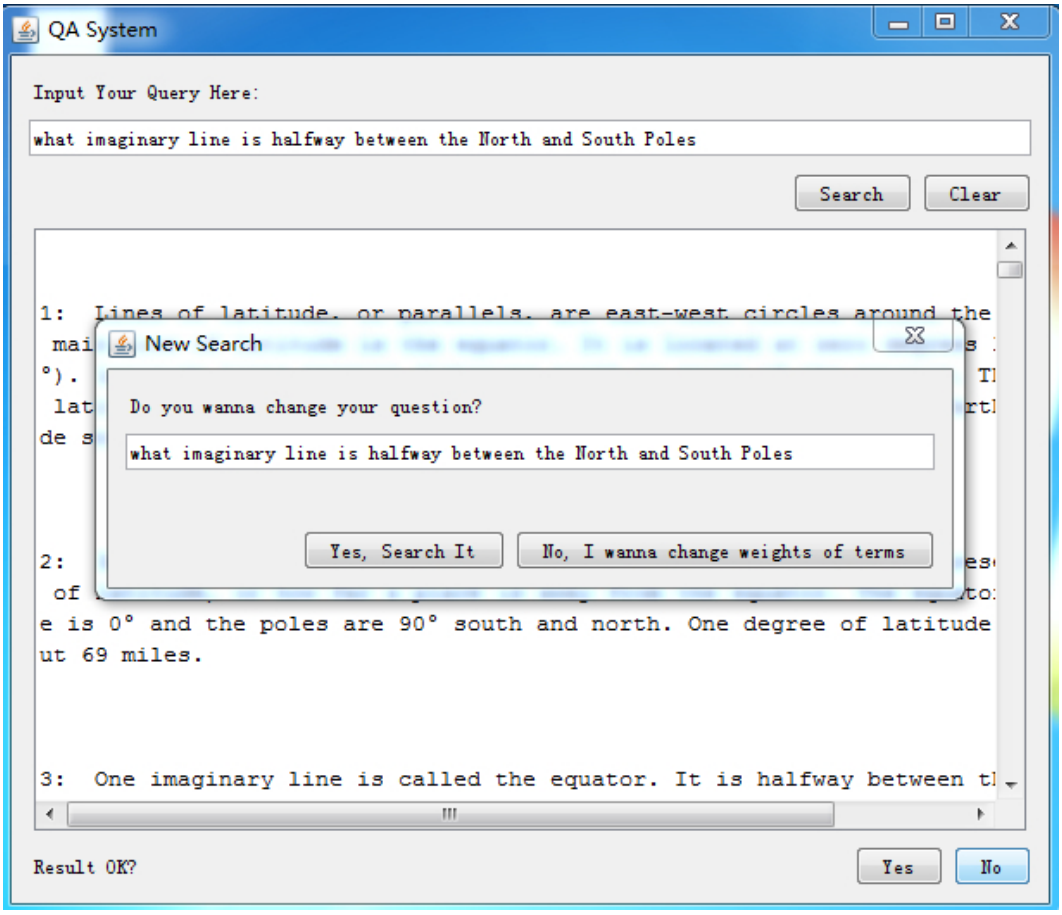
QA System
Input Your Query Here:
what imaginary line is halfway between the North and South Poles
Search
Clear
1: Lines of latitude, or parallels, are east-west circles around the
New Search
Do you wanna change your question?
what imaginary line is halfway between the North and South Poles
Yes, Search It
No, I wanna change weights of terms
e is 0° and the poles are 90° south and north. One degree of latitude
ut 69 miles.
3: One imaginary line is called the equator. It is halfway between t
Result OK?
Yes
No

Figure 7

CHAPTER 4

Evaluation

TREC (Text Retrieval Conference) is most popular and authoritative conference in Information retrieval research area. It was set up by DARPA (Defense Advanced Research Projects Agency) and NIST (National Institute of Standards and Technology). It was started from 1991 and many famous IT institutions and universities for academic research have attended such as Microsoft Research, Google, IBM Research, MIT, Stanford University, UCB, The National University of Singapore. The research of the conference divides into some specific areas such as Question Answering System Track, Legal Track, Medical Track, Robust Track, and Enterprise Track etc. Every year, TREC will provide the standard Corpus, Query Set and Evaluation method to the public. For the participant, TREC will collect the Runs, which are their results after running the standard data sets on their own system, and give each of them an evaluation results. TREC will holds a conference for academic exchange and announce the conference proceedings.

Accordingly, to evaluate the Boolean based Question Answering system, we used the data set from TREC[2]. TREC provides the data set which include the standard question sets and answer sets. Thus, we are able to use the question sets and run those questions on our system. Then, we can compare the results of our system with the standard answer sets. It is an objective method to evaluate the system and to analyze the

[2] http://trec.nist.gov/

running results.

As the purpose of the research is to analyze the Boolean Retrieval Model and there are different types of Boolean operators, we evaluate the Boolean operators separately which is more accurate and easy to analyze different retrieval results affected by different Boolean expressions in the questions. Also, we have evaluated the overall performance of the Boolean retrieval Model applied in the Question Answering System.

There are two main factors being taken into consideration of the retrieval results for Boolean based Question Answering System. Firstly, the ranking order of the retrieval results should be evaluated because it shows how the system working. Secondly, the number of indexing files also should be counted because different Boolean expressions presenting totally different results since the functions of different Boolean operators are different.

4.1 The evaluation results for AND operator

To evaluate the results for AND operator, we randomly selected 30 questions from TREC's data sets. These questions contain the Boolean operator "AND" randomly which means some of them may have only one AND operator but some of them may have multiple AND operators. During the experiment, we submitted the questions manually to the designed system and record the two factors' results. The ranking order

results of each question are shown in Figure 8.

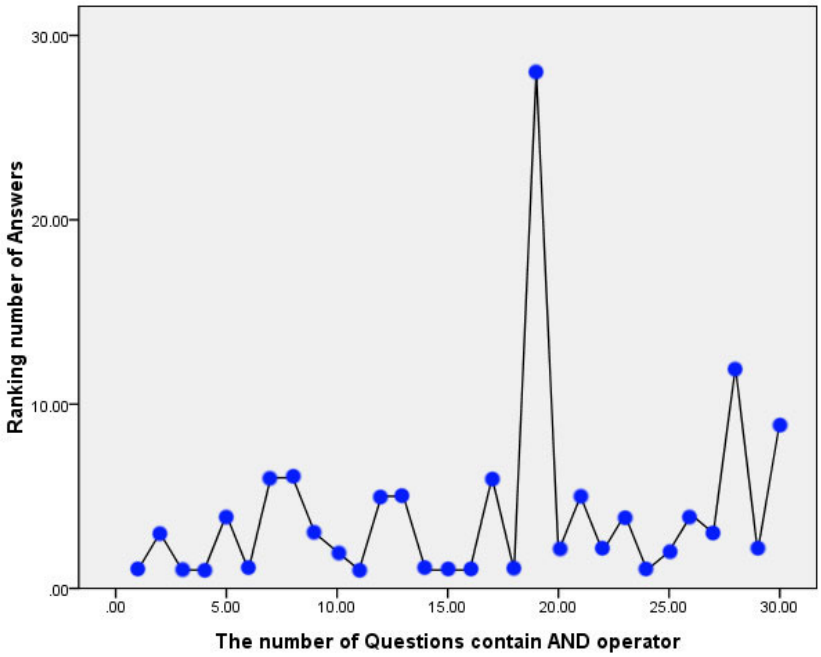

Figure 8

In Figure 8, the X-axis is the index number of 30 questions and the Y-axis represents the ranking order of each questions. It shows clearly that most of the questions rank well which is around the top five positions. In other words, users can easily find their answers in the top five answers when they type a question containing the AND operators. We also provide summaries for this case, which are shown in Table 1. It is easy to find the important value of the ranking order for AND operators.

Summaries for AND operators

Ranking number of Answers

N	Mean	Median	Minimum	Maximum	Range
30	4.1000	2.0000	1.00	28.00	27.00

Table 1

The numbers of indexing files results of each question are shown in Figure 9.

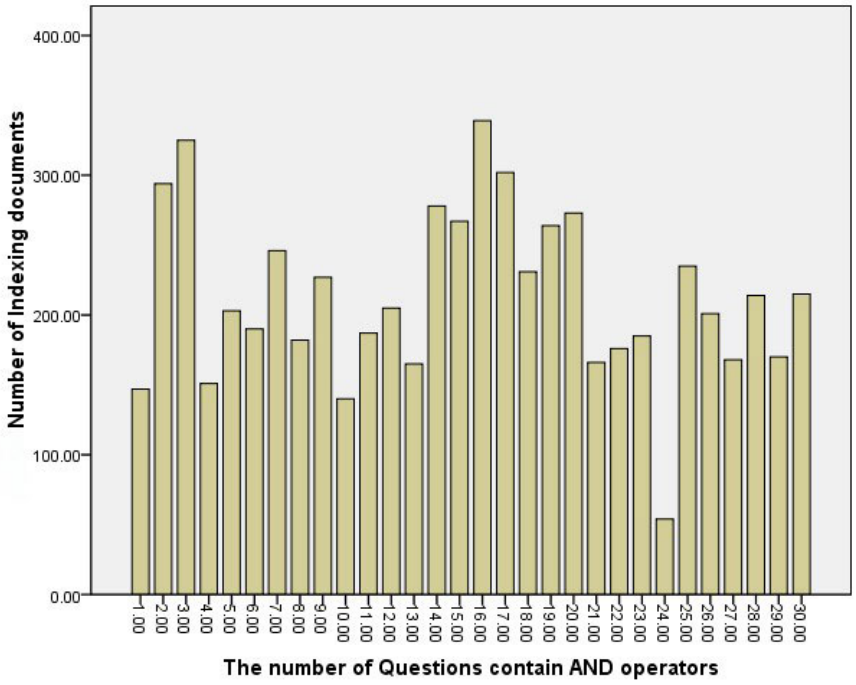

Figure 9

In Figure 9, the X-axis represents the index number of 30 questions and the Y-axis represents the number of indexing files of each questions. It shows generally about the number of indexing files. We also provide summaries in details for this case, which are shown in Table 2.

Summaries for AND operators

Number of Indexing documents

N	Mean	Median	Minimum	Maximum	Range
30	213.33	204.00	54.00	339.00	285.00

Table 2

From the Figure and Table are shown above, we can find that files being indexed is around 200 when users ask a question contains the AND operators. The minimum value is 54 which is the question containing two AND operators. From this particular result, we would conclude that the indexing files having decreased means the number of answers prepared by the system has decreased when the AND operator increased in

the question. In other words, the opportunities and possibility for users to find the correct answers are decreased when the question contains multiple AND operators. Thus, the relationship between the number of AND operators in the question and the number of indexing files presents an inversely proportional.

4.2 The evaluation results for OR operator

To evaluate the results for OR operator, we also randomly selected 30 questions from TREC's data sets. These questions contain the Boolean operator "OR" randomly. We submitted the selected questions manually to the designed system and recorded the results. The ranking order results of each question are shown in Figure 10.

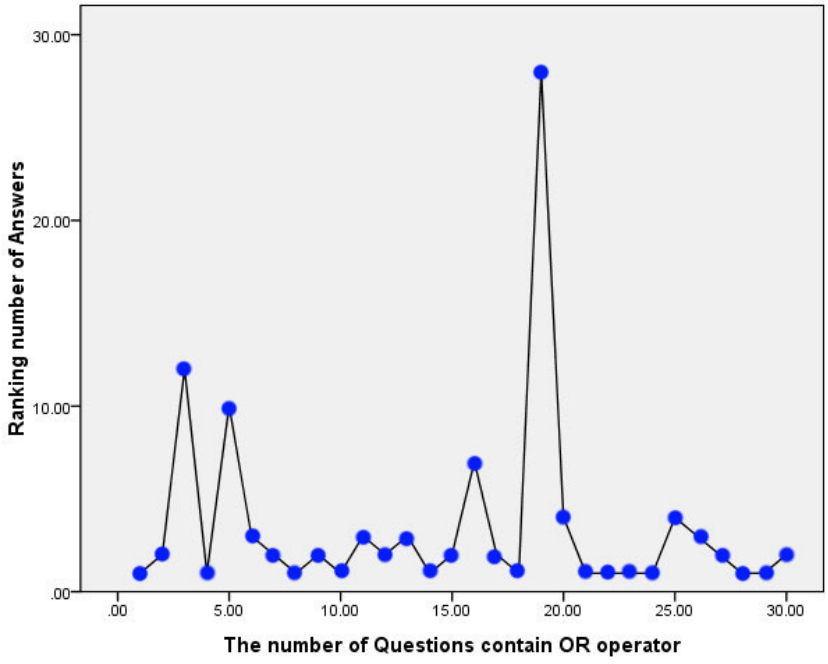

Figure 10

In Figure 10, the X-axis is the index number of 30 questions and the Y-axis represents the ranking order of each questions. We provide summaries in details for this case, which are shown in Table 3. It shows the results more specific.

Summaries for OR operators

Ranking number of Answers

N	Mean	Median	Minimum	Maximum	Range
30	3.5000	2.0000	1.00	28.00	27.00

Table 3

From the Figure and Table are shown above, we can find that the ranking order for OR operator achieved a better results than AND operator. Basically, the users can find their answers in top ranked four answers. The total performance of these two operators are roughly the same.

The numbers of indexing files results of each question are shown in Figure 11.

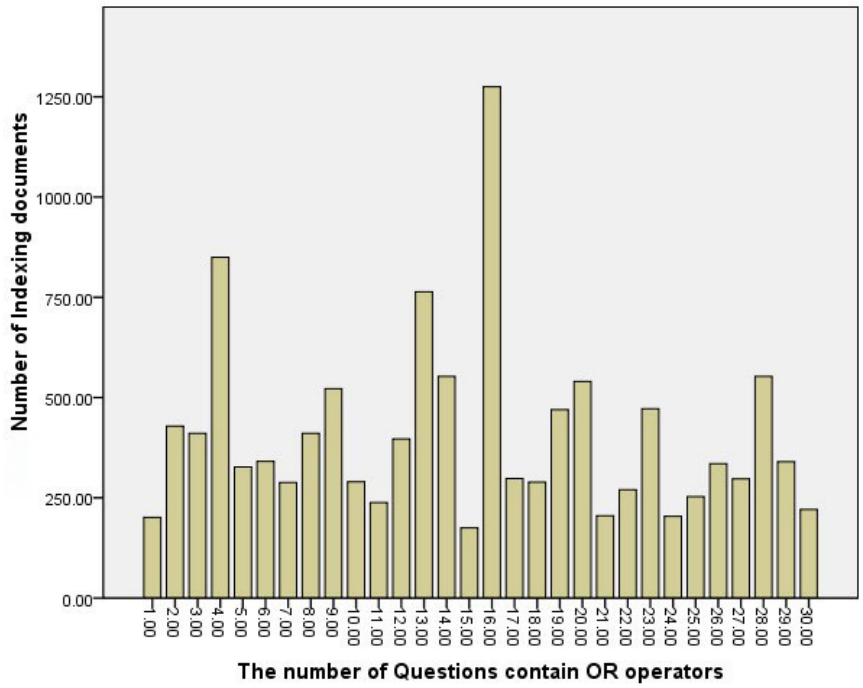

Figure 11

In Figure 11, the X-axis represents the index number of 30 questions and the Y-axis represents the number of indexing files of each questions. The summaries in details for this case are shown in Table 4.

Summaries for OR operators

Number of Indexing documents

N	Mean	Median	Minimum	Maximum	Range
30	407.30	337.00	175.00	1275.00	1100.00

Table 4

From the Figure and Table are shown above, we can find that the number of indexing files for OR operator is larger than AND operator. The average value is about 400. The maximum value is 1275 which is the question containing multiple OR operators. These results mean the indexing files have increased when the OR operators increased in the questions. In other words, the amount of the candidate answers become large when the OR operators increased and it is also hard for users to find the answers when the ranking order of the answer is not working well. It will also take a longer time while the system processes the index step.

4.3 The overall performance

In order to evaluate the overall performance of the Boolean based Question Answering System, we randomly selected 90 questions from TREC's data sets. These questions contain the Boolean operators such "AND", "AND…NOT", "OR…OR". We submitted the questions to the designed system and record the results. The ranking order results of each question are shown in Figure 12.

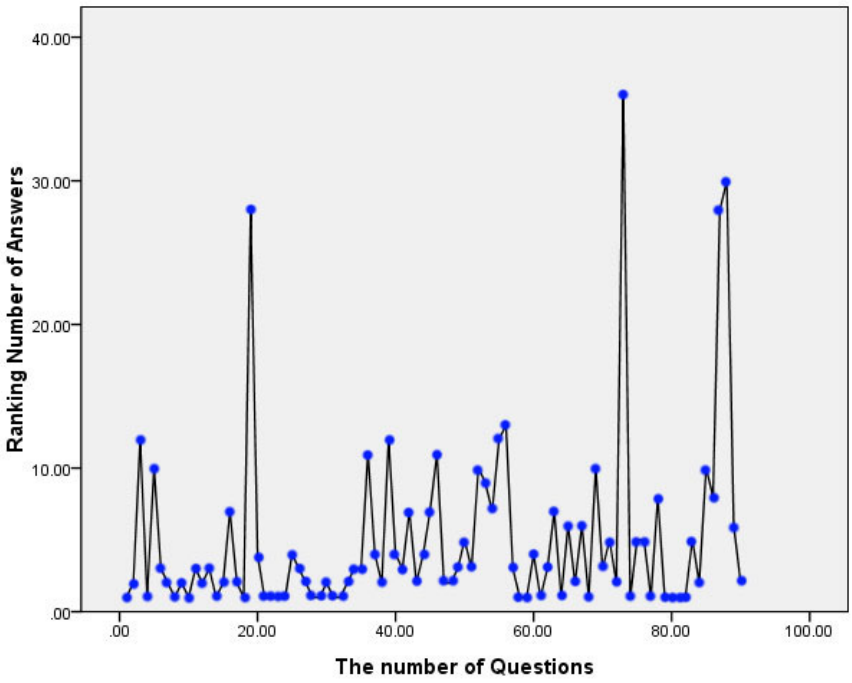

Figure 12

In Figure 12, the X-axis is the index number of 90 questions and the Y-axis represents the ranking order of each questions. The summaries in details for this case are shown in Table 5.

Summaries

Number of Indexing documents

N	Mean	Median	Minimum	Maximum	Range
90	5.0556	3.0000	1.00	36.00	35.00

Table 5

From the Figure and Table are shown above, we can find that most of the questions rank well which around top five positions. Thus, users will find the satisfied answer in the first five returned answers most of the times. In general, the system works well with the ranking functions for the Boolean based Question Answering System.

The numbers of indexing files results of each question are shown in Figure 13. See

next page.

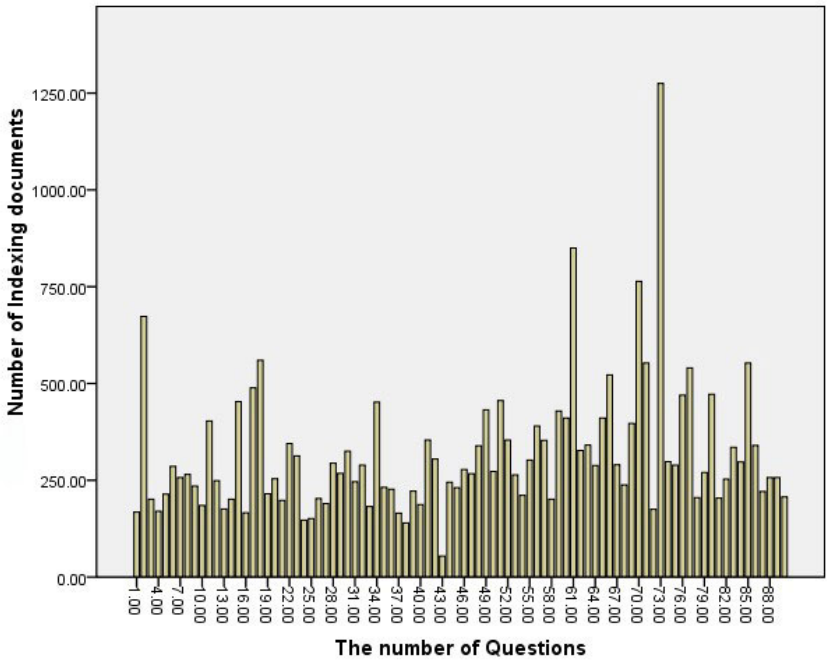

Figure 13

In Figure 13, the X-axis represents the index number of 90 questions and the Y-axis represents the number of indexing files of each questions. The summaries in details for this case are shown in Table 6.

Summaries

Number of Indexing documents

N	Mean	Median	Minimum	Maximum	Range
90	318.27	271.00	54.00	1275.00	1221.00

Table 6

From the Figure and Table are shown above, we can easily find that the average of the total number of the indexing files is around 300, which means the system will provide the users around 300 candidate documents for them to select the answers. The system is able to process the indexing step. Overall, the Boolean expression based questions work well in the Question Answering System.

CHAPTER 5

Conclusion

The purpose of our research is to improve the information retrieval on the Internet and analyze the Boolean Retrieval Model. Thus, we implemented the Boolean based Question Answering System. It use different techniques of information retrieval such as the Porter Stemming algorithm. It is a web-based Question Answering System, which does not contain any databases. Thus, Google search engine provides us the up to date data, which will be used as the candidate retrieval documents. The system uses the retrieval results from Google and continue to process the indexing step. The indexing step is structured as the Boolean Retrieval Model. Thus, the most significant improvement of the Boolean based Question Answering System is this system will provide users direct and correct answers rather than the links or webpages being provided by search engines such as Google. Besides, the Boolean based Question Answering System will use the Boolean Retrieval Model to analyze and index the candidate document prepared by Google search engine, which means the system optimizes the search results from Google search engine and improve the effectiveness of information retrieval.

In this paper, we used the Boolean based Question Answering System to evaluate the overall performance of Boolean operators and the two main types. The ranking order and the indexing files have been taken into consideration. After the experiment, the results show that most of the users would find the answers they want in the top five

results being returned by the designed system. For different Boolean operators, the ranking orders are almost the same and all of them are working well. The numbers of indexing files presents different results between different Boolean operators. For the AND operator, the indexing files decrease as the AND operator in the question is increasing. In our test data, the indexing files decrease about three fourths as one AND operator being added. Therefore, the candidate answers become insufficient as there are multiple AND operators in the question. However, the indexing files increase as the OR operator in the question is increasing. It will take a longer time for when the system processes the index step.

REFERENCES

AKUTSU, T., MIYANO, S. AND KUHARA, S. 1999. Identification of genetic networks from a small number of gene expression patterns under the Boolean network model. *Pacific Symposium on Biocomputing*, 17-28.

ALLAN, J., CALLAN, J. AND CROFT, B. 2002.*Challenges in Information Retrieval and Language Modeling: Report of a Workshop held at the Center for Intelligent Information Retrieval*, University of Massachusetts, Amherst, Massachusetts, USA.

BHARAT, K. AND MIHAILA, G. A. 2000. Hilltop: A search engine based on expert documents. *In Proc. of the 9th International WWW Conference (Poster)*, vol. 10.

BOOKSTEIN, A. 1980. Fuzzy Requests: An Approach to Weighted Boolean Searches. *American Society for Information Science.* 31(4), 240-247.

BORDOGNA, G., CARRARA, P. AND PASI, G. 1991. Query term weights as constraints in fuzzy information retrieval. *Information Processing & Management,* 27(1), 15-26.

BORDOGNA, G. AND PASI, G. 1993. A fuzzy linguistic approach generalizing Boolean Information Retrieval: A model and its evaluation. *Journal of the American Society for Information Science*, 44(2), 70–82.

BOUDREAU, T. 2002. NetBeans: the definitive guide. O'Reilly Media, Inc..

BUELL, D. A. 1982. An Analysis of some Fuzzy Subset Applications to Information Retrieval Systems. *Fuzzy Sets and Systems,* 7(1), 35-42.

BURKE R. D., HAMMOND, K. J. AND KULYUKIN, V. A. 1997. Question-Answering from Frequently-Asked Question Files: Experiences with the FAQ-Finder System. *Technical Report TR-97-05*, University of Chicago, Department of Computer Science.

CASTELLS, P., FERNANDEZ, M. AND VALLET, D. 2007. An Adaptation of the Vector-Space Model for Ontology-Based Information Retrieval. I*EEE Transactions on Knowledge and Data Engineering*, 19(2), 261-272.

CHAU, M. AND CHEN, H. 2003. Comparison of three vertical search spiders. *Computer,* 36(5), 56-62.

MANNING, K. D., RAGHAVAN, P. AND SCHUTZE, H. 2008. An introduction to information retrieval. Vol.1. Cambridge: Cambridge university press.

DEL, C., GIANNA, M., GULLI, A. AND ROMANI, F. 2005. Fast PageRank Computation via a Sparse Linear System. *Internet Math*. 2(3), 251-273.

GREEN, B., WOLF, A., CHOMSKY, C., AND LAUGHERY, K. 1986. BASEBALL: an automatic question answerer. *In: Readings in natural language processing*, Morgan Kaufmann Publishers Inc., 545-549

GULLI, A., AND SIGNORINI, A. 2004. The Indexable Web is More than 11.5 billion pages. *Poster proceedings of the 14th international conference on World Wide Web.*

HALLINAN, A. J. 1993. A review of the Weibull distribution. J*ournal of Quality Technology*, 25(2), 85-93.

HAVELIWALA, T. H. 2002. Topic-sensitive PageRank. *In Proceedings of the 11th international conference on World Wide Web (WWW '02). ACM,* New York, NY, USA, 517-526.

IYENGAR, A., JESSANI, V. AND CHILANTI, M. 2007. WebSphere business integration primer: Process server, BPEL, SCA, and SOA. IBM Press.

JIN, R., HAUPTMANN, A. G. AND ZHAI, C. 2002. Title language model for information retrieval. *In Proceeding: SIGIR '02 Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval,* 42-48.

JONES, K. S. AND WILLET, P. 1997, Readings in Information Retrieval. San Francisco: Morgan Kaufmann, ISBN 1-55860-454-4.

KANTOR, P. B. 1981. The logic of weighted queries. *IEEE Transaction on Systems Men and Cybernetics,* 11(12), 816-821.

KOBAYASHI, M. AND TAKEDA, K. 2000. Information retrieval on the Web. *ACM Computing Surveys*, 32(2), 144-173.

KRAFT, D. H. AND BUELL, D. A. 1983. Fuzzy sets and generalized Boolean retrieval systems. *International Journal of Man-Machine Studies,* 19(1), 45-56.

KRAFT, D. H., BORDOGNA, G. AND PASI, G. 1994. An extended fuzzy linguistic approach to generalize boolean information retrieval. *Information Sciences – Applications,* 2(3), 119-134.

LAFFERTY, J. AND ZHAI, C. 2001. Document Language Models, Query Models, and Risk Minimization for Information Retrieval. *In Proceeding: SIGIR '01 Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval,* 111-119.

LATIRI, C. C., YAHIA, S. B. AND CHEVALLET, J. P. 2003. Query expansion using fuzzy association rules between terms. *In Knowledge discovery and discrete mathematics. International conference*, 231-242.

LEE, J. H., KIM, W.Y., KIM, M. H. and LEE, Y. J. 1993. On the Evaluation of Boolean Operators in the Extended Boolean Retrieval Framework. *Proceeding: SIGIR '93 Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval,* 291-297.

LEE, J. H. 1995. Analyzing the Effectiveness of Extended Boolean Models in Information Retrieval. *Computer Science Technical Reports.* Cornell University Ithaca, NY, USA.

LIOMA, C. AND BLANCO, R. 2009. Part of Speech Based Term Weighting for Information Retrieval. *Advances in Information Retrieval,* Vol. 5478, 412-423.

LOVINS, J. B. 1968. Development of a stemming algorithm. MIT Information Processing Group, Electronic Systems Laboratory.

MCCANDLESS, M., HATCHER, E., AND GOSPODNETIC, O. 2010. Lucene in Action: Covers Apache Lucene 3.0. Manning Publications Co.

MILLER, D. R., LEEK, T., AND SCHWARTZ, M. R. 1999. A Hidden Markov Model Information Retrieval System. *In Proceeding: SIGIR '99 Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, 214-221.

MONZ, C. 2007. Model Tree Learning for Query Term Weighting in Question Answering. *Lecture Notes in Computer Science*, Vol. 4425, 589-596.

MORRISSEY, J. AND ZHAO, R. 2013. R/quest: A Question Answering System. *Flexible Query Answering Systems*, 79-90.

NA, S.H., KANG, I.S. AND ROH, J.E. 2007. An Empirical Study of Query Expansion and Cluster-Based Retrieval in Language Modeling Approach. *Information Processing and Management*, 302-314.

PAGE, L., BRIN, S., MOTWANI, R., AND WINOGRAD, T. 1999. The PageRank citation ranking: Bringing order to the web. *Technical report*, Stanford InfoLab, SIDL-WP-1999-0120.

PASI, G. 1999. A logical formulation of the Boolean model and of weighted Boolean models. *In Proceedings of the Workshop on Logical and Uncertainty Models for Information Systems,* London, UK, 1-11.

PATRO, S. AND MALHOTRA, V. 2005. Characteristics of the Boolean Web search query: Estimating success from characteristics. *In Proceedings of First International conference on WEB Information Systems and Technologies (WEBIST 2005)*, May 26-28, 2005, Miami, Florida.

PATRO, S., MALHOTRA, V., JOHNSON, D. 2007. An Algorithm to Use Feedback on Viewed Documents to Improve Web Query. *Web Information Systems and Technologies,* 177-189.

POHL, S., MOFFAT, A. and ZOBEL, J.. 2012. Efficient Extended Boolean Retrieval. Knowledge and Data Engineering, *IEEE Transactions on,* 24(6), 1014-1024.

PONTE, J. M. AND CROFT, W. B. 1998. A Language Modeling Approach to Information Retrieval. *In Proceeding: SIGIR '98 Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, 275-281.

PORTER, M. F. 1980. An algorithm for suffix stripping. *Program: electronic library and information systems,* 14(3), 130−137.

RICHARDSON, R. AND SMEATON, A. F. 1995. Using WordNet in a knowledge-based approach to information retrieval. *Technical Report CA-0395,* School of Computer Applications, Dublin City University, Dublin, Ireland.

ROGERS, I. 2002. The Google Pagerank algorithm and how it works. IPR Computing Ltd..

SALTON, G. AND MCGILL, M. J. 1984. *Introduction to modern information Retrieval*. New York: McGraw-Hill.

SALTON, G. AND WU, H. 1980. A term weighting model based on utility theory. *Proceeding: SIGIR '80 Proceedings of the 3rd annual ACM conference on Research and development in information retrieval,* 9-22.

VAN RIJSBERGEN, C. J.1979. *Information Retrieval.* London: Butterworths.

VAN RIJSBERGEN, C. J., ROBERTSON, S. E. AND PORTER, M. F. 1980. New models in probabilistic information retrieval. London: British Library. (British Library Research and Development Report, no. 5587).

VICEDO, J.L. AND FERRANSEZ, A. 2000. A semantic approach to Question Answering systems. *Ninth Text REtrieval Conference (TREC-9)*. Gaithersburg,

MD. November 13-16.

VIGNESH, U. AND SIVAKUMAR, M. 2013. Implementing High Performance Retrieval Process by Max-Score Ranking. *Journal of Computer Engineering (IOSR-JCE),* 8(5), 28-33.

VOORHEES, E. M, AND HARMAN, D. K. 2005. *TREC: Experiment and Evaluation in Information Retrieval.* Boston: The MIT Press.

WANG, Y. 1989. On the mathematical model of information retrieval. *Seventh National Conference Paper machine inspection, Intelligence Journal,* Shanghai, Vol. 8.

WEIBULL, W. 1951. A statistical distribution function of wide applicability. Journal of Applied Mechanics, 28(4), 613-617.

WONG, S. K. M., ZIARKO, W., RAGHAVAN, V. V., WONG, P. C.N. 1986. On extending the vector space model for Boolean query processing. *Proceeding: SIGIR '86 Proceedings of the 9th annual international ACM SIGIR conference on Research and development in information retrieval.* 175-185.

WU, H. C., LUK, R. W. P., WONG, K. F., KWOK, K. L. 2008. Interpreting TF-IDF term weights as making relevance decisions. *ACM Transactions on Information Systems (TOIS)*, 26(3).

XING, W., AND GHORBANI, A. 2004. Weighted pagerank algorithm. In Communication Networks and Services Research. *Proceedings. Second Annual Conference. IEEE*. 305-314

ZHAO, R. 2013. Improving Retrieval of Information from the Internet. Electronic Theses and Dissertations. Paper 4766.

APPENDICES

Appendix A

Testing question collection from TREC

TREC-8(1999)	What percentage of the world's plant and animal species can be found in the Amazon forests?
TREC-8(1999)	When did Spain and Korea start ambassadorial relations?
TREC-8(1999)	How large is the Arctic refuge to preserve unique wildlife and wilderness value on Alaska's north coast?
TREC-8(1999)	What is the fare cost for the round trip between New York and London on Concorde?
TREC-8(1999)	What is the name of the rare neurological disease with symptoms such as: involuntary movements (tics), swearing, and incoherent vocalizations (grunts, shouts, etc.)?
TREC-8(1999)	When did Israel begin turning the Gaza Strip and Jericho over to the PLO?
TREC-8(1999)	What is the name of the "female" counterpart to El Nino, which results in cooling temperatures and very dry weather?
TREC-8(1999)	What is the name for the technique of growing certain plants in soils contaminated with toxic metals, wherein the plants take up the toxic metals, are harvested, and the metals recovered for recycling?
TREC-8(1999)	How many people did the United Nations commit to help restore order and distribute humanitarian relief in Somalia in September 1992?
TREC-8(1999)	What is the length of border between the Ukraine and Russia?
TREC-8(1999)	When was General Manuel Noriega ousted as the leader of Panama and turned over to U.S. authorities?
TREC-8(1999)	What was the first Gilbert and Sullivan opera?
TREC-9(2000)	What was the name of the famous battle in 1836 between Texas and Mexico?
TREC-9(2000)	What are John C. Calhoun and Henry Clay known as?
TREC-9(2000)	What is the exchange rate between England and the U.S.?
TREC-9(2000)	When did Princess Diana and Prince Charles get married?
TREC-9(2000)	Who was Samuel Johnsons's friend and biographer?
TREC-9(2000)	What was the name of the movie that starred Sharon Stone and Arnold Schwarzenegger?
TREC-9(2000)	When did the royal wedding of Prince Andrew and Fergie take place?
TREC-9(2000)	What are Cushman and Wakefield known for?

TREC-9(2000)	Who wrote "The Pit and the Pendulum"?
TREC-9(2000)	What city houses the U.S. headquarters of Procter and Gamble?
TREC-9(2000)	What film or films has Jude Law appeared in?
TREC-2001	What imaginary line is halfway between the North and South Poles?
TREC-2001	What river flows between Fargo, North Dakota and Moorhead, Minnesota?
TREC-2001	What day and month did John Lennon die?
TREC-2001	What did Edward Binney and Howard Smith invent in 1903?
TREC-2001	What is the difference between AM radio stations and FM radio stations?
TREC-2001	What is the conversion rate between dollars and pounds?
TREC-2001	What is foot and mouth disease?
TREC-2001	What is bangers and mash?
TREC-2001	What do you call a word that is spelled the same backwards and forwards?
TREC-2001	What U.S. state's motto is "Live free or Die"?
TREC-2001	Name 4 countries which sanction human slavery or bonded labor.
TREC-2001	Name 4 people from Massachusetts that were candidates for president or vice-president.
TREC-2003	Name recipients of funds given by the various foundations of Bill and Melinda Gates.
TREC-2003	Which past and present NFL players have the last name of Johnson?
TREC-2003	What governments still officially recognize and support International Labor Day?
TREC-2003	List art museums that have returned looted Nazi art to their owners or descendants.
TREC-2003	What cities in Russia or the Commonwealth of Independent States (CIS) have statues of Lenin?
TREC-2003	List Hezbollah members killed or apprehended by Israeli forces.
TREC-2003	Which U.S. government employees (either military or civilian) have been accused of spying for foreign countries?
TREC-2003	What countries provided troops or support to the NATO-led peacekeeping mission in Bosnia?
TREC-2003	What cruise ships have sunk, caught fire, or have been beached?
TREC-2003	List female astronauts or cosmonauts.
TREC-2003	What player on a basketball team usually plays the post or pivot position?
TREC-2003	What artist recorded the song "At Last" in the 40's or 50's?
TREC-2003	Which U.S. government employees (either military or civilian) have been accused of spying for foreign countries?
TREC-2004	List members of the Berkman Center for Internet and Society
TREC-2004	What prizes or awards has Frank Gehry won?
TREC-2004	Which Italian city is home to the Cathedral of Santa Maria del Fiore

	or the Duomo?
TREC-2004	Who is its top official or CEO for AARP?
TREC-2004	Who is the president or chief executive of Amtrak?
TREC-2004	What does the Crips mean or come from?
TREC-2005	List other horses that have won the Kentucky Derby and Preakness but not the Belmont
TREC-2005	Provide a list of names or identifications given to meteorites.
TREC-2005	What international leaders sent or gave congratulations?
TREC-2005	How many soldiers or police officers were used to carry out the evacuation?
TREC-2005	What countries participated in this "Food-for-Oil" agreement by providing food or medicine?
TREC-2006	What evidence is there for transport of military equipment and weaponry from South Africa to Pakistan?
TREC-2006	What financial relationships exist between drug companies and universities?
TREC-2006	What familial ties exist between dinosaurs and birds?
TREC-2006	What financial relationships exist between the Israeli government and the Palestinian National Authority (PNA)?
TREC-2006	What financial relationships exist between Greece and Cyprus?
TREC-2006	What financial relationships exist between the United States and supporters of the Irish Republican movement?
TREC-2006	What common interests exist between the Abu Sayyaf and MILF (Moro Islamic Liberation Front)?
TREC-2006	Is there evidence to support the involvement of Christopher M. Davidge in illegal price fixing by the auction houses Christie's and Sotheby's?
TREC-2006	Name past and present LPGA commissioners.
TREC-2006	In what cities were the matches between Deep Blue and Kasparov held?
TREC-2006	What national leaders and spokespersons sent congratulatory messages following Thabo Mbeki's election as president of South Africa?
TREC-2006	Which European Union countries originally chose not to adopt the Euro?
TREC-2006	What effect do psychological or emotional problems have on obesity?
TREC-2006	What effect does second-hand smoke have on non-smokers?
TREC-2006	What is the position of John McCain with respect to the Moral Majority or the Christian Coalition?
TREC-2006	What is the position of the United States with respect to BSE (Bovine Spongiform Encephalopathy, or "Mad Cow Disease")?
TREC-2006	In what cities or towns have illegal methamphetamine labs been

	found?
TREC-2006	What charities have benefited from the sale or auction of his paintings?
TREC-2006	What are the locations or names of other stone circles in the UK?
TREC-2006	Name famous artists whose works have been purchased by Stephen Wynn or are displayed in his galleries.
TREC-2007	What financial relationships exist between Google and its advertisers?
TREC-2007	What individuals with professional experience in medicine or ethics commented unfavorably on the procedure?
TREC-2007	What financial relationships exist between DARPA and BBN?
TREC-2007	What common interests exist between President Bush and Bono, the U2 Rock Star?
TREC-2007	What financial relationships exist between the Chinese government and the Cuban government?
TREC-2007	What financial relationships exist between Syria and Iran?
TREC-2007	What common interests exist between Yo Yo Ma and Itzhak Perlman?

VITA AUCTORIS

NAME:	Jiayi Wu
PLACE OF BIRTH:	Chongqing, China
YEAR OF BIRTH:	1987
EDUCATION:	Capital Normal University, B.Eng., in Software Engineering, Beijing, China, 2011
	University of Windsor, M.M., in Management, Windsor, ON, 2013
	University of Windsor, M.Sc., in Computer Science, Windsor, ON, 2015