A Semantic Web based Meta Search Engine

MPhil Thesis

Siyan Li

Centre for Creative Computing
Bath Spa University
2017

Declaration

I declare that the work described in this thesis was originally carried out by me during the period of registration for the degree of Doctor of Philosophy at Bath Spa University, U.K., from October 2013 to December 2016. It is submitted for the degree of Master of Philosophy at Bath Spa University.

Acknowledgements

First of all, I would like to express my sincerest respect and deepest gratitude to my supervisor, Professor Hongji Yang, for supporting me to pursue my ideas in this study. Professor Yang has always been patient, encouraging, and supportive at each stage of my study. In research perspective, his kindly guidance open up for me a new world and new expectations.

I would like to thank my second supervisor, Dr. Jerry Fishenden, for his support and feedback on the project development and thesis writing. I would like to thank to the Graduate School Office at Bath Spa University for the helpful management.

Last but not least, I dedicate this thesis to my parents, Hong and Xiaoyan. I could not achieve what I have achieved without their persistent, unreserved love.

Abstract

With the rapid development of economy and technology, people who are influenced by the era pay more attention to the Internet.

Creative computing is a relatively new research area, for enhancing human creativity through requiring computing itself to be creative. It involves the Internet, web language, art, humanities and other areas. It motivates creative thinking during the process of study.

The aim of my research is to develop a targeted and personalised a meta search engine (MSE) based on the semantic web technology and at the same time, to explore whether it will resolve problems and create applied value for a group of people.

The semantic web integrates Internet into an integral database by adding meta data which can be acknowledged by computer. By adding meta data, all content will be re-graded into data by document unit. As a result, data will cover meanings in different semantics and can generate certain logical relationship.

On account of this concept, after receiving searching needs, Meta Search Engine hunts on multiple search engines and return result to users. It mainly focuses on improving searching speed, intelligently handling of search result and personalising searching function. What is more, recall ratio and precision ratio is high with increasing friendliness on users' retrieval surface. The author can personally provide targeted search of a restricted field by compiling algorithm of metadata search engine

Compared with the present search engines, the Writer's Portal that I am building is more personalised. Writer's Portal can provide the writers with more information on funding opportunities and scholarship support, etc., in order that writer could find a variety of aids on finance. Therefore, this is a very practical and a focused project. It can also fully show the characteristics of accuracy and personalisation of the meta search engines.

List of Figures

Table of Contents

Chapter 1. Introduction and Motivation

Objectives

- To observe the need for a Meta Search Engine.
- To discuss research questions and develop research questions.
- To highlight original contributions and define the measure of success.
- To outline the organisation of the thesis.

1.1 Overview of Problem and Proposed Research

Creative computing is a relatively new research area that aims to enhance human creativity through requiring computing itself to be creative. It involves the combination of knowledge and technologies from various areas, such as computing, humanities, art and so on. It motivates creative thinking during the process of study. This thesis is to develop an MSE (Meta Search Engine) based on the semantic web technologies and at the same time, to explore whether it will resolve problems and create practical value for a specific group of people, i.e., writers. The practicality of creative calculation in life is proved in this way.

Meta search engine is a search tool aiming to help users to choose and use appropriate search engines, even simultaneously, from multiple search engines through a uniform user interface to realise retrieval operation, and it is a global control mechanism of various retrieval tools distributed on the network. Meta

search engine, as an auxiliary retrieval tool, makes up the shortcomings of traditional search engines and it has many advantages compared to traditional search engines.

In order to improve acknowledgment and acquisition of objects, people need to receive different information of daily life and work tasks. Some of our affairs are based on interests. For example, to find a long-lost friend, one may turn to social networks like Facebook. Some may aim at acquiring knowledge. For example, to know more about management, one may think of *On Management* written by Steven P. Robbins, even though there are many ways for obtaining those resources. The most convenient and common method is surfing the Internet by clicking frequently used search engine and inserting keywords. Search engine refers to the system, which provides users with retrieval service and the relevant retrieval information by collecting information from the Internet according to certain strategy and using specific computer program, and organising and processing the information.

However, limited by searching space and accuracy of common search engines, people sometimes cannot accurately find what they need to know through search engines. One reason is that search engines do not take account of the different behavioural habits of each individual. Meta search engines based on the semantic web concept try to resolve this problem. In the process of research, the following some obvious questions are inevitable: what is a meta search engine; what is semantic web; and how is creative computing manifested in MSE?

In order to answer the above questions, the following chapters will give detailed explanations, prove them one by one after refining them, answer them by editing a meta search engine, and prove the roles of creative calculation in meta search engine and the practicability of creative calculation in real life.

1.2 Research Objectives

Presently search engines have been widely applied. However, they often fail to fully meet the needs of people who have growing pursuit of quality of life due to the limit of search scope and precision of search engines.

In order to solve the inconvenience caused by the search engines, the study describes the ways to solve the above problems through construction of a meta search engine. To meet the specific requirements, the study proposes a new idea that reflects the uniqueness of meta search engine. In the process of design of meta-search engine, specific search result screening is designed through the concept of semantic web and creative algorithms. By limiting the search contents to a specific area, the personalisation and humanisation of search engine are reflected.

The aim of my research is to develop an MSE based on the semantic web technologies, at the same time, to explore whether it will resolve problems and create practical value for writers. The core contribution is that the purposed mata search engine can provide the writers with more information on paying opportunities, funding opportunities, and scholarship support submissions, etc.

In order to achieve the research aim above, there are several objectives that need to be accomplished. Meanwhile, it briefly summarises the up-to date progress of each sub-objective.

The web page creation scheme of creative meta search engine must meet particular users' demands. In the project, the web page scheme aims to offer writers with a variety of economic aid and opportunities to work. After the plan is confirmed, its practicality is evaluated.

The completion standard of web page is defined and the roles of concept of semantic web in the project are studied and confirmed.

A set of creative algorithms that meet the above proposals and reflect the roles of semantic web are defined.

A prototype meta search engine is then developed.

Overall, the proposed research is to investigate the logical basis for improving creativity of meta search engines, especially the improvement of the innovation aspects of such system via the proposed rules.

1.3 Research Questions and Hypotheses

The research question in this study has been formulated as follow:

How to develop a meta search engine by using creative computing and the semantic web? The research work described in this thesis aims to address this research question effectively. In order to achieve this aim, a set of manageable and tractable sub-questions is defined that address the problem in details.

RQ1: What is a meta search engine with an innovative approach?

What is the basic structure of setting up meta-search engine?

How can people make web pages original and unique?

How to define new ideas and creativity?

RQ2: What is the proper development approach to achieve the requirement of creativity in the proposed MSE？

What is the framework of the proposed Meta Search Engine?

What are the main phases of the proposed Meta Search Engine?

What are the fundamental methods and technologies in the proposed Meta Search Engine?

RQ3: How to get and manage relevant knowledge?

Which technologies can be used to support knowledge mining and management?

RQ4: How can people reasonably use semantic web knowledge into the creation of a search engine?

How is semantic web applied in the existing search engine? Can people use the application of semantic web in other search engines when creating meta search engine?

The fundamental proposition of this thesis is:

The creative computing can be based on the creative algorithm derived from semantic web, and make the resultant algorithms for to form an innovative meta search engine.

Whether the above is true can be tested from many aspects including deployment and decomposition of creative computing and semantic web. Another two propositions are derived as follows:

RP1: The algorithm of proposed meta search engine is the specific form of expression of creative computing in semantic web. Its originality is the standard of the detection result of the reasonable use of creative computing in this project and also the standard of testing whether the proposition holds up.

RP2: Creative computing and relevant techniques can support programming creative meta search engine. To conform to the expected demonstration is the standard of judging the proposition holds up.

In addition to the above several propositions, rational concept, logic, technology and knowledge can be used to solve the sub-problems earlier mentioned. To achieve the goal, the project tries to meet the requirements by means of creative calculation, semantic web, art and creative progress. Finally, through reasonable transformation and combination of the various areas of knowledge, the particular the field of meta search engine that provides writers with economic support solutions is created. Besides, the prototype of meta search engine with creative algorithm is experimented with.

1.4 Choosing Research Methods

Before conducting the project, I need to choose a research method. Creative computing is a subject based on many basic knowledge concepts to reach new knowledge possibilities. The essence of this subject is based on that some research methods of these subjects can be applied in creative computing. My experiment is designed to establish a more innovative MSE. I select three research methods to start with according to this subject and end up with the most appropriate research method. Following are implications of the three methods and the examples applied in this project during the process of research as well as the statement of grounds for selecting this method and rejecting other methods.

1.4.1 Empirical Research

Empirical research is practicable by using evidence to test a resolution method. Based on the semantic web concept [1], MSE realises search by general search engines. There are several existing cases in this field, such as InfoSpace, Dogpile, Vivisimo and so on. It is undoubted that I can quickly get many useful experiences and find their weaknesses by researching existing MSEs. At the same time, with the help of published research results, I can implement more a specialised MSE by changing algorithms when building MSEs. Nonetheless, as mentioned in the objective, there are some limitations to learn the strengths of meta-search engines and develop new meta-search engines based on the knowledge of existing meta-search engines and the World Wide Web in the research. Although the existing

meta-search engines are being experimented with, it is very difficult to identify problems and work out a meta search engine while completely relying on the established studies.

1.4.2. Prototype Development and Case Studies

Having studied how general MSEs are working, I will develop a method to construct an MSE and actually develop a prototype of an MSE, with the following three steps. The first step is to master the needed programming languages, such as HTML and JavaScript. The second step is to develop semantic web algorithms for constructing an MSE, where these algorithms can make an MSE creative and practical. The third step will be actually coding and testing of the prototype. Afterwards, experiments will be carried out as cases studies to demonstrate that the prototype can validate the proposed MSE constructing method.

1.4.3. Secondary Research

Secondary research is a process including summary, collation and/or synthesis of existing research [3]. By the way of summarising and sorting the existing research results to create a specific plan in order to keep the whole project running more specifically is the research method of secondary research. But first, the semantic web and MSEs are still developing and not with many established research results. Second, the research direction of this project prefers to technical research rather

than theoretical research. As a result, the research methods of secondary research and empirical research do not apply to this project. Thus, I choose the case study as my research method.

1.4.4. Success Criteria

The standard of determining whether the research methods of creative meta search engine is successful is that the meta-search engine can obtain more desired searching results in specific areas. The following questions can be used as the criteria to judge whether of the project successes:

- How many domains the proposed meta search engine is suitable to be applied to?
- How to evaluate creativity of meta search engine according to designed algorithms?
- How to determine the meta search engine in the project is different from the existing search engine in a specific field in terms of functions?
- How many creative techniques of reference are applied in designing meta search engine?

1.5 Contributions

Creative algorithms and a meta search engine in a particular field are proposed based on the background of creative computing and semantic web. In this research,

the core contributions are the meta search engine containing creative algorithms. The following are explanations of all of original contributions:

C1: A meta search engine is proposed based on creative computing and related technologies.

C2: A creative algorithm is designed to support personalisation and humanisation of meta search engine.

C3: The design method of creative algorithm is applied based on the concept of semantic web.

Overall, this research provides a creative computing method for meta search engine, which is able to provide creative meta search engine in specific areas.

1.6 Organisation of Thesis

The structure and contents of this thesis are summarised as follows:

Chapter 1 involves the introduction and starting point of the study, lists the goals, problems, research methods, success criteria and main contributions of this research, and puts forward the overall framework of the study.

Chapter 2 summarises an overview of research background and basic concepts. The areas include creative computing, semantic web, meta search engine and data mining.

Chapter 3 discusses the related work including the specific applications and contributions existing in creative computing, semantic web and meta search engines.

Chapter 4 introduces the designing principles and algorithms of the proposed meta search engine, Writer's Portal, which is designed according to the creative computing theory and semantic web techniques.

Chapter 5 introduces the design plan and workflow diagram of the Writer's Portal.

Chapter 6 displays the different experiment results of Writer's Portal in the form of web pages.

Chapter 7 concludes the completed work, reviews the core contribution, answers the research questions, judges whether it meets the standards of success and describes what can be improved in the future.

Chapter 2. Background and Basic Concepts

Objectives

- To introduce creative computing as the kernel concept.
- To introduce semantic web as the theoretical basis.
- To explain working principles and advantages of meta search engine.
- To explain how data mining works in search engine.

2.1 Creative Computing

In the Internet era, people pay more attention to the Internet. Their focus has shifted from traditional industries to Internet businesses. Nowadays they can complete tasks and acquire information they need by media like computers and mobile phones, web pages or application programs. They might be functional or aiming at information acquisition. They want to attract their exclusive customers from market. However, many traditional industries (like real estate and automobile industries) begin to change to adapt to market. With increasing attention from people, the software engineering field has also been gradually expanded. However, people cannot solve the problems if they only rely on single domain knowledge. Thus, people try to explore a kind of creative and alternative method to solve the problems. At this point, the key idea is innovation. People help newly rising enterprises to flourish in Internet industries. In conclusion, only by innovation and

integrating independent innovation with users' needs can people find a method to solve the problems.

And one of the methods of solving the problems is creative computing, which refers to a means of computing through a certain creative approach. Generally speaking, creative computing in the field of software engineering realises the timely computing with the theories or laws in other disciplines (such as physics, chemistry or art), with the specific computing being the development of the software [4]. Therefore, people can also understand creative computing as interdisciplinary approach, which provides new ways for the development of the software with and the integrated use of different professional background knowledge, and solves the problems in the process of traditional software design [5].

Creative computing's fundamental purpose is to serve the software development [6]. According to the general way of thinking, people usually are considering how to use computer software to assist other areas, such as education, math, art, and economy. In other words, other disciplines can also help the design and development of computer software [7]. Besides, creative computing also includes some other research orientation [8-11], such as creative design, creative requirement engineering, creativity analysis and evaluation, creative collaboration, and E-Learning among others. As previously mentioned, there are still a lot of difficult problems in the field of software engineering despite of long-term efforts. Therefore, people can consider extending it to other areas, and try to solve these problems via the theory, methods and laws of other disciplines. In fact, to solve the problems in

the field of software engineering through the ideas of other disciplines is the interdisciplinary significance of creative computing.

2.2 Semantic Web and WWW

WWW, the abbreviation of World Wide Web (or named “Web” or “W3”), is normally divided into the Web client and a Web server program. WWW enables Web client (commonly-used browser) to access to Web pages on the server. It is a system composed of many interlinking hypertexts, and realises the access through the Internet. In this system, each useful thing is called as “resource” and is identified with a global “uniform resource identifier” (URI) [12]. These resources are sent to users through Hypertext Transfer Protocol (HTP), while HTP obtains resources by clicking on the links [13].

The concept of semantic web was proposed by Tim Berners-Lee from WWW Consortium in 1998 and its core is as follows: the semantic web integrates Internet into an integral database by adding meta data which can be acknowledged by computers. Since it is a database, all contents will be regraded into data by document unit. As a result, data will cover meanings in different semantics and can generate certain logical relationship. And the whole Internet becomes a general medium of exchange of information. The semantic web expands the ability of the WWW through the use of standard, mark-up language and related processing tool.

Principle of the Semantic Web

The idea of the semantic web is risen up with OWL-S, Ontology Web Language for services. In this case, a new concept as entity set, the function of which is express contents (or vocabularies) with similar meaning in the form of entity (or concept). Or I can say those contents are put in a group. Those words with similar meanings gather by entity set, so that barriers caused by different expressing method are eliminated [13].

Connection between Semantic Web and WWW

The differences between the semantic web and WWW are as follows:

I. Different Objects

Currently, WWW mainly uses HTML to express web page contents. The web pages marked with HTML indeed can express some information, like the control web page display format and make people think computer really can “understand” our intentions [14].

In fact, HTML only pays attention to the forms of texts, such as color, font size and type, and fails to consider the specific contents and meanings of the texts. Although some automatic scripts on WWW can help people to achieve some of the functions, they cannot effectively realise the interaction between the computers in the open network environment. Thus, the WWW is mainly for the use and reading of human. By contrast, the semantic web, which also includes some semantic information that computer can “understand” based on WWW, not only can enable people to read and use conveniently, but also can facilitate the mutual exchanges and cooperation

between computers. Therefore, the main object of WWW is “human”, while the main object of the semantic web is “machine” [15].

II. Different Information Organising Modes

Because of different objects, they naturally have considerable differences in the information organising mode. When organising information resources, the WWW mainly focuses on “human” and is based on people’s thinking habits and convenience, while the semantic web must consider the computers’ “understanding” of the text contents and their mutual exchange and communication at the same time [16].

III. Different Focuses

WWW focuses on the information display format and style, and does not care the specific contents. As for the relatively important information, for example, the WWW may adopt large font, or bright colour font on the display. By contrast, the semantic web is more focused on the semantic contents of information, and must provide a certain annotation or explanation to the texts with specific meaning.

IV. Different Primary Tasks

The primary task of WWW on which people mainly read, communicate and use, is information release and acquisition. Through the release or acquisition of information on the Internet, it achieves the goal of sharing and communication. The main task of the semantic web is communication and sharing between computers, so that computers can replace people to complete part of the work, and make web applications more intelligent, and full of automation and humanisation [17].

V. Different Working Modes

Due to the different objects, it is natural that the WWW and the semantic web have different working modes. To be specific, most of the work of the human-centered WWW is including information collection, retrieval and collection, sorting and analysis. By adding some semantic information that computers can “understand”, the semantic web can make people free from the above all kinds of tedious work, and help to complete the above most of the work via “intelligent agents”. A typical example is the information retrieval: through intelligent search agent, the semantic web provides the information contents that people really want, rather than output of tens of thousands of useless results like current search engine [18].

2.3 Definition and Advantage of Meta Search Engines

Based on the examples listed in the fifth part of the above sections, there are usually a lot of useless results among the search result retrieval of search engine, which is mainly highlighted by the congenital deficiency of WWW because the traditional search engine is built on the concept of the WWW [19].

Two obvious deficiencies of the WWW are as follows: Firstly, computer cannot understand the semantics of web contents: because the WWW is a document carrier, and it is only for the purpose of reading [20]. Secondly, information on the Internet is disordered: although there is much information on the Internet, it is often difficult to find the desired information, and has low precision ration even with the help of powerful search engines because of too much redundant information, and the

naturally disordered relations between all kinds of information [21]. Therefore, it costs more to find the needed information on the Internet - according to the survey, many people spend more than six hours in searching every week [22].

These problems emerge because the WWW currently adopts the Hyper Text Markup Language (HTML) in that the contents of the web page are designed for human to browse, rather than for computer to process and understand [23]. Thus, it cannot provide the Internet users with the functions of automatic processing or analysis of online data [24]. In addition, the WWW locates the information resources according to “web address”, instead of “semantic content” (because of the lack of semantic connection). All the information on the Internet is issued by different websites, and the same topic information is scattered in many different servers around the world, and lacks of effective tools to integrate the different sources of information. Therefore, there is an information island (including a lot of repetition), and it is quite difficult to find out the required information. Another reason is the search engines fails to consider the different behavioural habits of each individual [25].

Based on the above-mentioned situation, Meta-search engines based on the semantic web concept tries to resolve this problem.

Definition of Meta Search Engine

Meta-search engine is a search tool aiming to help users to choose and use appropriate search engine (even several search engines simultaneously) from multiple search engines through a uniform user interface to realise retrieval

operation, and it is the global control mechanism of various retrieval tools distributed in the network. It mainly focuses on improving searching speed, intelligently handling of search result, personalising searching function, and increasing friendliness on the users' retrieval surface.

Advantages of Meta Search Engine

1. On account of the concept shown above, after identifying searching needs, a meta search engine hunts on multiple search engines and returns result to the users. Under the promise, the recall ration of a meta search engine is obviously higher than that of the traditional search engines.

2. According to the concept of semantic web, by adding the semantic information that can be understood (namely, mata-data), the contents in the database will have semantic relevance and logicality. In the process of retrieving, the accuracy of content extraction will be higher than that of the traditional search engine.

3. The author can personally provide targeted search of a restricted field by compiling algorithms of metadata search engine [26].

Algorithms for Constructing A Meta Search Engine

In the section, the algorithm logic of a meta search engine based on the concept of semantic web is briefly introduced.

1. Algorithm about Extended Inquiry

In the extension of search, the user model plays a role of bridge between the query retrieve and the index file. Through the user model, the retrieval intention can be speculated and as well as the user's needs for information, then improving the user's search and realising the extension of search. Based on the extending of the semantics of the entity user model, the main process of the search extension is divided into two stages:

1. Mapping. Though the match of key words, each of the key words of the Q1 (K , K , , K) can be mapped to the user model, forming an conceptual set Q2{C , C , , C }.
2. Semantic Extension. In the knowledge of entity, if there are one or more property sequences between two concepts, these two concepts could be called semantic relatedness. The intention of the semantic relatedness indicates the close of the semantic relations. The semantic extension of the entities can be realised by the semantic relevance of the entity concepts. This means to find the close concepts of Q2, thus accomplishing the semantic extension of the user's interest.

2. Algorithms for Gathering Customer Interests

The research on user interest preferences means a process that adopts learning methods to gradually specify user interest preferences according to the selections of users on browse information as well as their feedback information. In this study, the personalised information of users is mainly extracted by the methods of direct

learning and feedback learning. First of all, it is to extract information of interest characteristics in accordance with users' interactive manipulation and form users' interest. Then, it is to make adjustments toward the model of user interest preferences by extracting the users' search information and browsing history. The user interest model is not a general description for individual users, but an algorithm-oriented user description with particular data structure and formalisation. It is composed by objects signifying user interests, and each object has the weight information. The higher the weight is, the stronger the users' interest toward such information. In terms of the user interest topic, it can be denoted with keywords and weight's vector space model. The methods of calculating keywords in the document weight include Boolean value algorithm and term frequency algorithm.

Development of Meta Search Engine

There is still developing space for MSE since each user has different behavioural habits. Then what they search may be related with interests and work or meaningless. Thus the search engine needs to know further about users and backup for users' information, and then match concerned data to resolve these problems.

Even though the development of MSE is not completed, it becomes a development trend. The aim of search engine is to provide information to users, and the accuracy of information will decide customer's experience. More innovative meta data search engines will appear with people's gradual innovation in the future.

2.4 Roles of Data Mining in Improving Functions of Search Engine

It was found that a meta search engine had higher recall ratio and precision ratio than that of the traditional search engines. In addition to the advantages of the meta search engine itself, the technique of Data mining used in search is also one of the factors that effectively enhance the recall ratio and precision ratio of meta search engines [27].

1. Data Mining and Its Roles in the Search Engine

Data mining in network is developed from the mining technique of traditional data. The traditional data mining generally refers to the process in which people extract the implicit and beforehand unknown but potentially useful information from large, incomplete, noisy, and fuzzy data via algorithm [28]. The main tasks of data mining include classification, clustering, association rules and deviation detection. Because the information on the Internet is no longer common database data, but irregular data, such as documents, pictures and multimedia, it is more difficult on the technology processing [29].

Network information mining can be divided into web content mining, web structure mining and web usage mining. The emerging technologies of network data mining and search engine make they complementary, rather than mutually replaceable because of intersection of the use range. Through the better use of data mining technology in search engine, people can fundamentally improve the performance and the efficiency of search engine [30].

2. Applications of Data Mining Technology in Search Engine Information Retrieval

The document mining technology used in web content mining has the same idea with the automatic information retrieval technology used in the early search engines, which expressed the main ideas of the document by extracting a number of phrases from the document [31]. Because the web document mining extracts larger number of and more comprehensive words that can more effectively express the real meaning of the document on technical treatment, web documents mining technology in the search engine can improve the quality of the automatic summarisation.

3. Applications of Data Mining Technology in Search Engine Text Classification

In terms of classification and clustering, data mining technology also provides search engines with good reference; in data mining, text classification is realised via space vector, fuzzy model and other technology, which can achieve the fast automatic classification of a large number of texts [32]. However, there have not been better technologies in search engine classification, which depends on artificial means in many cases. The use of the text classification of data mining technology can realise the automatic classification of web pages, narrow search scopes and improve the search accuracy [33].

The text clustering of data mining technology is the opposite of the classification, because it divides a large number of documents into multiple clusters through a variety of text clustering algorithm, and makes the relationship between the different clusters small as far as possible, and the relationship between the same

cluster big as far as possible. For growing large document problems on the network, the text clustering can be processed objectively and quickly, gather information of the same class together, provide the most relevant contents for users' search, reduce the amount of information provided to users, avoid the reduplicative response when users input one question, and making users being surrounded by a sea of information [34].

The application of structure mining of data mining technology in the analysis of search engine retrieval web mainly refers to the analysis of web hyperlinks, the excavation of the potential semantics of web, and providing more accurate basis for the analysis of the retrieval of search engine. Web information organisation adopts the flat structure method, so it is difficult for general search to deal with it; through the web structure mining technology, it can retrieve more accurate page, improve retrieval performance, and enhance the ability of retrieval. The specific solutions include the page rank and authority page.

The applications of network use data mining technology in the analysis of search engine user interface. It mainly refers to the mining of the user retrieval, so as to find out the patterns users like and potential rules, and constantly improves the search engine. Data mining technology also has many other aspects of specific application in the search engine system. Hyperlinked information in the web page provides rich information on the relevance, quality and structure of document contents. By using the information, people can prioritise page, find authority website [35]. Thus, such method can improve the search quality of robot. For example, each server on the

WWW keeps the access log (Webac, Eesslog), and records the information about the user access and interaction. Learning users' access pattern and tendency through the analysis, use and the record can improve the organisational structure of the website, establish efficient access methods for the highly relevant objects, rather than access according to the original page method. PageGather is an application. Besides, through tracking access patterns of the single user, it can analyse a single user's preferences, and provide personalised service, with the research projects in the aspect including SiteHelper, Letiza, etc. Additionally, Data mining techniques can also find the relationship between the different words from the collection of documents through the correlation analysis of Web document, explore more profound connotations via the relationship, and obtain the value trend of specific data in a historical moment or in the future through the analysis and trend prediction of Web document distribution [36].

2.5 Summary

In this chapter, the background and basic concepts of creative computing, semantic web, meta search engine and data mining are introduced and explained. Firstly, it contains the design principle of creative meta search engine. Secondly, it includes basics framework of meta search engine. Thirdly, it introduces data mining domain for meta search engine development.

Chapter 3. Related Studies

Objectives

- To review specific applications of Creative Computing.
- To review and discuss specific applications of Semantic Web.
- To review and discuss related meta search engines.

3.1 Applications of Creative Computing

Two applications of creative computing are illustrated to show it can help with building an MSE.

Sygyzy Surfer -- Creative computing resolves problems though the assistance of knowledge in different fields. Two examples are as follows. The Sygyzy Surfer is a newly proposed web search engine whose purpose is to enrich users' search experience by enhancing the individuation and creativity of search engines. Compared to the traditional searching engines such as Google, Bing，Yahoo!, it is more humanised. The design concept of this search engine was proposed by Hendler, Hugill, Yang and Racizinski [37]. The Sygyzy Surfer uses the technology of Pataphysics in design, which is a science of imaginary solutions for governance of exceptions, and this provides users with a new data processing software [37，38]. Meanwhile it is also a good example that combines creative computing and the contents of an application field.

Scratch -- Scratch is a simple programming tool designed for teenagers by MIT. Almost all the children will love it at the first sight. The purposes of this program are to improve teenagers' abilities in various aspects through programing, including the cooperation ability after acquiring various kinds of knowledge, which is consistent with the principle of creative computing [39].

It is not necessary for software users to understand English words or use keyboards. Orders and parameters of the program can be achieved by modules with shapes of building blocks. It is difficult for children to learn how to use this software all by themselves. Therefore the guidance from teachers and parents is needed. With the help, a six-year-old child can build a complete program, which is able to run. Children do not need to know the language during the process, which involves in graffiti, recording, looking for pictures and other interesting processes. The finished products can be released on the official website through the software because the official website provides every registered user with space for the program of releasing products [40].

The programs released by users can be found on the official website. Unfinished programs can only run in the software while the released programs run in the website, which means children's works can be seen by people through the internet. The official website has the function of making friends and commenting. In this way of "frank exchange and thought collision", it is more helpful to promote the cultivation of teenagers' creativity.

3.2 Specific Applications of Semantic Web

Even though Semantic Web is in research stage, it is applied on some degree in some different fields. For example, in EU plans, Ontoweb uses the semantic web make basic information exchange for ontology of knowledge management and e-commerce; Equally, Adobe tools add RDF-base meta data in all their file format on account of the semantic web concept; Boeing Co. uses meta data integrate data of company resources into semantic groups [41].

The next two sections will introduce semantic web and function of semantic web in search engines.

3.2.1 Service of Semantic Web

The application of semantic web in realising the personalised service is recommendation system, especially content recommendation. The relatively successful application is TWINE [42]. The principal services by twine homepage are as follows:

- Find user's interests.
- Collect shared bookmarks, and provide network bookmarking service.
- Provide personalised recommendation service in accordance with the interests.

The core service is the third point, namely personalisation recommendation. And the premise of personalisation recommendation is to collect user information [43]. After grapping the semantics, Twine provides the following a few good services. Firstly, it

can be an automatic archiving system of a user's personal data and automatically classify the information file you submit (like your own personalised library), without requiring users to build directory like traditional way, or waste time and energy in tags, but inaccurately. Secondly, it can recommend users contents according to users' preferences, such as the web sites and web page users may be interested in (it seems that there is no goods on sale at present). Thirdly, it also further aims at the socialisation direction of network and concretely will make the same kinds of users (the ones who submit the same category of contents) form circle of small community, and realise the functions of SNS. Finally, it also provides the functions of community information sharing, and allows users to share contents (search) through label.

3.2.2 Meta Search based on Semantic Web Techniques

The main demand source of semantic web is possibly because of "the contradiction between increasing requirements of user for information retrieval and the current backward search tools". The current web search engine based on string matching might enough for most people but it is quite difficult for some high-end users or very low-end users to use the current search engine. For example, low-end users are not able to screen keywords, so they often feel confused in the face of hundreds of pages of search results. In fact, they may prefer to obtain accurate results via the way of question and answer. In addition, some high-end users, such as some analysts or the ones who mainly write reports, hope to access to the various dimensions of related online data accordingly, but the multidimensional data cannot be included in

one web page (also not fully within one web site), so people need search engine and retrieve all the relevant data on the net according to semantic links [44].

The semantic search required by low-end customers is called “semantically enabled search engine” in some places, where main characteristic is to improve users’ search experience by means of natural language processing, and pattern recognition technology. It supports users to retrieve information needed via questions, but the returned results are still limited to the contents of one web page. The high-end users prefer to the entire network information index across the information island, which is more like semantic database search. Therefore, some reference mentioned below is summarised according to the level.

Here are three examples of the search engines of semantic analysis in the field:

Powerset -- The technology core is to try to understand the meaning of the sentences via natural language and match them. Its data mainly come from Wiki, meaning people can search through the problems in life. Powerset will analyse your problems, and find out the corresponding sentences including the answers in wiki. Additionally, in order to provide users with better overview summary, it also shows by combining with the structured data of some freebase [45].

Cognition -- The engine is quite similar to Powerset, because their specialty is natural language processing. Early cognition seems to provide the search of any content, but it seems to be more focused or only provide limited fields of semantic search including law, health and Wiki at the moment [46].

Evri -- The above two web sites mainly provide natural language search, but Evri is a very good content organisation engine. Its core technology is not a simple search engine, because it adds a complex semantic layer to the search results so as to emphasize the relationship between the different search contents rather than show the search results to users without any processing. It has technical advantage because it can mine and classify the disorderly contents of the search results. To be specific, to provide summarisation of the search contents within one page seems to be a trend of displaying the current search results. The most prominent aspect of its semantic classification is its emphasis on the several important dimensions relevant to the result subject, such as people, matter and place. For example, when people search “Obama”, the results will include the classified information, such as his basic information, and the time and place of his participation in events.

3.3 Different Design Principle of Meta Search Engines

In current MSE, developers have proposed different ideas to improve accuracy and efficiency of traditional search engines. Zhu’s [47] team has recommended building a middle-ware between client-side and server to research users’ habits, coming up with new extension words. These words will operate at background in well-known search engines to form more related searching results. However, Scout [48] confirms the relativity of result by collecting users’ feedback and resubmitting searching contents. However, Outside [49] integrates users’ searching results with users’ information to check content relativity, finally based on which to rearrange and

filtrate user's searching results. I can find easily that the application of the meta search engine has application that has many directions by these examples.

Next, two different meta search engines are respectively introduced from the aspects of the design concepts and solutions.

3.3.1 Meta Search Engine based on Agent Technology

3.3.1.1 Agent Technology [50， 51]

Currently, agent technology is one of the fastest growing technologies in the field of artificial intelligence. Its autonomy, initiative, reactivity, mobility, sociality and intelligence make it superior to the traditional distributed technologies. Combining with the agent technology and search technology, the author mainly hopes to make Web information retrieval more intelligent. Under the network and distributed environment, each agent is independent, can work on themselves and the environment, control part expression of the environment, and react to the changes in the environment. More importantly, it can realise communication, interaction, cooperative work and complete tasks together with other agents. In information retrieval systems, for example, learning agent makes the system have more ability to learn, capture users' changing interest, and filter out irrelevant information of users. In view of the retrieval efficiency of meta search engine, mobile agent can save network bandwidth, reduce the network time-delay, and enhance retrieval efficiency by using its asynchronous and mobile computing, and parallel retrieval ability.

3.3.1.2 Search Engine Model

In order to solve the problems of traditional meta search engine, the author of this type of search engine puts forward a kind of agent-based retrieval model.

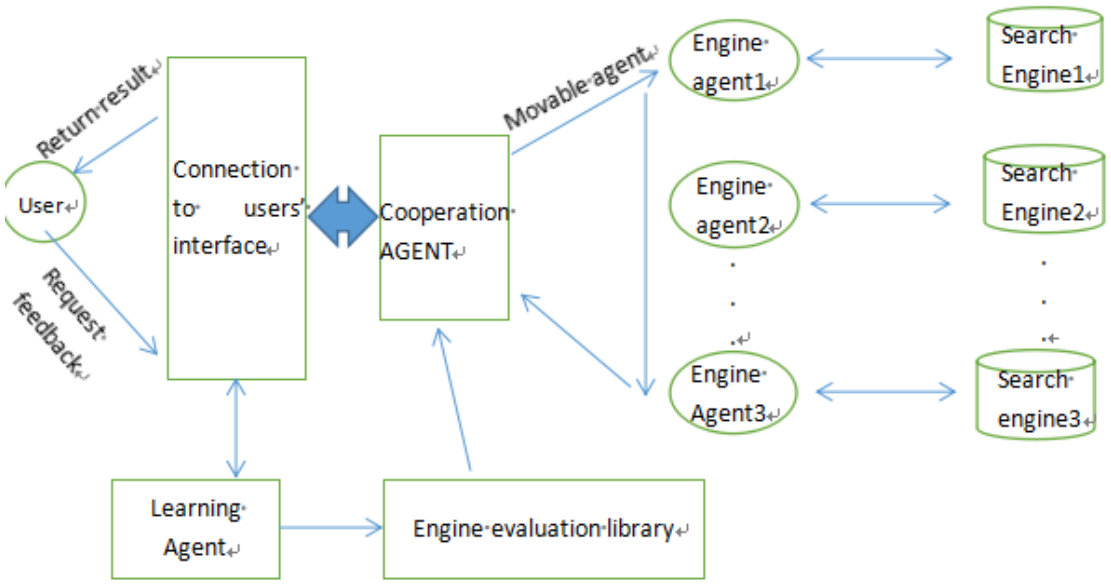


Figure 1. Search Engine based on Agents

Unlike the traditional structure, retrieve server can choose the members engine according to the engine evaluation library when receiving user retrieval request, and then create one or more of the mobile agents at the same time and assign them to each proxy server. By carrying the query words, mobile agent sends them to each engine agent respectively. The engine agent converts the query words to the formats applicable to each big search engine, connects the search engines and gets the results. Then, the mobile agent carries the results and returns to collaboration agent, which will blend the results, and return them to the users. Besides, learning agent reorders the results through user feedback, first show the information that the users have interest, and updates engine evaluation library. In this model, the thought of

distributed is applied. Moreover, the engine agent, collaboration Agent, mobile agent as well as learning Agent are added to jointly complete the retrieval.

3.3.1.3 Algorithm Design Ideas Based on User Feedback

Users' click through behaviour and designs were studied for more practical algorithms. Many studies [52] have made detailed investigations for users' click through, and their finale experimental results show the users' click data are mixed; click data indeed can show users' interest, but they contain a lot of noise, and need targeted processing. Noise mainly includes two aspects of factors: trust bias and quality bias. Trust bias means that users tend to click the front documents, which are not necessarily the most relevant documents due to the trust of the search engine. Quality bias refers to the influence of the quality of current ranking results on users' click on documents. Through correlation analysis, it is proved that users' click on the document correlation is low when sorting the results are of poor quality, so the quality of the retrieved results will affect the quality of users' clicks. The author ranks the search results through the index QR of measuring the quality of search engine ranking results [53], and the index can be used to measure the quality of ordering results of a document.

3.3.2 Meta Search Engine Design Based on ASPHTTP Components

3.3.2.1 Definition of ASP

ASP is an application developed by Microsoft, which means "active server pages". ASP is relatively powerful technology of currently developed interactive Web page and Web database application, in which ActiveX, HTML, DHTML VB2Script or JavaScript can mix. In addition, using the server components can further extend the functions of ASP. Server component, actually a DLL running on the server, can complete any task that the regular DLL can accomplish, but it is different from the regular DLL because it is composed of ASP page calls, takes Web page as interactive objects, reads users' input (values of each input domain on Web page), and returns results to the Web page after processing. These interactions, of course, use the Web server as an intermediary. The main properties needed in the design of meta search engine include BinaryData, RequestMethod, TimeOut and URL in ASP.

3.3.2.2 Designing A Meta Search Engine Through ASPHTTP Components

Specific steps of this design are as follows:

1. Designing search engine interface. A meta search engine needs to send search requests to multiple search engines through an interface. Therefore, it is necessary to know each search engine program interface. In fact, it is easy to access to these interfaces because people only need to enter one key word in the frame of each

search engine a key word, and then can find out the corresponding interface in the address bar of result window.

2. Designing characteristic analysis result return page of search engine. Different return result pages of search engine have different structure. To extract retrieval results from various result pages, it is necessary to analyse the characteristics of Web page returning to each search engine so as to reach the purpose of identification. It mainly includes two characteristics: one is the initial identification of each search result and another one is the termination identification of the last search result. For the Web page that the same search engine returns, different designers can provide different analysis results of the page features, as long as people can extract the search results in the Web page accurately and quickly.

3. Deciding limit of search time. The search time of meta search engine is conditioned by each search engine. If the search time of every search engine is not limited, the total search time will be quite long. The search time of each search engine can be extended or shortened by modifying the above TimeOut attribute in the code. Relatively reasonable TimeOut attribute value can be achieved via trial and error method.

4. Programming realises the dispatch of search request as well as the extraction, duplicate removal, sorting, formatting, and paging display of the search results. On the basis of the first three steps of work, meta search engine is compiled through hands-on writing code. Besides, dispatch of search request through program interface of each search engine and the extraction of search results via web feature

are conducted on the basis of ASPHTTP component. The rest of the work includes duplicate removal in the summarised result set, result sorting according to the rules specified by designers, formatting according to the display style selected by the designers and paging display to users.

Chapter 4. Design Principle of Proposed Meta Search Engine

Objectives

- To introduce the working principles of the meta search engine in detail.
- To introduce the detailed process of meta search engine in single domain designed according to the creative computing and the idea of semantic web in detail.
- To introduce the specific process of the algorithm.

4.1 Proposed Meta Search Engine

The proposed meta search engine aims to integrate some universal search engines together so as to provide a unified query interface for the user. The user queries the format that can be recognised which is transformed to the universal search engine by the meta search engine. Then those search engines will complete the actual information retrieval after it is sent to the universal search engines. In the end, the meta search engine will collect the results returned by the search engine and will return to the user after analysis and processing [54，55]. Figure 2 is the structure chart of typical meta search engine.

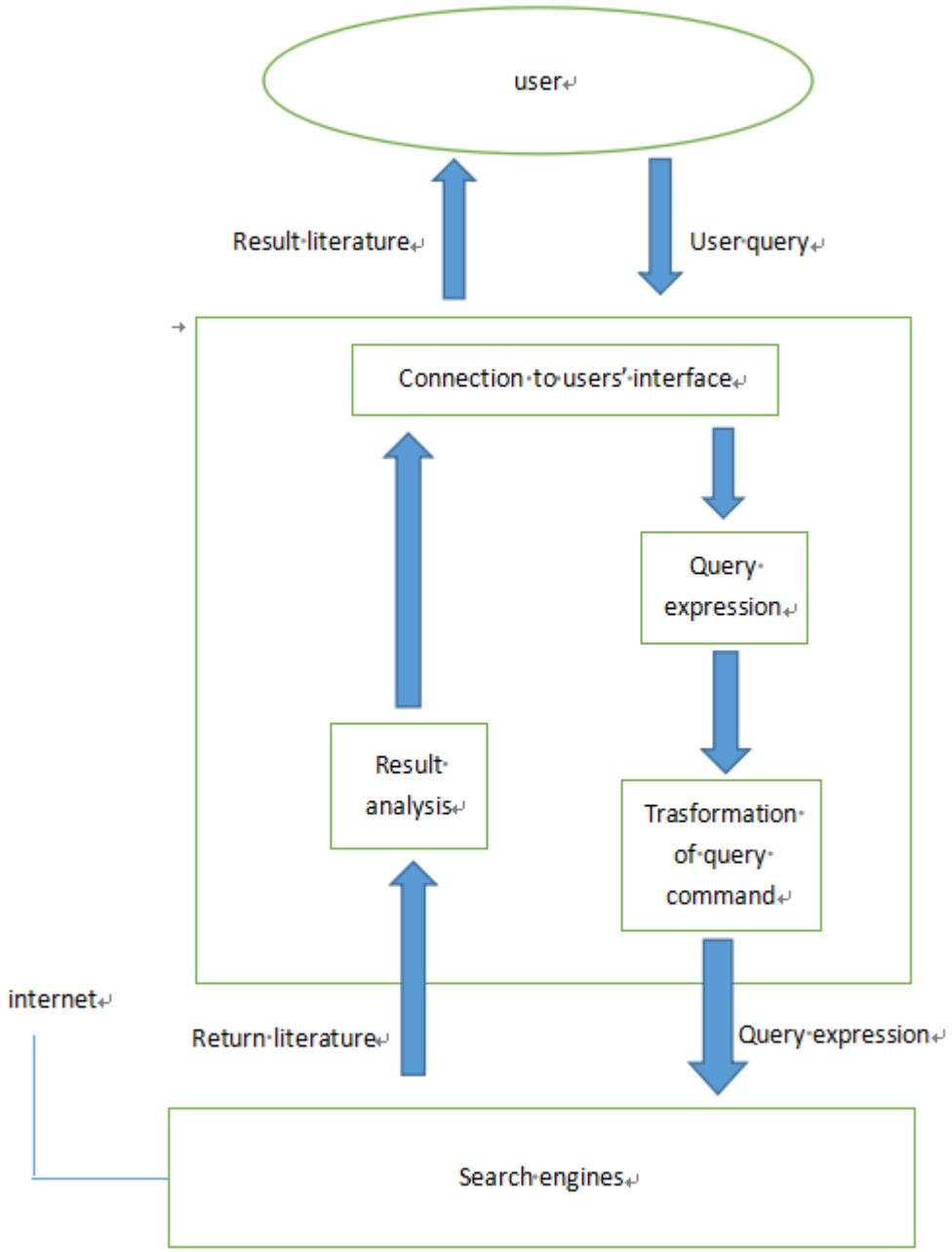


Figure 2. Structure of Meta Search Engine

The meta search engine has the following features: (a) do not have huge page index database or need the roaming net or page indexing; (b) can link many universal search engine and retrieve many index databases at the same time so as to expand the query range and increase the recall of information searching greatly; (c) because

the meta search engine visits many universal search engines at the same time, there may have the complicated indexing whose searching time may be a little long, which is hard to deal with, and whose ranking results may be unsatisfactory. The proposed search engine is to provide some financial aid and publishing opportunities for the professionals so it has limitations and professional qualities in the filtering of search results. With the advantage of the meta search engine, considering the relevant indexing requirements of the queried domain, further development can add query expansion techniques in the module of query expression and add the functions of words weight information, ranking of the literature in the domain and the re-ranking through user's feedback in the module of result analysis and retrieval. By doing so, domain search effects of meta search engine can be improved and create a personal meta search engine [56-67]. The proposed meta search engine is designed on the above basis.

4.2 Design of Search Model in Writing Domain

The study uses the bottom search engines, i.e., a popular search engine (Google), to personally display results by designing creative algorithms on account of the semantic web concept. Then it meets clients' needs specifically. The new client search engine is named Writer's Portal. Based on the idea of semantic web, the chosen search method is with a domain-keyword (KeywordSpice, KS) to search the Web. The method uses the "decision tree" construction domain query expression to reflect the requirement for querying domain of the user and submits it together with

the querying words of the user to the universal search engine. The proposed method pays attention to the following aspects. First, this study defines the domain-specific keyword as the indexing words which can reflect the features of literature in the domain and distinguish the irrelevant literature and results in that the generated domain query-based expression cannot reflect the querying requirements of the user fully; second, the domain specific-keyword has no weighting, ignoring the difference of domain specific-keyword between the literature and the query; thirdly, KS method is designed aiming to enhance the search result quality instead of a whole solution. Therefore, it is more than just a Boolean query without ranking for the result literature. The proposed meta search engine applies the key techniques in KS method. Based on the features of creative computing of knowledge utilisation in various domains, the model applies the odd ratio (OR) [65] to extract the domain specific-keyword; uses TF-IDF (Term Frequency-Inverse Document Frequency) method to set the weight of the domain-specific keyword; uses decision tree (DT) to generate the domain querying expression based on the domain-specific keyword, and then uses extended Boolean model (EBM) to rank the search results.

4.2.1 Structure of Writer's Portal

My experience gained during working on the Bertie project was referenced when designing the proposed MSE. The Bertie project, conducted at the Centre for Creative Computing of Bath Spa University, was to build a search engine to search a set of education related websites specified by researchers in the School of Education

at the university. The technical angle for the Bertie project was “personalisation”, i.e., accommodating users’ interests, emphasising classifying search results.

The technical angle of this research was “data mining” and ”applying semantic web techniques to ‘meta’ concepts being searched”. I constructed an ontology for the domain of writing, designed a different workflow, new user interface and eventually coded a prototype.

The designed Writer’s Portal is on the structure shown as Figure 3. In order to strengthen the relevant indexing ability in single domain of the meta search engine, the author implemented the following functionalities for the proposed meta search engine.

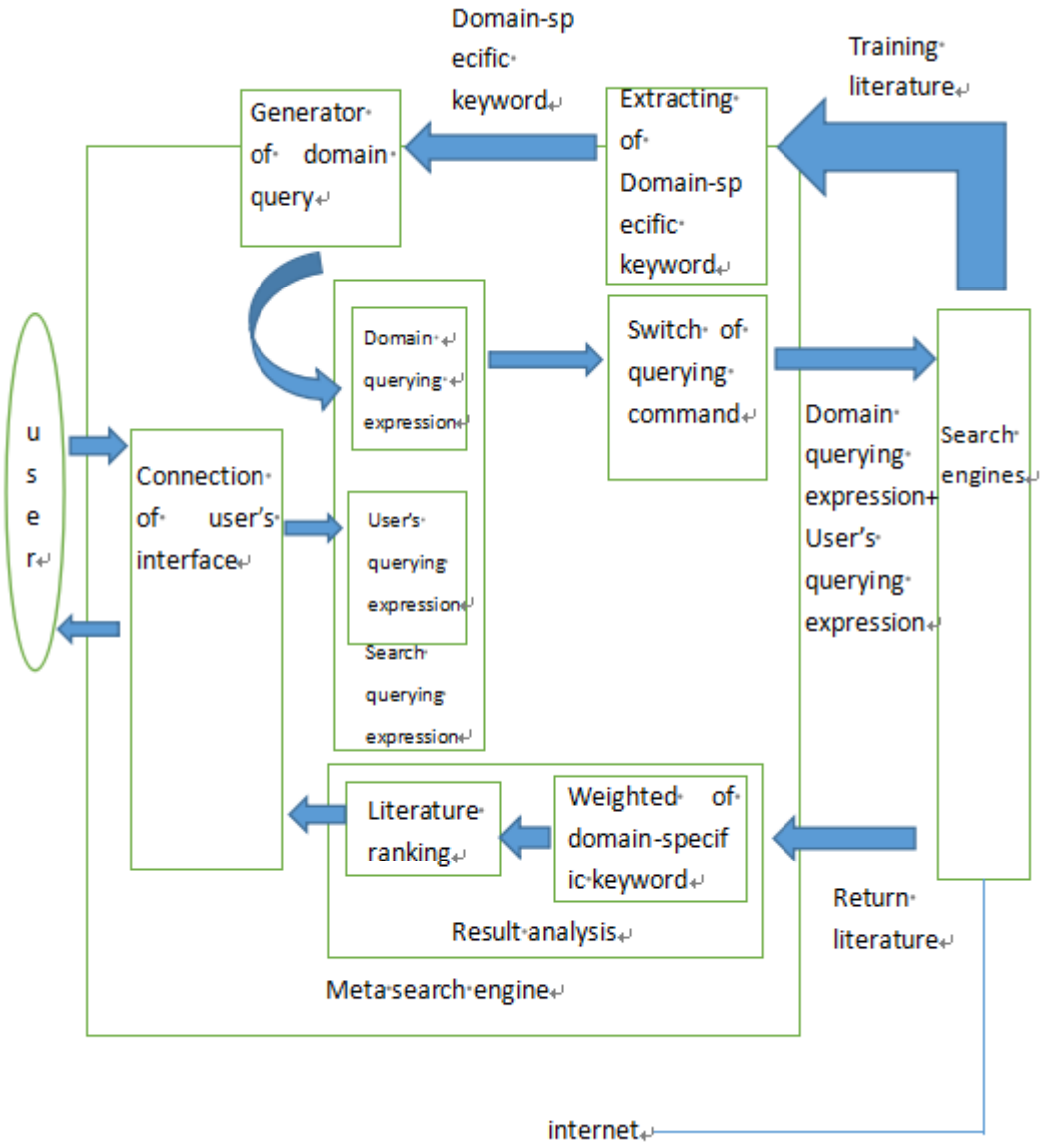


Figure 3. Structure of Writer's Portal

(a) Module of domain-specific keyword: choose the domain-specific keyword according to the training literature from the universal search engine; **(b)** module of domain querying generator: generating the domain querying expression based on the relevant domains with writer's economic support to express the information features of the relevant literature in the required domain; **(c)** the querying expressions are divided into two parts: user's querying expression and domain querying expression; combining the two querying expressions as the input of

universal search engine; **(d)** the result analysis is divided into two parts: weight of domain-specific keyword and module of literature ranking. The module of domain-specific keyword weight measures the importance of the domain-specific keyword in the search domain; the module of literature ranking will return the literature and feedback to the user after ranking them according to the relevance with domain.

Technical details of the following prototype system modules, i.e., the domain-specific keyword extracting, domain-querying generator, querying expressions, domain-specific keyword weight, and literature ranking, will be discussed further.

4.2.2 Domain-Specific Keyword Extracting

There are several steps in this module.

a. Choosing training ontology -- Ontology is selected as the way of training literature. Based on the idea of entity set in semantic web, a set of network ontology can be obtained after the input of several keywords belonging to the same domain in the universal search engine. This set of network ontology includes enough relevant ontology and division is done on the ontology into relevant and irrelevant ontology to the domain so as to get the training ontology.

b. Extracting domain-specific keyword -- After a series of ontology pre-treatment such as removing punctuations, prepositions, and pronouns, choose the nouns and some important verbs as the alternative domain-specific keyword. The method of

extracting the domain-specific keyword is similar with the choosing method of textual features [65] such as information gain (IG) and mutual information (MI). This study chooses the OR method because OR method is applicable to the binary classifier. In binary classification, itcan recognise as many positive kinds as possible while ignoring the recognition of negative kinds. The OR method is shown as follows:

$$OR = \log \frac{P(W \mid C)(1 - P(W \mid \overline{C}))}{P(W \mid \overline{C})(1 - P(W \mid C))} \quad (1)$$

In which:

$$P(W \mid C) = \frac{\text{The total number of literature Class C which contains the key word "W"}}{\text{The total number of literature Class C}}$$

The larger |OR|, the stronger power of word W for classifying the two kinds of ontology. When OR>0, most of the ontologies containing W belong to this domain, so W should be chosen as the domain-specific keyword; When OR<0, most of the ontologies containing W do not belong to this domain. However, because such kind of word in the domain querying expressions can remove the irrelevant ontologies effectively and therefore improve the indexing effects, W is chosen as the domain-specific keyword. The smaller |OR|, the weaker power of word W for classifying the two kinds of ontology and it cannot be the domain-specific keyword. Therefore, the domain-specific keyword not only includes the feature words of the relevant ontology of the search domain but also the feature words of the irrelevant ontology of the domain.

4.2.3 Query Generator for A Single Domain

The query generator of the single domain is the independent functional module of this study and it can be achieved through many construction methods under the condition of having no influence on other modules. This study chooses KS method to generate the simple and effective domain querying expressions, integrates the relevant knowledge of the domain into the user's querying request and increases the relevance of search results and user's request domain while narrowing the search scope.

4.2.3.1 Generation of Domain Query Expression

This step is to divide the training ontology into two parts. One part is used to generate the original domain query expressions and another part is used to simplify it. The specific method is based decision tree method, expressing the relevance of domain-specific keyword to the domain as a decision tree, as is shown in Figure 3. The node represents the domain-specific keyword and the value of the branches represents the attribute value of node (when the ontology includes the domain-specific keyword, the value is 1, otherwise it is 0). Each leaf knot represents the classifying condition of literature. From the root knot, check whether the literature contains this feature (domain-specific keyword) and receives a corresponding branch. Such kind of process will continue till it gets to a leaf and the literature has been concluded as belonged to the corresponding classification. This tree divides the literature into T (relevant ontology of domain) and F (irrelevant ontology).

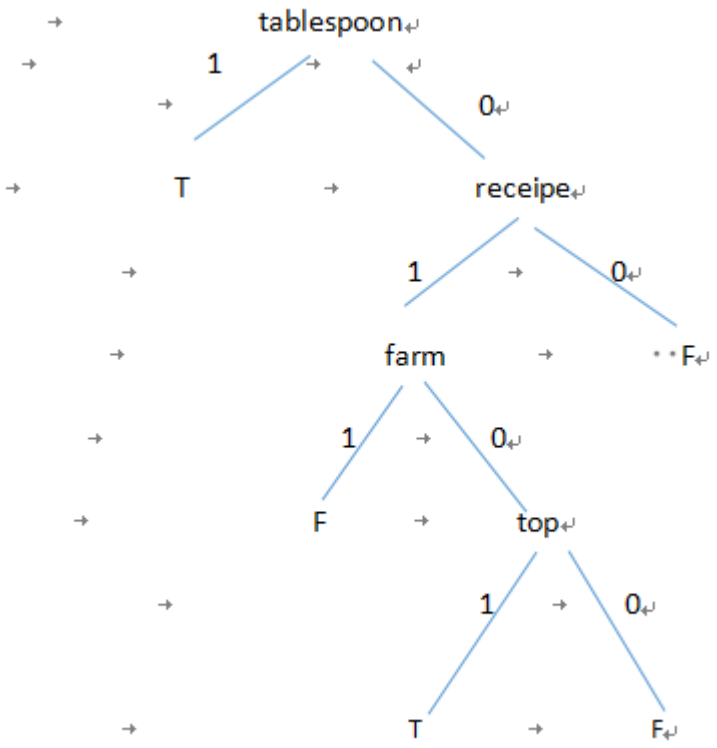


Figure 4. Examples for A Decision Tree

As for the tree in Figure 3, the positive path (the path from the root knot to the leaf knot with the value of) is expressed as the conjunction expression based on the domain-specific keyword. The disjunction expression consisting of all conjunction expressions is called the domain querying expression to express the relevance between the literature and domain. For example, the decision tree in Figure 3 expresses the domain querying expression:

$$tablespoon \vee (\overline{tablespoon} \wedge recipe \wedge \overline{farm} \wedge top)$$

4.2.3.2 Simplifying Domain Querying Expression

Generally, the original decision tree is very huge and the generated domain querying expressions are very complicated so the universal search engine is hard to be used directly. Therefore, simplifying is needed for the original domain querying expression. In the theory of information indexing, the precision and recall are frequently used as the evaluation index of searching results.

$$\Pr ecision = \frac{The\ \text{total number of retrived ralated literature}}{The\ total\ \text{number of retrived literature}}$$

$$\text{Re}call = \frac{The\ \text{total number of retrived ralated literature}}{The\ \text{total number of related literature}}$$

The relatively high precision means that the searching results contain little irrelevant ontology; the relatively high recall means that the searching results miss little the relevant ontology. This study regards the harmonic mean of precision and recall as the evaluation function for simplifying the domain querying expressions:

$$Z = \frac{2}{1/R + 1/P} \qquad (2)$$

The simplification of domain querying expression is conducted in two steps: (1) as for each conjunction expression in the domain querying expression, try to remove the domain-specific keyword till that the removing of any domain-specific keyword will lessen the value of Z; (2) try to remove the conjunction expression in the domain querying expression till that the removing of any conjunction will lessen the value of Z. Through the simplification, we can get a simple domain query expression reflecting the ontology feature. The KS method referred in [64] can get good indexing result but it has the disadvantages of the three aspects. Therefore, this

study designs the modules of domain-specific keyword extracting, domain-specific keyword weight and literature ranking to make up for those disadvantages so as to improve the indexing effects.

4.2.4 Querying Expression

The querying expression is defined as the form of conjunction

$$q = q_p + q_s \quad (3)$$

The user's querying expression q_p is the Boolean expression input by the user to reflect the querying request, normally as the single search word; the domain querying expression q_s is the domain querying expression generated by the domain querying generator automatically, reflecting the features of the domain ontology. The q_s in the designed meta search engine in single domain in this study is different from the q_s in KS method; the difference lies in that the domain-specific keyword in KS method is chosen randomly. However, in the meta search engine in single domain, the domain-specific keyword used to generate q_s is chosen by the module of domain-specific keyword extracting. Therefore, such kind of q_s can reflect the features of domain ontology more accurately.

4.2.5 Module of Domain-Specific Keyword Weight

This step deals with the following issues.

a. Ontology-relevant domain-specific keyword weight

There are many determination methods for the ontology-relevant domain-specific keyword weight. Without loss of generality, this study applies the method of "TF-IDF":

$$w_{ij} = \frac{freq_{ij}}{\max_l freq_{ij}} * \frac{idf_i}{\max idf_i} \quad (4)$$

w_{ij} represents the weight of domain-specific keyword i in literature j; $freq_{ij}$ is the frequency of domain-specific keyword i in literature j; $idf_i = \log(N / n_i)$ represents the inverse literature frequency of domain-specific keyword I; N is the amount of literature; n is the amount of literature which contains the domain-specific keyword i. $\max idf_i$ is added to the denominator to make $0 \le w_{ij} \le 1$ so that it is applicable to extend the Boolean model. Each literature returned by the universal search engine needs the calculation of the domain-specific keyword relevant to the literature.

b. Query-relevant domain-specific keyword

From the choosing methods of the textual features, it can be found that the importance of the domain-specific keyword in the domain querying expressions is different. The larger value of $|OR_i|$, the more important of domain-specific keyword i. Therefore, the definition of the query-relevant domain-specific keyword is:

$$w_{iq} = |OR_i| \quad (5)$$

The query-relevant domain-specific keyword is got from the calculation of training ontology. It can be modified according to the user's request so it does not need to be calculated repeatedly in each indexing.

4.2.6 Module of Search Result Ranking

In information indexing theory, there are many methods to rank the search results according to the importance or relevance. The extended Boolean model integrates the advantages of Boolean model and vector space model and has unique advantage for dealing with Boolean expressions. The designed domain querying expression of search engine in single domain is also based on Boolean expression.

4.3 Indexing Algorithms for Meta Search Engine

To illustrate the search mechanism of the domain search model, this part will introduce the indexing algorithm of meta search engine in single domain. Through choosing the domain-specific keyword k and determining the weight of the domain-specific keyword w_{ij} and w_{iq}, the algorithm chooses the maximum value of the search target harmonic average value z because only when both the precision and recall are higher, the value of z will be larger and the better indexing effects. The parameters notification: D represents the literature collection; k represents the domain-specific keyword collection; q_s represents the domain querying expression generated by decision tree method.

S_j is the literature similarity calculated by formula (9) and (10); S is the critical value of similarity. If $S_j \geq$ S,, then literature d_j is the relevant literature; A represents the literature collection; |A| is the amount of the result literature; |G| is the total amount of all relevant literatures in D; |GA| is the total amount of the relevant literature in result literature; OR(D) represents to choose the domain-specific

keyword from literature D with the RO method; DT(k) represents to generate the domain querying expression of domain-specific keyword k by decision tree.

Evaluation function: $\max Z = \frac{2}{1/R + 1/P}$

Constraint condition: k=OR(D) (11)

$$w_{ij} = \frac{freq_{ij}}{\max_l freq_{lj}} * \frac{idf_i}{\max idf_i}$$

$$w_{iq} = | OR_i |$$

$$q_s = DT(k) \quad (12)$$

$$q = q_p \wedge q_s$$

$$S_j = sim(q, d_j) = sim(q_s, d_j)$$

$$A = \{d_j \mid S_j \geq S\} \quad (13)$$

$$P = | GA | / | A | \quad (14)$$

$$P = | GA | / | G | \quad (15)$$

Algorithm Steps:

Step1: collect and classify manually the training literature, mark the relevant literature and irrelevant literature, and divide it randomly into two parts of same amount of literatures. One part will be used to generate the domain querying expression and another part will be simplify the domain querying expression;

Step2: after the literature is pre-treated, choose the nouns and important verbs to extract the domain-specific keywords according to formula (1);

Step3: use the decision tree to generate the original domain querying expression based on the domain-specific keyword;

Step4: simplify the original domain-specific expression;

Remove domain-specific keyword in each conjunction expression in the domain querying expression

Under the condition of no reducing the value of Z in formula (2), no domain-specific keyword can be removed.

Remove the conjunction expression in the domain querying expression

Under the condition of no reducing the value of z, no conjunction can be removed.

Step5: use formula (4) and (5) to weight the domain-specific keyword;

Step6: search the literature through integrating the domain querying expression and user's querying expression and rank the result literature according to the formula (9) and (10).

Evaluate indexing effects and use formula (2), (14) and (15) to calculate the target value of Z.

4.4 Summary

The aim of the research is study principles to build MSEs. Related work has been narrowed to Semantic Web, Creative Computing and Search Technology [83, 84]. Semantic Web techniques can help with inferencing so that the "meta" function can be achieved in the sense of raising concepts to a higher level of abstraction. Creative computing adds elements of creativity in sorting and raking search results. Search technology will be basis for search based which meta search will be conducted. The workflow of the proposed MSE is in Figure 3, where the above three techniques are

integrated. In particular, it introduces the designing ideas and detailed operation process of the search engine in a single domain, Writer's interests. Writer's Portal is divided into four main parts, i.e., extracting domain-specific keyword, setting the value of domain-specific keyword, generating the domain querying expression based on domain-specific keyword and ranking the search results.

Chapter 5. Specific Knowledge and Workflow

Objectives

- To introduce the related professional background knowledge applied on the specific design of the project.
- To introduce the prototype interface design.
- To present the flow chart of prototype.

5.1 Related Professional Background Knowledge

The development of search engine is divided into two parts: the underlying search and client. Writer's Portal uses Google as one of the underlying search engines, and by way of generating field words returned results are compared to determine the correlation and reorder. In this chapter, background knowledge about the client, Ranking methods, as well as the impact of colour psychology on the design of web pages will be introduced.

5.1.1 Background Knowledge on Client Programme

Writer's Requirements for Their Portal

For a number of times, I have communicated with a group of writers led by Professor Kate Pullinger at Bath Spa University, whose needs can be summarised as Funding Opportunities, Dissemination Opportunities, Back Knowledge of Writing, Writing Appreciation and How to Start Writing. These needs form the main targets for meta

search of this study. The writers also supplied a list of websites and these sites should be included in the search.

Client Research and Selection

Client or user programme is a program (web browser) visiting server and providing services to customers. Besides the program running in local machine, it can also be loaded in common clients' computer to cooperate with server [68]. As for research direction of this study, the aim is to finish more efficient search achievement screening by creative design of the search engine server. The first step in operation needed is to have basic knowledge of a server. The next three parts will clarify function of server and design of the proposed meta search engine.

5.1.1.1 Development of Server

As stated in above section, server aims to providing services to users. Then understanding customers' needs becomes crucial, which directly decides its product's accessibility to customers. Customers' requirement will improve convenience of Internet and refinement of industries providing service in Internet. Understanding customers' needs means acquiring opportunities in the market. As for Meta search engine of this project, it acts as a guide for customers' needs and result it displays is the first step to localise users' requirements. At the same time, accuracy of information might be the core competitiveness of search engine. In other words, the aim of this project is to improve accuracy of search achievement. With more and more people acknowledging the importance of customers' needs, increasing kinds of products catering for customers' needs are produced.

5.1.1.2 Development of Client

At the beginning, Cookie is designed in March, 1993 by former employee Lou Montulli in Netscape Communication. Cookie, sometimes has plural forms as cookies, meaning some websites detect customerustomerome websites detect customercustomersome websites detect customerstomerimprove accuracy of sis generated by server and sent to User-agent (usually browser). The latter will record cookie and key/value in texts existing in some catalogues, and next time if customers surf the same website, cookie will be sent to server (the premise is cookie is enabled in browser). The most typical application of cookie is to judge whether customers have registered in the website. They will get prompt about recording their information to simplify entrance next time. All these are functions of cookies. The other important application is used in similar occasions like shopping basket. Users may select different products at the same website during a certain period, and that information will be recorded in cookies to extract information when get payment. The website can follow and summarise customerscustomersome catalogues, and next time if customers surf the same website, cookie will be sent to server (the personalised service to customers, on the other side, it can be a tool for understanding all customer behaviours. It provides reference to improve operation strategy of websites [69]. Even with high utility, cookie may violate users' privacy. Meanwhile, if customers' information recorded at the client is stolen by Hackers, it is likely customers' benefit will be violated on wide scope. As a result, for the problems brought by cookie, related agreements even laws are published to restrict the

development of cookie. On the other hand, people are warned to raise their understand of cookie by some experts to increase Internet security [70].

TELLIM provides multi-media presentation (like audio and video) to attract customers. Consequently, provide personalised online shopping service to customers after confirming their needs [71]. The server will send many presentations to customers, and then check their clicking ratio and duration to send related products. Technologically speaking, dynamic HTML (DHTML) and Java are applied to support server monitoring the client.

5.1.1.3 Client Application and Scheme

With people's increasing acceptance and acknowledgment of online platforms, companies take online advertisement as a mainstream. When refering to advertisement, Doubleclick is used as an example. Its DART Adapt allows any website to manage online advertisement and summarise report by their central server. Thus they can send more targeted advertisement to their objects. The detail process is Doubleclick using cookie named Doubleclick ID to record individual's information. Then they can send advertisement catering for customers' preferences [72].

The research direction of this project is to realise innovative MSE by semantic web concept. Search engine is a typical client application that knows itself and how to acquire information. It is the first step to complete more targeted search engines. First, I need to decide the project direction and its detail application in the field will improve understanding of the project itself.

5.1.2 Ranking Method

In this part, the main content is application of rank method and how to realise it.

5.1.2.1 Function of Ranking Method

When search engine finds out a mass of information, it can display high relative results by understanding different needs. Then how to realise this? After research, rank method is finally found, that is ranking search results to display relevance. It can be visible, for example, search engine ranks on account of customers' habits or interests when displaying results; or it can be invisible, for example, more than hundreds of results are found by using key words, but there may be 10 satisfying needs. In this case, it may rank by the interest relationship with search engine. Meanwhile, application scope of rank method in search engine is various, for example, it needs to rank index entry of expansion words for searching key words, also rank the most related content. The realisation of the ranking method will be introduced in detail in the next chapter.

5.1.2.2 Parameter Adjustment of Algorithm, Over-Fitting and DHC Algorithm

The author can design algorithm with different purposes. He can filtrate and ranking results by algorithms. Existing algorithm like BM25, which is a widely applied query dependent algorithm [73]; and web rank model is an independent algorithm for searching. The most important and difficult part in designing algorithm is parameter setting. The accuracy of parameter will directly decide whether the result will meet

customers' expectations. However, it is usually that parameter cannot accurately reveal market demand. Therefore, in order to resolve this problem, people use over fitting concept. Its meaning is in order to reach a common hypothesis; I need to make the hypothesis over complicated. Simply speaking, over fitting is strict and if the algorithm passed over fitting test, it is more practical in operation [74]. As for the project, the result still needs further screening. At this time, Divisive Hierarchy Clustering (DHC) is brought in. DHC algorithm builds client information according to content by individual bookmark. Clients' information will show their common needs or individual preferences. In the project, DHC algorithm will be applied in screening the last part.

5.2 Design Patterns

In this study, Web App is used to design the search engine of the whole field. General search engines are also presented through Web App. As the optimisation of general search engine, with the same way, using habit of users in general search engine will be continued, more in line with the existing perception of the users. For users, it is easier to accept.

Web App is a kind of application accessed through the network (such as Internet or Intranet); can also refer to the application of the computer software which is carried under the browser support environment or the use of the browser support language (such as JavaScript) and depending on the Web browser to render. Popularity of Web App attributes the success to the popularity of web browsers, as well as the

convenient user's experience upon using this light and thin client. It is not necessary to download and install, and the update and maintenance can be achieved, with the inherent attributes of supporting cross platform, which is the key reason for the popularity of Web App. Typical Web App products include Web mailbox, Web store, wikis, etc. The advantages of Web App covers that browser applications almost do not need the disk space on the client; do not need to install, and new functions are transferred from the server automatically to the users. It is not necessary for the users themselves to upgrade program; it can be easily integrated into other service-class web applications; compatibility cross platform; mobile devices can also use it [75].

At the same time, with the development of Web technology, the gap between user's experience of Web App and App Native is getting smaller and smaller. Browser's support for HTML5, CSS3 and other new standards gets improved gradually. The browser's ability to display and use's experience are also much better [76]. Hence this study uses Web App to realise the field search engine, which can well achieve the research objectives.

5.3 Design Scheme

5.3.1 Design Ideas

Meta search engine is to merge the functions of several general search engines together to provide users with a unified entrance, and meta search engine converts user's query into query content that general search engine can identify, collects

query results obtained by each search engine, and after some analysis and processing then returns to the user [77].

Meta search engine has the following characteristics: 1) there is no huge page index database, and it also does not need the roaming network and web indexing; 2) it can simultaneously link to a number of general search engines, retrieve many index databases, and expand the scope of the query, and consequently information recall rate increases significantly; 3) the meta search engine visits several general search engines at the same time, so the search time is sometimes too long, it is difficult to deal with complex search form, and the sort results are not satisfactory enough.

Take the advantages of meta search engine, consider the relevant retrieval requirements of queried field, add some query expansion technologies into query expressions, and add the word weight information and literature sorting function into the analysis of results, so that you can improve the search effects of meta search engine to construct a simple and efficient field search engine [78].

5.3.2 Detailed Design

This study adopts App Web to code, providing users with a simple and unified entrance, and still following the usage habits of general search engine. The optimisation for the query results is transparent for users. For users, they only input the query results, and then get the query results, and there is no difference with the user's experience of general search engine, while the user’s experience is better.

5.3.2.1 Foreground Design

Page design

The concept of the foreground design is simple and elegant. In addition to the necessary functions on the page, there are no extra things and no additional factors interfering with the users. In the page the users can focus on the search. The technologies used in page design are: HTML, JavaScript and CSS.

Background design

In this study, the main part of background is the generation module of field word expression. It accepts the training document as input, and then outputs the field expression. The specific coding logic and the partition of the modules about background design are described in detail in Section 3.2. So, this section will not repeat them. This section focuses on the core algorithms used in this study.

Decision tree learning algorithm

Decision tree learning algorithm is one of the most practical and widely-used methods for inductive inference, which has a very important theoretical significance and practical value in the artificial intelligence field, such as machine learning, data mining, intelligent control, etc. It has been successfully applied to broad areas, from learning medical diagnosis to learning credit risk assessment of loan applications.

The basic decision tree algorithm is a greedy algorithm, and the existing decision tree learning algorithm is a variant of this core algorithm. The algorithm uses a top-down

and divide-and-rule recursive fashion to search through the possible decision tree space. This method is the basis of the ID3 algorithm and the subsequent C4.5 algorithm. The pseudo code of the decision tree learning algorithm is given here.

Input: training sample set, each attribute value is discrete value [79].

Output: return a decision tree that can correctly classify the training sample set.

Process flow:

The root node of the created decision tree is N.

If all samples are in the same category C, return to N as a leaf node and mark as category C;

Else If Attribute List is blank, the N is returned as a leaf node, and is marked as the category containing the most in the sample in the node;

Else select one of the attributes of the best Example capability from Attribute List, marked as the root node N;

Among For Attribute List each of the known values are marked as V,

According to Attribute = V, a corresponding branch is generated from the root node;

Let S be a subset of the samples obtained with Attribute = V,

If S is empty, then the corresponding leaf node is marked as the category containing the most in the sample in the node;

Else recursively create a sub tree, and call Decision Tree (S, Attribute List)

From this, we can clearly see that the decision tree is a kind of greedy algorithm of the top-down growth tree. In each node, attributes that can best classify the sample are chosen. The process is repeated until it is known that this tree can perfectly classify the training examples, or all of the properties have been used.

The termination conditions of the recursive algorithm are:

All samples corresponding to the root node are in the same category;

If there is no attribute that can be used to partition the current subset of the sample, then use the voting principle, that is, the current node is mandatory for leaf node, and labeled as the dominant category in a sample set of current node;

If there is no sample meeting Attribute = V, then create a leaf node, and it is marked as the dominant category in a sample set of current node [80, 81].

In this study, List Attribute is the set of field words, and the ability of classification training set corresponds to the entropy of information. Through the above principles, field query expression module is the decision tree to use decision tree learning algorithm to generate the field words. In the decision tree, the path with each leaf node value as T is an expression. The conjunctive expressions of these expressions are the generated query expressions of initial field [82].

5.4 Work Flow

At work, firstly use field expression to generate modules, according to the training documents; generate query expression for a specified field; then upon the user's query, directly use generated field query expression to transform the user's query into a conjunctive query; request the general search engine to get the query result; again after the dispose of sort module, namely query structure is returned to the user.

Workflow is shown in the following figure:

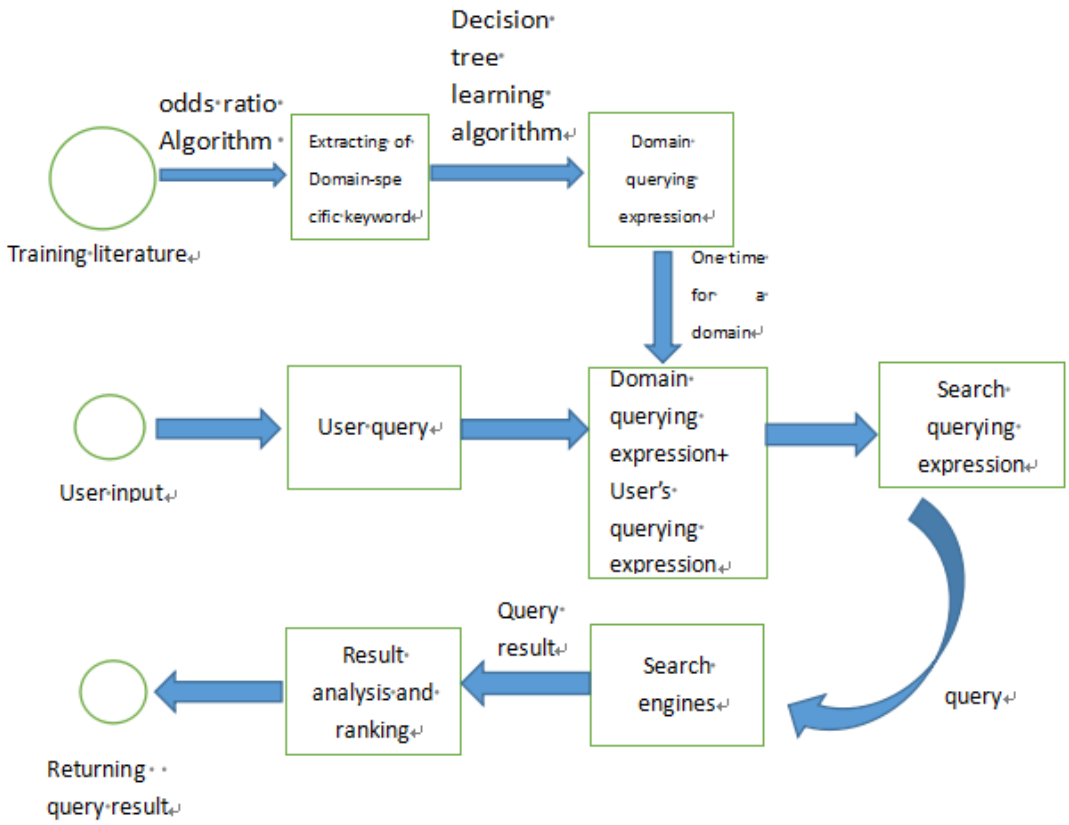


Figure 5. Workflow

5.5 Summary

Techniques used in the development of the Writer's Portal include the following aspects. In terms of programming languages used, HTML and Javascript are the main ones. In terms of the operations, the Writer's prototype consists of the Server programme and user programme. In terms of internal design, the system has the following function components: web data collection (e.g., from Google search engine), data inference (based on Semantic Web) and research result ranking (before returning search results to users). In terms of user Interface design, considerations include colours used (no more than six colours), clarity (eight grids), cheerfulness (using a book as the background) and simplicity (a small of choice buttons and only one input box).

Chapter 6. Experiments

Objectives

- To explain the experiment objectives.
- To explain the functions of models according to different keywords.
- To summarise characteristics of Writer's Portal.

6.1 Introduction of Experiments

Chapter 5 explains the specific implementation of the proposed prototype of meta search engine, Writer's Portal, on the following development platforms:

Hardware

Computer: Macbook Processor: Intel(R) Core(TM) i7-2677M CPU 1.80GHZ Internal storage: 4.00 GB System type: 64 bit operating system.

Software

Operating system: Windows 7 Explorer: Google Chrome, Sogou Explorer Development platform: Eclipse development language: CSS, JavaScript, HTML, and JSON**.**

In the following parts, Writer's Portal will be tested through inputting different keywords and user validation.

URL

http://www.lisiyan.co.uk/wp.html

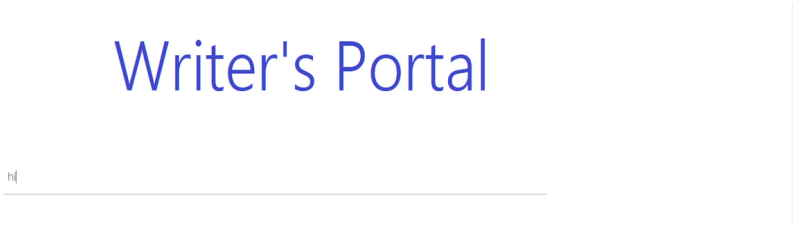
6.2 Demonstrations

Firstly, it introduces the constitution of each part of Writer's Portal briefly:

Figure 6. Home Page of Writer's Portal

As is shown in the above figure, the prototype can be divided into two parts: the logo of Writer's Portal and the search input box when no search keywords had been input.

The following page shows after the input of any keyword at random:



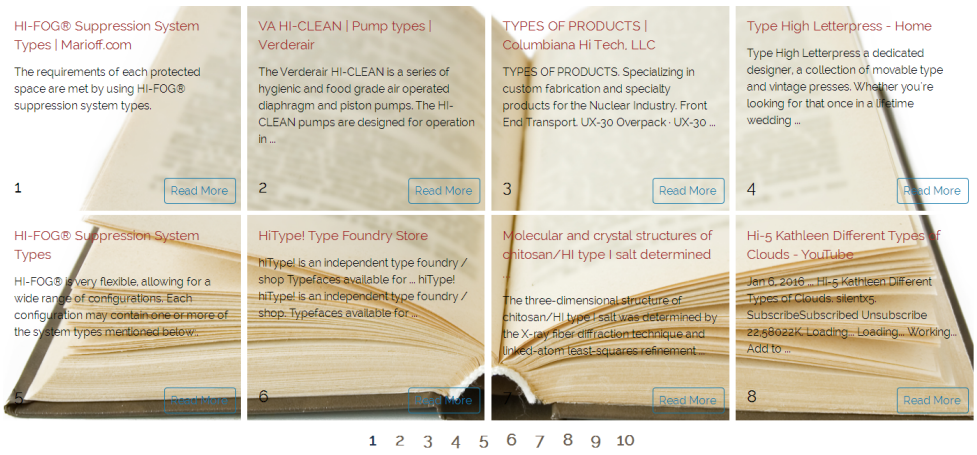


Figure 7. The Page Display of Writer's Portal after Searching

As is shown in the above figure, after searching the keyword “hi” at random, Writer’s Portal searches within the range of the bottom search of Google according to the user’s keyword and the conjunction value of domain expression and ranks the domain expressions according to relevance after the weighted approach. The part of search results is set below the search box and the content of one page can display eight results.

The function of the button of “READ MORE” is the hyperlink of corresponding page.

The numbers in the bottom is the page display. The more front of the page, the higher relevance of the displayed results will be. The background of the search results is set according to the preference of the writer himself or herself. In the end, because of the confinement of the search range, the displayed results are only relevant to the funding opportunities and scholarship Support Submissions.

6.2.1 Searching for Funding

In this part, input the keywords relevant to funding.

After inputting the keyword “Funding”, the page shows as the following figure:

Figure 8. Page Display of Writer's Portal after Searching for "Funding"

Click one search result at random (choose the third item in first page) and the page jumps to:

[image redacted from this digitised version due to potential copyright issues]

Figure 9. Page Displaying Content of Clicking on Third Result in First Page

After the input of the keyword, “fellowship”, the page displays as the follows:

Figure 10. The Page Display of Writer's Portal after Searching "Fellowship"

The page displays as the following figure after the input of the keyword “Award”:

Figure 11. Page Display of Writer's Portal after Searching "Award"

6.2.2 Searching for Dissemination Venues

In this part, input the keywords relevant to Dissemination Venues to search and display the corresponding results:

After inputting the keyword “Publishers”, the page displays as the following figure:

Figure 12. Page Display of Writer's Portal after Searching for "Publishers"

Click one searching result in the figure at random (choose the first item in second page) and the page jumps as the following figure shows:

[image redacted from this digitised version due to potential copyright issues]

Figure 13. Page Content after Clicking the 1st Result of 2nd Page

The page displays as the following figure after the input of the keyword "Competition":

Figure 14. Page Display of Writer's Portal after Searching for "Competition"

The page displays as the following figure after the input of “Media”:

Figure 15. Page Display of Writer's Portal after Searching for "Media"

6.2.3 Searching for Writing Information

In this part, input the keyword relevant to Writing Information to search and display the corresponding results:

The page displays as the follows after the input of the keyword “Background information”:

Figure 16. Page Display of Writer's Portal after Searching for "Background Information"

Click one searching result at random (choose the fourth item in the first page) and the page jumps as the following figure shows:

[image redacted from this digitised version due to potential copyright issues]

Figure 17. Page Content after Clicking the Fourth Result in First Page

The page displays as the following figure after the input of the keyword “figures”:

Figure 18. Page Display of Writer's Portal after Searching for "Figures"

The page displays as the following figure after the input of the keyword “Liberties”:

Figure 19. Page Display of Writer's Portal after Searching for "Libraries"

6.3 Summary

The experiment part of this thesis mainly introduces the single domain search engine, Writer’s Portal, which is based on the creative computing and semantic web idea and the different result display inputs according to different keywords. It is not hard to find that from the corresponding jumping of search results page, Writer’s Portal has some precision and recall in terms of Funding opportunities, scholarship, Support Submissions and Background Writing Information. It reflected pursuit of

user's search interests. Tests have been conducted: the prototype can always supply search results normally with a good coverage of information and can always supply speedy results. Feedback from writers in Professor Pullinger's group was very positive in that the needed information can be easily searched with Writer's Portal.

Chapter 7 Conclusions

Objectives

- To summarise the thesis and draw conclusions.
- To revisit original contributions.
- To evaluate the research by answering the research questions and revisiting the success criteria.
- To illustrate the limitation of the work.
- To propose future work.

7.1 Summary to Thesis

This project proposes the design of a meta search engine for providing various financial supports and writing opportunities for writers according to the requirements of objectives and use semantic we technique in it reasonably. Through the utility of creative computing, semantic web, and programming language, Writer's Portal designed can extract the domain-specific keywords, set the domain-specific keyword weight, use decision tree (DT) to generate the domain querying expression based on domain-specific and uses extended Boolean model (EBM) to rank the searching results. The designed algorithms in this project meet the requirement of the objectives. During the process of searching, the writer's needs for knowledge of various aspects are met. In general, this project fulfils the task requirements proposed at the original design fundamentally.

7.2 Revisiting Original Contributions

The kernel contribution is the purposed mata search engine can provide the writers with more information on Funding opportunities, and Scholarship Support Submissions opportunities, etc. Through the actual test in Chapter 6, searching according to keywords, it is easy to find that Writer's Portal meets the objectives of the projects.

7.3 Answering Research Questions and Revisiting Success Criteria

Evaluations are as follows:

Research questions

How can people establish a meta search engine by using creative computing and the semantic web?

Answer: The designed model is based on the features of creative computing of using the knowledge of various fields to solve the problems and the idea of semantic web so that it uses odds ratio (OR) method to extract the domain-specific keyword; uses TF-IDF (Term Frequency- Inverse Document Frequency) method to set the weight of domain-specific keyword; uses decision tree to generate the domain querying expression based on the domain-specific keyword and uses the extended Boolean model (EBM) to rank the search results.

RQ1: What is a meta search engine with an innovative approach?

Answer: The core connotation of creation lies in the uniqueness and the practicability under the condition of uniqueness. A Meta search engine with

innovative ideas should ensure the practicability of Meta search engine under the condition of uniqueness.

What is the basic structure of setting up meta search engine?

Answer: A meta search engine consists of four fundamental modules of user's interface connection, switching of querying commands, many traditional search engines, and result analysis.

How can people make web pages original and unique?

Answer: The creation can be understood as uniqueness. The uniqueness of web pages can be reflected from the design of the foreground interface of web page and the background operation. The specific methods can be created through the web artistic design and programming.

How to define new ideas and creativity?

Answer: A new idea is the source of creation and the creative idea can be modified through the extraction of different techniques by the known works in specific fields and develops its own creative works. Also, it can be created originally through the different knowledge and experience.

RQ2: What is the proper development approach to achieve the requirement of creativity in the proposed MSE?

Answer: By understanding the creative computing and semantic web practical application, the proposed meta search engine, represented by Writer's Portal, was implemented.

What is the framework of the proposed Meta Search Engine?

Answer: This project builds a search engine with more detailed information compared to the current search engine according to the concept and knowledge of the semantic web. The concept of the semantic web, application of meta-search engine and creation of search results are the scheme and theoretical basis of the whole project. Finally, it will fulfill and realise detail MSE function of this project by comprehension of these three factors.

What are the main phases of the proposed Meta Search Engine?

Answer: MSE consists of four main parts of extracting querying expression, order switching, drawing results from many search engines and result analysis.

What are the fundamental methods and technologies in the proposed Meta Search Engine?

Answer: Principle of semantic web, theoretical basis of creative computing, method and technique of programming technique for completing proposed Meta Search Engine.

RQ3: How to get and manage relevant knowledge?

Answer: This is achieved through studying web pages, books, conference proceedings and learnt journals on techniques and knowledge relevant to MSE, and through organising the knowledge according to semantic web idea and creative computing theoretical basis and programming methods.

Which technologies can be used to support knowledge mining and management?

Answer: Choosing the reasonable research method and choosing for the most efficient knowledge platform are helpful for increasing the efficiency of knowledge elicitation and organisation.

RQ4: How can people reasonably use semantic web knowledge into the creation of search engine?

Answer: Extracting the training literature to train the search content and extracting the domain expressions through decision tree are the reflection of semantic web in this study.

Success criteria

How many domains the proposed meta search engine is suitable to be applied to?

Answer: According to the designing principles of this model, the search range is simplex and it is confined to provide the writers with more information on financial opportunities, Funding opportunities and Scholarship Support Submissions, etc.

How to evaluate creativity of meta search engine according to designed algorithms?

Answer: We can evaluate and verify through the designing principles of algorithm, page display comparison, and search results display comparison.

How to determine the meta search engine in the project is different from the existing search engine in a specific field in terms of functions?

Answer: We can distinguish the search engine in existing specific field according to the difference of search results ranking, page displaying, ranking priority of searching results.

How many creative techniques of reference are applied in designing meta search engine?

Answer: In designing algorithm, the creative computing and semantic web is the creative techniques that are regarded as the theoretical basis. In terms of page designing, the color psychology, as one core creative technique for designing the web page, is applied.

7.4 Limitation of Work

According to the designing principles, the search range is confined to only provide the writers with more information on Funding opportunities and Scholarship Support Submissions, etc. Therefore, whether the keyword input is relevant to the above-mentioned aspects, it will display the results according to the priority ranking of relevance in the design search range. In short, the displayed results have no content beyond the above-mentioned fields.

7.5 Future Work

The Writer's Portal can complete the expected objectives in the current operation and it has acceptable functions in the information of funding opportunity, Dissemination Venues and Writing Information. However, it has some space for improvement in two parts according to the current work:

1. In terms of the ranking of search results, ranking rules can be designed for double or multi-ranking according to different requirements of the users in terms of result relevance, posting time, and click rate.

2. In results display, the accuracy has a space for improvement. The user's feedback algorithm mentioned in Section 3.2.1.3 collects and compares user's click rate for different search results. Other feedbacks may also be used to supply higher accuracy that is based on the extended Boolean model, which is closer to the actual needs of the writers.

Reference

[1]. C. J. Goodwin, *Research in Psychology: Methods and Design,* John Wiley and Sons, Inc, USA, 2005.

[2]. R. K. Yin, *Case Study Research: Design and Methods (Applied Social Research Methods),* SAGE Publications, Inc, 2013.

[3]. D. W. Stewart, *Secondary Research: Information Sources and Methods (Applied Social Research Methods),* SAGE Publications, Inc, 1992.

[4]. B. L. Qiu, *Comparative Study of Educational Modernisation Index System,* Nanjing Normal University, China, 2006.

[5]. J. Zhou, S. Zheng, D. Jing, and H. Yang, "An Approach of Creative Application Evolution on Cloud Computing Platform", *Proceedings of 2011 ACM Symposium on Applied Computing,* ACM, New York, pp. 54–58, 2011.

[6]. H. Yang and L. Zhang, "Controlling and Being Creative: Software Cybernetics and Creative Computing", *IEEE 38th Annual International Computers, Software and Applications Conference Workshops,* pp. 19–24, 2014.

[7]. Z. Song, *Education in the Internet Age,* University of Nankai, China, 2009.

[8]. E. A. Carroll, C. Latulipe, R. Fung and M. Terry, "Creativity Factor Evaluation: Towards A Standardised Survey Metric for Creativity Support", *7th ACM Conference on Creativity and Cognition*, ACM, USA, pp. 127–136, 2009.

[9]. C. Chao, J. Yang, and W. Jen, "Determining Technology Trends and Forecasts of RFID: *A Historical Review and Bibliometric Analysis"*, Technovation, 27(5), pp. 268–279, 2007.

[10]. D. Janssen, T. Schlegel, M. Wissen, and J. Ziegler, "MetaCharts - Using Creativity Methods in A CSCW Environment", *Human-Computer Interaction Theory and Practice (Part II)*, pp. 939–943, 2003.

[11]. L. Nguyen and G. Shanks, "A Framework for Understanding Creativity in Requirements Engineering", *Information and Software Technology*, vol. 51, no. 3, pp. 655–662, 2009.

[12]. T. Berners-Lee, J. Hendler, and O. Lassila. *The Semantic Web. Scientific American,* 284(5): 34–43*,* Springer Press, 2011.

[13]. N. Shadbolt, W. Hall and T. Berners-Lee, "The Semantic Web revisited*", IEEE Intelligent Systems,* IEEE, 2006.

[14]. H. Dietze and M. Schroeder, "GoWeb: A Semantic Search Engine for the Life Science Web*", CEUR Workshop Proceedings (Vol. 435),* Bio Med Central, 2009.

[15]. J. Hendler, "Web 3.0: The Dawn of Semantic Search", *Computer,* 43(1), pp.77–80, IEEE, 2010.

[16]. A. Fallis, "Programming the Semantic Web*", Journal of Chemical Information and Modeling,* (Vol. 53)*,* Cambridge University Press, 2013.

[17]. P. S eredi, G. Lukacsy, T. Benko, and Z. Nagy, *The Semantic Web Explained: The Technology and Mathematics behind Web 3.0,* Cambridge University Press, 2014.

[18]. D. Allemang, *Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL,* Morgan Kaufmann Press, 2001.

[19]. *A. Gulli, and A. Signorini, "*Building An Open Source Meta-Search Engine", *Special Interest Tracks and Posters of the 14th International Conference on World Wide Web* (*WWW '05), pp.1004,* ACM, New York, 2005.

[20]. J. Zhang, and A. Cheung, *"*Meta Search Engine Feature Analysis. Online Information Review", 27(6): 433–441, MCB UP Ltd, 2003.

[21]. R. W. White, M. Richardson, M. Bilenko and A. P. Heath, *"*Enhancing Web Search by Promoting Multiple Search Engine Use*",* SIGIR, 2008.

[22]. A. Zeckman*, "*Google Search Engine Market Share Nears 68%*", Search Engine Watch, http://searchenginewatch.com/article/2345837/Google-Search-Engine-Market-Share-Nears-68,* 2014.

[23]. A. E. Howe, and D. Dreilinger*,* "SavvySearch: A Meta-Search Engine that Learns which Search Engines to Query", *AI Magazine,* 18(2), pp.19–25, 1997.

[24]. M. Beigi, A. B.Benitez , and S. F. Chang, *"*Meta{SEE}k: A Content--Based Meta-Search Engine for Images*", Symposium on Electronic Imaging: Multimedia*

Processing and Applications - Storage and Retrieval for Image and Video Databases VI, IST/SPIE'98, San Jose, CA, pp. 118–128, 1998.

[25]. X. H. Li, Z. Y. Feng, and L. Zhang, "Personalization of Meta-Search Engine"*, Tianjin Daxue Xuebao Ziran Kexue Yu Gongcheng Jishu BanJournal of Tianjin University Science and Technology*, 41(5), pp.616–620, 2008.

[26]. N. Akter, S. H. Khan, and M. Z. Karim, *An Intelligent Meta-Search Engine with Personaisation: Personal Intelligent Meta-Search Engine,* LAP LAMBERT Academic Publishing, 2012.

[27]. S. Bin, L. Yuan, and X. Y. Wang, "Research on Data Mining Models for the Internet of Things", *International Conference on Image Analysis and Signal Processing,* IEEE, pp.127–132, 2012.

[28]. C. W. Tsai, C. F. Lai and M. C. Chiang, and L. T. Yang, "Data Mining for Internet of Things: A Survey", *IEEE Communications Surveys and Tutorials,* 16(1), pp.77–97, 2014.

[29]. S. Muthukrishnan, "Data Management and Mining in Internet and Systems", *Proceedings of the VLDB Endowment, 3(1-2),* pp.1655–1656, 2010.

[30]. H. P. Kriegel, K. M. Borgwardt, P. Kröger, A. Pryakhin, M. Schubert, and A. Zimek, "Future Trends in Data Mining", *Data Mining and Knowledge Discovery,* pp.87–97, Kluwer Academic Publishers-Plenum Publishers, 2007.

[31]. G. Williams, D*ata Mining with Rattle and the Art of Excavating Data for Knowledge Discovery,* Springer Press, 2011.

[32]. M. S. Chen, J. Han, and P. S. Yu, "Data Mining: An Overview from a Database Perspective", *IEEE Transactions on Knowledge and Data Engineering,* IEEE, 1996.

[33]. F. Chen, P. Deng, J. Wan, D. Zhang, A. V. Vasilakos, and X. Rong, *Data Mining for the Internet of Things: Literature Review and Challenges*, International Journal of Distributed Sensor, Networks Press, 2015.

[34]. S. Chakrabarti, "Mining the Web: Discovering Knowledge from Hypertext Data", *Online Information Review*, 27(3), pp.290–291, 2013.

[35]. V. S. Subrahmanian, "Mining Online Opinions", *Computer, 42(7),* pp.88–90, 2009.

[36]. M. Kantardzic, *Advances in Data Mining,* Data Mining Press, http://doi.org/10.1002/9781118029145.ch12, 2011.

[37]. J. Hendler and A. Hugill, "The Syzygy Surfer: Creative Technology for the World Wide Web", *ACM Web Science Conference*, pp. 1–3, 2011.

[38]. H. Yang and L. Zhang, "Controlling and Being Creative: Software Cybernetics and Creative Computing", *IEEE 38th International Computer Software and Applications Conference Workshop,* IEEE, (pp. 19–24, 2014.

[39]. M. Resnick, B. Silverman, Y. Kafai, J. Maloney, A. Monroy-Hernández, N. Rusk and J. Silver, "Scratch", *Communications of the ACM*, *52*(11), 60, ACM New York, 2009.

[40]. M. Resnick, J. Maloney, A. Monroy-Hernández, M. Rusk, E. Eastmond, K. Brennan and Y. Kafai, "Scratch: Programming for ALL", *Communications of the ACM,* 52, pp. 60-67, ACM New York, 2009.

[41]. I. Herman, *W3C Semantic Web Introduction,* http://www.w3.org/2003/Talks/1112-BeijingSW-IH/Chinese/31.html, 2015.

[42]. R. W. White, M. Richardson, M. Bilenko, and A. P. Heath, *Enhancing Web Search by Promoting Multiple Search Engine Use,* SIGIR, 2008.

[43]. Y. Lei, V. Uren, and E. Motta, "SemSearch: A search Engine for the Semantic Web", *15th International Conference,* EKAW, Springer Press, pp.238–245, 2006.

[44]. T. Finin, P. Reddivari, R. S. Cost, and J. Sachs, "Swoogle : A Search and Metadata Engine for the Semantic Web", *ACM Conference on Information and Knowledge Management,* pp.652–659, ACM, New York, 2004.

[45]. M. D'Aquin, E. Motta and T. Watson, "More Than a Semantic Web Search Engine", *Semantic Web,* 2(1), pp.55–63, IOS Press, 2011.

[46]. D. Mukhopadhyay, M. Sharma, G. Joshi, T. Pagare, and A. Palwe, "Experience of Developing a Meta-Semantic Search Engine", *2013 International Conference on Cloud and Ubiquitous Computing and Emerging Technologies,* CUBE 2013, pp. 167–171, IEEE, 2013.

[47]. Z. Zhu, J. Xu, X. Ren, Y. Tian and L. Li, "Query Expansion based on a Personalised Web Search Model", *Third International Conference on Semantics, Knowledge and Grid,* pp. 128-133,2007.

[48]. S. Kumar, L. ErtaAoz, S. Singhal, B. U. Oztekin, E. Han, and V. Kumar, "Personalised Profile Based Search Interface with Ranked and Clustered Display", *Technical report, University of Minnesota, Department of Computer Science,* https://wwws.cs.umn.edu/tech reports/listing/tr2001/01-023.pdf, 2015.

[49]. J. E. Pitkow, H. Schtze, T. A. Cass, R. Cooleym, D. Turnbull, A. Edmonds, E. Adar and T. M. Breuel, "Personalised Search", *Communications of the ACM,* 45(9), pp. 50-55, 2002.

[50]. R. C. Wang, X. L. Xu and H. P. Huang, *Intelligent Agent and its Application in Information Network,* Beijing University of Posts and Telecommunications Press, China, 39(10): 33-36, 2006.

[51]. J. J. Chen，Y. Xun and H. T. Song, "Research and Design of Agent-based Meta Search Engine", *Computer Engineering and Applications*, 39(10), pp.33-36, 2003.

[52]. T. Joachims and L. Granka, *Accurately Interpreting Click through Data Simplicifcation*, SIGIR, 2005.

[53]. B. H. Gong, *Personalized Retrieval Based on User Feedback Technology*, University of Nankai, China, 2007.

[54]. H. H. Li and P. Y. Shao, "Design and Implementation of ASP Infrastructure*", Computer Engineering,* 28(10), pp.254-255, 2002.

[55]. X. Fu, "Trend of the Third Generation of Intelligent Search Engine*"*, *Library and Information Technology*, pp.28-30, 2002.

[56]. J. C. Wang，R. Xiao，Z. X. Sun, et. al., *"*Web Information Retrieval Progress"*, Computer Research and Development, pp.*56-58, 2001.

[57]. A. McCallum, K. Nigam, J. Rennie, and K. Seymore, "A Machine Learning Approach to Building Domain-Specific Search Engines", *16th International Joint Conference on Artificial Intelligence,* San Francisco：M organ Kaufmann Publishers Inc, pp.662-667, 1999.

[58]. J. Chen, "A Framework for Intelligent Meta-search Engine based on Agent", *Third International Conference on Information Technology and Applications,* Washington, pp.276-279, 2005.

[59]. W. Yue, Z. P. Chen and Y. P. Lin, "Information-Retrieval Algorithm Based on Query Expansion and Classification", *Journal of System Simulation,* 18(7), pp.1926-1929, 2006.

[60]. X. Z. Gao, S.Mumgesan and B. Lo, "Extraction of Keyterms by Simple Text Mining for Business Inform ation Retrieval", *IEEE International Conference on E-Business Engineering, W*ashington, pp.332-339, 2005.

[61]. X . W. Sheng and J. M. Hu, "An Information Retrieval System Based 0n Automatic Query Expansion and Hopfield Network", *IEEE International Conference on Neural Networks and Signal Processing* Washington, pp. 1624-1627, 2003.

[62]. P. Calado and M. E. Moura, "Combining Link-based and Content-based Methods for Web Document Classification", *Twelfth International Conference on Information and Knowledge Management,* New York: ACM Press, pp. 540-549, 2006.

[63]. A. A. Macedo, M. G. C. Pimentel and J. A. C. Guerrero, "An Infrastructure for Open Latent Semantic Linking", *Thirteenth ACM conference on Hypertext and hypermedia,* New York: ACM Press, 107-116, 2002.

[64]. M. J. Zheng， Y. P. Lin and Z. P. Chen, "An Information-Retrieval Algorithm based on Classification and Key Phrase Extraction", *Journal of System Simulation*, 16(5), pp. 1009-1012, 2006.

[65]. S. Oyama, T. Kokubo and T. Ishida, "Domain Specific Web Search with Keyword Spices", *IEEE Transactions on Knowledge and Data Engineering,* pp.17-27, 2004.

[66]. Y. Yang and J. P. Pedersen, "A Comparative Study on Feature Selection in Text Categorization", *14th International Conference on Machine Learning,* San Francisco：Morgan Kaufmann Publishers Inc, pp. 412-420, 1997.

[67]. J. R. Quinlan, "Induction of Decision Trees", *Machine Learning,* pp. 81-106, 1986.

[68]. J. Choi, M. Kim and V. V. Raghavan, "Adaptive Relevance Feedback Method of Extended Boolean Model using Hierarchical Clustefing Techniques", *Information Processing and Management：AIL International Journal,* pp.331-349, 2006.

[69]. J. Munro, *Knockout.js: Building Dynamic Client-Side Web Applications,* O'Reilly Media, 2015.

[70]. M. B. Cooley, M. Robert, and J. Srivastava, "Automatic Personalisation based on Web Usage Mining", *Communications of the ACM,* pp. 143-151, 2000.

[71]. J. Schwartz, *Giving Web a Memory Cost its Users Privacy,* New York Times: C1, 2001.

[72]. T. Jörding and S. Michel, "Personalised Shopping in the Web by Monitoring the Customer*", Active Web,* 2001.

[73]. D. Waisberg, *Google Analytics Integrations,* Sybex Press*,* 2015.

[74]. L. Zhang, "*Development and Exploration of based on the HTML5 Web App", Changsha University*, China, pp.50-53, 2015.

[75]. C. G. Lu, "*HTML5 Help Web Application Development Practice", Information Technology and Standardisation*, pp.14-17, 2012.

[76]. G. C. Li, "*Status and Development of Network Search Engine Analysis", Computer Engineering*， 2009.

[77]. J. Pokorny, "Web Searching and Information Retrieval", *IEEE Computer Society,* 2004.

[78]. S. Oyama, "Qurey Refinement for Domain Specific Web Search", *Kyotou Doctor Thesis,* 2002.

[79]. T. M. Mitchell, *Machine Learning*, USA: McGraw-Hill*,* pp.184-185, 1997.

[80]. T. Oates and D. Jemen, "The Effects of Training Set On decision Tree", *14th International Conference on Machine Learnin*g, Nashville：Morgan Kattfman, pp.254-262, 2011.

[81]. S. Mukherjea, *"*WTMS: A System for Collecting and Analysing Topic-Specific Web Information", *9th International World Wide Web Conference* New York: ACM Press, pp.245-253, 2000.

[82]. S. Oyama, T. Kokubo, and T.Ishida, "Domain-Specific Web Search with Keyword Spices" *IEEE Transactions on Knowledge and Data Engineering*, pp. 17-27, 2004.

[83]. S. Ma, S. Li and H. Yang, "Creative Computing for Personalised Meta-Search Engine Based on Semantic Web", *21st International Conference on Automation and Computing, IEEE, 2015*.

[84]. S. Ma, S. Liand H. Yang, "Utilising Creative Computing and Data Mining Techniques to Analyse Queries in A Meta-Search System", 22nd *International Conference on Automation and Computing,* IEEE*, 2016*.

Appendix 1: Sample Ontology Items

</owl:Class><owl:Class
rdf:about="http://dbpedia.org/ontology/University">
<rdfs:label xml:lang="el">πανεπιστήμιο
</rdfs:label><rdfs:label xml:lang="fr">université</rdfs:label><rdfs:label
xml:lang="en">university</rdfs:label><rdfs:label xml:lang="ja">University
</rdfs:label><rdfs:label xml:lang="nl">universiteit</rdfs:label><rdfs:label

xml:lang="ko">대학</rdfs:label><rdfs:label

xml:lang="pt">universidade</rdfs:label><rdfs:label
xml:lang="es">universidad</rdfs:label><rdfs:label
xml:lang="de">Universität</rdfs:label><rdfs:subClassOf
rdf:resource="http://dbpedia.org/ontology/EducationalInstitution"/><owl:equivalen
tClass rdf:resource="http://schema.org/CollegeOrUniversity"/>
</owl:Class><owl:Class rdf:about="http://dbpedia.org/ontology/Magazine">
<rdfs:label xml:lang="el">Περιοδικό</rdfs:label><rdfs:label
xml:lang="fr">magazine</rdfs:label><rdfs:label
xml:lang="en">magazine</rdfs:label><rdfs:label
xml:lang="ja">magazine</rdfs:label><rdfs:label
xml:lang="ko"> </rdfs:label><rdfs:label
xml:lang="de">Publikumszeitschrift</rdfs:label><rdfs:comment
xml:lang="en">Magazines, periodicals, glossies or serials are publications, generally
published on a regular schedule, containing a variety of articles. They are generally
financed by advertising, by a purchase price, by pre-paid magazine subscriptions, or
all three.</rdfs:comment><rdfs:comment xml:lang="el">Περιοδικά ή γυαλιστερές φωτογραφίες περιοδικών εκδόσεων δημοσιεύονται σε τακτά χρονικά διαστήματα, περιέχει μια ποικιλία από αντικείμενα.Γενικά χρηματοδοτείται από διαφημίσεις, με τιμή αγοράς, με προπληρωμένες συνδρομές περιοδικών, ή και των τριών.</rdfs:comment><rdfs:comment xml:lang="de">Als Publikumszeitschrift (auch
Magazin) bezeichnet man eine Gattung von Zeitschriften, die sich an eine sehr breite
Zielgruppe wendet und keine fachlichen Prägungen oder andere spezifische
Merkmale voraussetzt. Publikumszeitschriften dienen der Unterhaltung und
Information, sie sollen unangestrengt gelesen werden
können.</rdfs:comment><rdfs:subClassOf
rdf:resource="http://dbpedia.org/ontology/PeriodicalLiterature"/>
</owl:Class><owl:Class rdf:about="http://dbpedia.org/ontology/Galaxy">
<rdfs:label xml:lang="el">γαλαξίας</rdfs:label><rdfs:label
xml:lang="fr">galaxie</rdfs:label><rdfs:label
xml:lang="en">galaxy</rdfs:label><rdfs:label

xml:lang="tr">galaksi</rdfs:label><rdfs:label
xml:lang="ja">Galaxy</rdfs:label><rdfs:label
xml:lang="pt">galáxia</rdfs:label><rdfs:subClassOf
rdf:resource="http://dbpedia.org/ontology/CelestialBody"/>
</owl:Class><owl:Class
rdf:about="http://dbpedia.org/ontology/Manhwa">
</owl:Class><owl:Class rdf:about="http://dbpedia.org/ontology/CollegeCoach">
<rdfs:label xml:lang="en">college coach</rdfs:label><rdfs:label
xml:lang="el">προπονητής κολεγίου</rdfs:label><rdfs:label
xml:lang="fr">entraîneur universitaire</rdfs:label><rdfs:subClassOf
rdf:resource="http://dbpedia.org/ontology/Person"/>
</owl:Class><owl:Class rdf:about="http://dbpedia.org/ontology/Writer">
<rdfs:label xml:lang="el">συγγραφέας</rdfs:label><rdfs:label
xml:lang="fr">écrivain</rdfs:label><rdfs:label
xml:lang="en">writer</rdfs:label><rdfs:label xml:lang="ja">Authors
</rdfs:label><rdfs:label xml:lang="nl">auteur</rdfs:label><rdfs:label

xml:lang="ko">작가</rdfs:label><rdfs:label

xml:lang="es">escritor</rdfs:label><rdfs:label
xml:lang="de">schriftsteller</rdfs:label><rdfs:subClassOf
rdf:resource="http://dbpedia.org/ontology/Artist"/>
<rdfs:label xml:lang="el">κινούμενα σχέδια
</rdfs:label><rdfs:label xml:lang="fr">manga</rdfs:label><rdfs:label
xml:lang="en">manga</rdfs:label><rdfs:label
xml:lang="it">manga</rdfs:label><rdfs:label
xml:lang="ja"></rdfs:label><rdfs:comment xml:lang="en">Manga are comics created
in Japan</rdfs:comment><rdfs:subClassOf
rdf:resource="http://dbpedia.org/ontology/Comics"/>
</owl:Class><owl:Class rdf:about="http://dbpedia.org/ontology/College">
<rdfs:label xml:lang="el">μουσείο</rdfs:label><rdfs:label
xml:lang="fr">musée</rdfs:label><rdfs:label
xml:lang="en">museum</rdfs:label><rdfs:label
xml:lang="ja">museum</rdfs:label><rdfs:label
xml:lang="nl">museum</rdfs:label><rdfs:label
xml:lang="ko"> </rdfs:label><rdfs:subClassOf
rdf:resource="http://dbpedia.org/ontology/Building"/><owl:equivalentClass
rdf:resource="http://schema.org/Museum"/>
</owl:Class><owl:Class
rdf:about="http://dbpedia.org/ontology/FigureSkater">
<rdfs:label xml:lang="en">fashion designer</rdfs:label><rdfs:label xml:lang="el">
σχεδιαστής μόδας</rdfs:label><rdfs:subClassOf
rdf:resource="http://dbpedia.org/ontology/Artist"/>
</owl:Class><owl:Class rdf:about="http://dbpedia.org/ontology/Library">

<rdfs:label xml:lang="en">anime</rdfs:label><rdfs:label xml:lang="el">άνιμε</rdfs:label><rdfs:label xml:lang="it">anime</rdfs:label><rdfs:label xml:lang="ja">アニメ</rdfs:label><rdfs:comment xml:lang="en">A style of animation originating in Japan</rdfs:comment><rdfs:comment xml:lang="el">Στυλ κινουμένων σχεδίων με καταγωγή την Ιαπωνία</rdfs:comment><rdfs:subClassOf rdf:resource="http://dbpedia.org/ontology/Cartoon"/>
</owl:Class><owl:Class rdf:about="http://dbpedia.org/ontology/Book">
<rdfs:label xml:lang="en">synagog</rdfs:label><rdfs:label xml:lang="fr">synagogue</rdfs:label><rdfs:label xml:lang="es">sinagoga</rdfs:label><rdfs:label xml:lang="nl">synagoge</rdfs:label><rdfs:comment xml:lang="en">A synagogue, sometimes spelt synagog, is a Jewish or Samaritan house of prayer.</rdfs:comment><rdfs:comment xml:lang="fr">Une synagogue est un lieu de culte juif.</rdfs:comment><rdfs:subClassOf rdf:resource="http://dbpedia.org/ontology/ReligiousBuilding"/>
</owl:Class><owl:Class rdf:about="http://dbpedia.org/ontology/Project">
<rdfs:label xml:lang="el">δικαστής</rdfs:label><rdfs:label xml:lang="fr">juge</rdfs:label><rdfs:label xml:lang="en">judge</rdfs:label><rdfs:label xml:lang="it">giudice</rdfs:label><rdfs:label xml:lang="ja">Judge</rdfs:label><rdfs:label xml:lang="nl">rechter</rdfs:label><rdfs:label xml:lang="de">richter</rdfs:label><rdfs:subClassOf rdf:resource="http://dbpedia.org/ontology/Person"/>
</owl:Class><owl:Class rdf:about="http://dbpedia.org/ontology/Ship">
<rdfs:label xml:lang="el">πλοίο</rdfs:label><rdfs:label xml:lang="fr">navire</rdfs:label><rdfs:label xml:lang="en">ship</rdfs:label><rdfs:label xml:lang="ja">ship</rdfs:label><rdfs:label xml:lang="nl">schip</rdfs:label><rdfs:label xml:lang="ko"> </rdfs:label><rdfs:label xml:lang="pl">statek</rdfs:label><rdfs:label xml:lang="es">barco</rdfs:label><rdfs:label xml:lang="de">Schiff</rdfs:label><rdfs:subClassOf rdf:resource="http://dbpedia.org/ontology/MeanOfTransportation"/><rdfs:subClassOf rdf:resource="http://schema.org/Product"/>
</owl:Class><owl:Class rdf:about="http://dbpedia.org/ontology/Award">
<rdfs:label xml:lang="en">hollywood cartoon</rdfs:label><rdfs:label xml:lang="el">κινούμενα σχέδια του Hollywood</rdfs:label><rdfs:subClassOf rdf:resource="http://dbpedia.org/ontology/Cartoon"/>
</owl:Class><owl:Class rdf:about="http://dbpedia.org/ontology/Musical">
<rdfs:label xml:lang="en">sport facility</rdfs:label><rdfs:label xml:lang="el">αθλητικές εγκαταστάσεις</rdfs:label><rdfs:label

xml:lang="fr">installation sportive</rdfs:label><rdfs:subClassOf
rdf:resource="http://dbpedia.org/ontology/Place"/>
</owl:Class><owl:Class rdf:about="http://dbpedia.org/ontology/Agent">
<rdfs:label xml:lang="el">Μύλος</rdfs:label><rdfs:label
xml:lang="fr">Moulin</rdfs:label><rdfs:label
xml:lang="en">Mill</rdfs:label><rdfs:label
xml:lang="it">mulino</rdfs:label><rdfs:label
xml:lang="nl">Molen</rdfs:label><rdfs:label
xml:lang="de">Mühle</rdfs:label><rdfs:comment xml:lang="en">a unit operation
designed to break a solid material into smaller
pieces</rdfs:comment><rdfs:subClassOf
rdf:resource="http://dbpedia.org/ontology/Building"/>
</owl:Class><owl:Class
rdf:about="http://dbpedia.org/ontology/Painting">
<rdfs:label xml:lang="el">Στρατιωτική Μονάδα
</rdfs:label><rdfs:label xml:lang="fr">unité militaire</rdfs:label><rdfs:label
xml:lang="en">military unit</rdfs:label><rdfs:label
xml:lang="ko"> </rdfs:label><rdfs:label xml:lang="pt">unidade
militar</rdfs:label><rdfs:label xml:lang="es">unidad
militar</rdfs:label><rdfs:subClassOf
rdf:resource="http://dbpedia.org/ontology/Organisation"/>
</owl:Class><owl:Class
rdf:about="http://dbpedia.org/ontology/PublicService">
<rdfs:label xml:lang="el">μνημείο</rdfs:label><rdfs:label
xml:lang="fr">monument</rdfs:label><rdfs:label
xml:lang="en">monument</rdfs:label><rdfs:label
xml:lang="nl">monument</rdfs:label><rdfs:label
xml:lang="de">Denkmal</rdfs:label><rdfs:comment xml:lang="en">A type of
structure (a statue or an art object) created to commemorate a person or important
event, not necessarily of a catastrophic nature.</rdfs:comment><rdfs:subClassOf
rdf:resource="http://dbpedia.org/ontology/Place"/>
</owl:Class><owl:Class rdf:about="http://dbpedia.org/ontology/Artist">
<rdfs:label xml:lang="en">Gaelic games player</rdfs:label><rdfs:label
xml:lang="el">Γαελικός παίκτης παιχνιδιών
</rdfs:label><rdfs:label xml:lang="fr">joueur de sports
gaéliques</rdfs:label><rdfs:subClassOf
rdf:resource="http://dbpedia.org/ontology/Athlete"/>
</owl:Class><owl:Class rdf:about="http://dbpedia.org/ontology/Novel">
<rdfs:label xml:lang="en">eukaryote</rdfs:label><rdfs:label xml:lang="el">
ευκαρυωτικό</rdfs:label><rdfs:label
xml:lang="fr">eucaryote</rdfs:label><rdfs:label
xml:lang="ja">Species</rdfs:label><rdfs:subClassOf
rdf:resource="http://dbpedia.org/ontology/Species"/>

```
        </owl:Class><owl:Class
rdf:about="http://dbpedia.org/ontology/Professor">
    <rdfs:label xml:lang="el">επιστήμονας</rdfs:label><rdfs:label
xml:lang="fr">scientifique</rdfs:label><rdfs:label
xml:lang="en">scientist</rdfs:label><rdfs:label
xml:lang="ja">scientist</rdfs:label><rdfs:label
xml:lang="nl">wetenschapper</rdfs:label><rdfs:label
xml:lang="ko">          </rdfs:label><rdfs:label
xml:lang="bn">□□□□□□□□□</rdfs:label><rdfs:label             (((((?????)))))
 xml:lang="de">Wissenschaftler</rdfs:label><rdfs:subClassOf
rdf:resource="http://dbpedia.org/ontology/Person"/>
        </owl:Class><owl:Class
rdf:about="http://dbpedia.org/ontology/Website">
        <rdfs:label xml:lang="el">εγκληματίας</rdfs:label><rdfs:label
xml:lang="fr">criminel</rdfs:label><rdfs:label
xml:lang="en">criminal</rdfs:label><rdfs:label
xml:lang="it">delinquente</rdfs:label><rdfs:label
xml:lang="nl">crimineel</rdfs:label><rdfs:label
xml:lang="ko">          </rdfs:label><rdfs:label
xml:lang="pt">criminoso</rdfs:label><rdfs:label
xml:lang="de">Verbrecher</rdfs:label><rdfs:subClassOf
rdf:resource="http://dbpedia.org/ontology/Person"/>
        </owl:Class><owl:Class
rdf:about="http://dbpedia.org/ontology/Surname">
<rdfs:label xml:lang="en">minority of a settlement</rdfs:label><rdfs:domain
rdf:resource="http://dbpedia.org/ontology/Settlement"/><rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
                </owl:DatatypeProperty><owl:DatatypeProperty
rdf:about="http://dbpedia.org/ontology/reference">
<rdfs:label xml:lang="en">massif</rdfs:label><rdfs:domain
rdf:resource="http://dbpedia.org/ontology/SkiResort"/><rdfs:range
rdf:resource="http://dbpedia.org/ontology/Place"/>
                </owl:ObjectProperty><owl:ObjectProperty
rdf:about="http://dbpedia.org/ontology/britishComedyAwards">
          <rdfs:label xml:lang="en">administrative
collectivity</rdfs:label><rdfs:label xml:lang="nl">administratieve
gemeenschap</rdfs:label><rdfs:label xml:lang="el">διοικητική συλλογικότητα</rdfs:label><rdfs:domain
rdf:resource="http://dbpedia.org/ontology/Settlement"/><rdfs:range
rdf:resource="http://dbpedia.org/ontology/PopulatedPlace"/>
                </owl:ObjectProperty><owl:ObjectProperty
rdf:about="http://dbpedia.org/ontology/chiefEditor">
```

<rdfs:label xml:lang="en">military unit size</rdfs:label><rdfs:comment xml:lang="en">the size of the military unit</rdfs:comment><rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty><owl:ObjectProperty rdf:about="http://dbpedia.org/ontology/author">
<rdfs:label xml:lang="en">formation date</rdfs:label><rdfs:label xml:lang="nl">formatie datum</rdfs:label><rdfs:label xml:lang="el">Ιδρύθηκε</rdfs:label><rdfs:domain rdf:resource="http://dbpedia.org/ontology/Organisation"/><rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#date"/>
</owl:DatatypeProperty><owl:DatatypeProperty rdf:about="http://dbpedia.org/ontology/course">
<rdfs:label xml:lang="el">πρόεδρος</rdfs:label><rdfs:label xml:lang="en">president</rdfs:label><rdfs:label xml:lang="nl">president</rdfs:label><rdfs:label xml:lang="pt">presidente</rdfs:label><rdfs:label xml:lang="de">Präsident</rdfs:label><rdfs:range rdf:resource="http://dbpedia.org/ontology/Person"/>
</owl:ObjectProperty><owl:ObjectProperty rdf:about="http://dbpedia.org/ontology/storyEditor">
<rdfs:label xml:lang="en">picture format</rdfs:label><rdfs:label xml:lang="de">Bildformat</rdfs:label><rdfs:domain rdf:resource="http://dbpedia.org/ontology/Broadcaster"/>
</owl:ObjectProperty><owl:ObjectProperty rdf:about="http://dbpedia.org/ontology/show">
<rdfs:label xml:lang="en">musicians</rdfs:label><rdfs:label xml:lang="el">μουσικοί</rdfs:label><rdfs:domain rdf:resource="http://dbpedia.org/ontology/Instrument"/><rdfs:range rdf:resource="http://dbpedia.org/ontology/MusicalArtist"/>
</owl:ObjectProperty><owl:ObjectProperty rdf:about="http://dbpedia.org/ontology/student">
<rdfs:label xml:lang="en">classes</rdfs:label><rdfs:label xml:lang="el">τάξεις</rdfs:label><rdfs:domain rdf:resource="http://dbpedia.org/ontology/School"/><rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#nonNegativeInteger"/>
</owl:DatatypeProperty><owl:ObjectProperty rdf:about="http://dbpedia.org/ontology/writer">
<rdfs:label xml:lang="en">monument code (municipal)</rdfs:label><rdfs:label xml:lang="nl">monumentcode gemeentelijke monumenten</rdfs:label><rdfs:comment xml:lang="en">We should be able to distinguish status types since different codes refer to different protection regimes.</rdfs:comment><rdfs:comment xml:lang="nl">We moeten soorten codes kunnen onderscheiden al naar gelang het een rijks-, provinciaal of gemeentelijk monument, of een MIP-code, betreft</rdfs:comment><rdfs:domain

rdf:resource="http://dbpedia.org/ontology/Place"/><rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty><owl:ObjectProperty
rdf:about="http://dbpedia.org/ontology/previousEvent">

Appendix 2: Program Code (Inference Engine)

```
<!doctype html>
<html lang="en-GB">
<!--[if lt IE 7]>
<html class="no-js lt-ie9 lt-ie8 lt-ie7" lang="en-GB"> <![endif]-->
<!--[if IE 7]>
<html class="no-js lt-ie9 lt-ie8" lang="en-GB"> <![endif]-->
<!--[if IE 8]>
<html class="no-js lt-ie9" lang="en-GB"> <![endif]-->
<!--[if gt IE 8]><!-->
<html class="no-js" lang="en-GB">
    <!--<![endif]-->
    <head>
        <meta charset="utf-8">
        <title></title>
        <meta name="description" content="">
        <meta name="keywords" content="">
        <meta name="author" content="">
        <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <link rel="stylesheet" href="css/gfont.css" />
        <link rel="stylesheet" href="css/bootstrap.min.css" />
        <link rel="stylesheet" href="css/bootstrap-theme.min.css" />
        <link rel="stylesheet" href="css/template.css"/>
    </head>
    <body class="results-page">
        <header>
<!--                <h1 class="identity"><a href="#">Writer's Portal</a></h1>-->
<!--                <h2 class="parent-id"><a href="#">Bath Spa University</a></h2>-->
<div align="center">
    <img src="./{@LIZIHUW}5LUIU}$ZE@3IL.png" style="width:600px; height:150px;"/>
    <div id="cse" style="width:50%;"></div>
        </header>
        <div class="search">
```

```
                <input type="search" placeholder="Search..." id="search-box" spellcheck="false" class="entry" name="q" size="50" />

                <button type="button" id="search-btn" style="position: absolute; left: -9999px" onclick="search(1);">Search</button>

            </div>

<!--            <div>-->

<!--                <ul class="categories pie-clearfix">-->

<!--                    <li class="all">-->

<!--                        <a href="javascript: void(0);" onclick="changeSearch('Open Submissions');" id="searchAll">Open Submissions</a>-->

<!--                    </li>-->

<!--                    <li class="Fundings">-->

<!--                                                <a href="javascript: void(0);" onclick="changeSearch('Fundings');" id="searchFundings">Fundings</a>-->

<!--                    </li>-->

<!--                    <li class="Bursaies">-->

<!--                                                <a href="javascript: void(0);" onclick="changeSearch('Bursaies');" id="searchBursaies">Bursaies</a>-->

<!--                    </li>-->

<!--                    <li class="Commissions">-->

<!--                                                <a href="javascript: void(0);" onclick="changeSearch('Commissions');" id="searchCommissions">Commissions</a>-->

<!--                    </li>-->

<!--                    <li class="Jobs">-->

<!--                        <a href="javascript: void(0);" onclick="changeSearch('Jobs');" id="searchJobs">Jobs</a>-->

<!--                    </li>-->

<!--                      <li class="Competitions">-->

<!--                                                <a href="javascript: void(0);" onclick="changeSearch('Competitions');" id="searchCompetitions">Competitions</a>-->

<!--                    </li>-->

<!--                      <li class="Scholarships">-->

<!--                                                <a href="javascript: void(0);" onclick="changeSearch('Scholarships');" id="searchScholarships">Scholarships</a>-->

<!--                    </li>-->

<!--                      <li class="Organisations">-->

<!--                                                <a href="javascript: void(0);" onclick="changeSearch('Organisations');" id="searchOrganisations">Organisations</a>-->

<!--                    </li>-->

<!--                      <li class="FundedResidencies">-->

<!--                                                <a href="javascript: void(0);" onclick="changeSearch('FundedResidencies');" id="searchFundedResidencies">Funded Residencies</a>-->
```

```
<!--                        </li>-->
<!--                        -->
<!--                    </ul>-->
<!--                </div>-->
                    <!-- Single button -->
                <div id="searchResult"></div>
                <div class="owl-dots" id="pages">
                </div>
                    <!-- Modal -->
<div class="modal" id="cart" tabindex="-1" role="dialog" aria-labelledby="myModalLabel" aria-hidden="true">
    <div class="modal-dialog fadeIn">
        <div class="modal-content">
            <div class="modal-header">
                <button type="button" class="close" data-dismiss="modal"><span aria-hidden="true">&times;</span><span
                        class="sr-only">Close</span></button>
                <h4 class="modal-title" id="myModalLabel">Your Saved Searches</h4>
            </div>
            <div class="modal-body">
                <table class="table table-responsive" id="citations">
                    <thead>
                    <tr>
                        <th>Link</th>
                        <th>Accessed</th>
                    </tr>
                    </thead>
                    <tbody id="citationBody">
                    </tbody>
                </table>
            </div>
        </div>
    </div>
</div>
<div class="modal" id="export" tabindex="-1" role="dialog" aria-labelledby="exportLabel" aria-hidden="true">
    <div class="modal-dialog fadeIn">
        <div class="modal-content">
```

```
                <div class="modal-header">
                    <button type="button" class="close" data-dismiss="modal"><span aria-hidden="true">&times;</span><span
                            class="sr-only">Close</span></button>
                    <h4 class="modal-title" id="exportLabel">Export Your Saved Searches</h4>
                </div>
                <div class="modal-body">
                    ... options ...
                </div>
                <div class="modal-footer">
                    <button type="button" class="btn btn-default" data-dismiss="modal">Close</button>
                    <button type="button" class="btn btn-primary" id="download">Download</button>
                </div>
            </div>
        </div>
    </div>
    <div class="modal" id="extra-1" tabindex="-1" role="dialog" aria-labelledby="extra-1-label" aria-hidden="true">
        <div class="modal-dialog flipInX">
            <div class="modal-content">
                <div class="modal-header">
                    <button type="button" class="close" data-dismiss="modal"><span aria-hidden="true">&times;</span><span
                            class="sr-only">Close</span></button>
                    <h4 class="modal-title" id="extra-1-label">Human-Computer-Interaction</h4>
                </div>
                <div class="modal-body">
                    Author's: Jenny Preece, Yvonne Rogers, Helen Sharp, David Benyon, Simon Holland, Tom Carey
                </div>
                <div class="modal-footer">
                    <button type="button" class="btn btn-default close-button">Close</button>
                    <button type="button" class="btn btn-primary cite">Cite</button>
                </div>
            </div>
        </div>
```

```
</div>
<div class="modal" id="extra-2" tabindex="-1" role="dialog" aria-labelledby="extra-2-label" aria-hidden="true">
    <div class="modal-dialog flipInX">
        <div class="modal-content">
            <div class="modal-header">
                <button type="button" class="close" data-dismiss="modal"><span aria-hidden="true">&times;</span><span
                        class="sr-only">Close</span></button>
                <h4 class="modal-title" id="extra-2-label">An Investigation of the Theory and Practice of Fault-Tolerant Computer Design</h4>
            </div>
            <div class="modal-body">
                An Investigation of the Theory and Practice of Fault-Tolerant Computer Design
            </div>
            <div class="modal-footer">
                <button type="button" class="btn btn-default close-button">Close</button>
                <button type="button" class="btn btn-primary cite">Cite</button>
            </div>
        </div>
    </div>
</div>
<div class="modal" id="extra-3" tabindex="-1" role="dialog" aria-labelledby="extra-3-label" aria-hidden="true">
    <div class="modal-dialog flipInX">
        <div class="modal-content">
            <div class="modal-header">
                <button type="button" class="close" data-dismiss="modal"><span aria-hidden="true">&times;</span><span
                        class="sr-only">Close</span></button>
                <h4 class="modal-title" id="extra-3-label">How to Make a Multiprocessor Computer That Correctly Executes Multiprocess Programs</h4>
            </div>
            <div class="modal-body">
                How to Make a Multiprocessor Computer That Correctly Executes Multiprocess Programs
            </div>
            <div class="modal-footer">
                <button type="button" class="btn btn-default close-button">Close</button>
```

```
                    <button type="button" class="btn btn-primary cite">Cite</button>
                </div>
            </div>
        </div>
    </div>
    </body>
        <!--js framework -->
        <script src="js/jquery.min.js"></script>
        <script src="js/bootstrap.min.js"></script>
        <script src="js/jquery.embedly-3.1.1.min.js" type="text/javascript"></script>
        <script src="js/MetaEngine.sort.js" type="text/javascript"></script>
        <script src="js/MetaEngine.keywords.js" type="text/javascript"></script>
        <!--        <script src="js/script.min.js"></script>-->
            <script>
                $("#citationNumber").text(getCitations().length);
                $("#searchAll").addClass('fontBold');
                var theCustomSearchEngineId = '008468171151941459668:ds0fnjd03ay';
                var theApiKey = 'AIzaSyAQE9jPxnjYputbIGp-bPgorq8mGvTFsaQ';
//'AIzaSyAQE9jPxnjYputbIGp-bPgorq8mGvTFsaQ';
                var theRequestedTotalNumber = 8;
                var colorCount=0;
                var sortCondition = '';
                $("#sortRelevance").css("font-weight", "bold");
                $("#sortDate").css("font-weight", "normal");
                $("#sortPopularity").css("font-weight", "normal");
                // Bind the enter key to start a search
                $('#search-box').bind('keypress', function (e) {
                    var code = (e.keyCode ? e.keyCode : e.which);
                    if (code == 13) {
                        //Enter key code
                        //Do something
                        $('#search-btn').click();
                    }
                });
                // Get number of citations from local storage
                function getCitations() {
                    var localCitations = localStorage.citations;
```

```
                if (localCitations) {
                    var localCitationArray = JSON.parse(localCitations);
                    return localCitationArray;
                } else {
                    return [];
                }
            }
            function getUrl(queryString, startIndex) {
                var theRestUrl = 'https://www.googleapis.com/customsearch/v1?q=' +
queryString
                    + '&cx=' + theCustomSearchEngineId
                    + '&key=' + theApiKey
                    + '&num=' + theRequestedTotalNumber
                    + '&start=' + startIndex
                    + sortCondition;
                return theRestUrl;
            }
            function changeSearch(customSearchEngineId) {
                switch (customSearchEngineId) {
                case 'Bursaies':
                    theCustomSearchEngineId =
'003134293120551003557:3w-4tbhp7c4';
                    $("#searchBursaies").addClass('fontBold');
                    $("#searchFundings").removeClass('fontBold');
                    $("#searchCommissions").removeClass('fontBold');
                    $("#searchJobs").removeClass('fontBold');
                      $("#searchCompetitions").removeClass('fontBold');
                    $("#searchScholarships").removeClass('fontBold');
                    $("#searchOrganisations").removeClass('fontBold');
                      $("#searchFundedResidencies").removeClass('fontBold');
                    $("#searchAll").removeClass('fontBold');
                      search(1);
                    break;
                case 'Fundings':
                    theCustomSearchEngineId =
'003134293120551003557:u1wxkgpiu8o';
                    $("#searchFundings").addClass('fontBold');
                    $("#searchBursaies").removeClass('fontBold');
```

```
                $("#searchCommissions").removeClass('fontBold');
                  $("#searchJobs").removeClass('fontBold');
                  $("#searchCompetitions").removeClass('fontBold');
                $("#searchScholarships").removeClass('fontBold');
                $("#searchOrganisations").removeClass('fontBold');
                  $("#searchFundedResidencies").removeClass('fontBold');
                $("#searchAll").removeClass('fontBold');
                  search(1);
                break;
            case 'Commissions':
                theCustomSearchEngineId                                    =
'003134293120551003557:yc-gyfsbraq';
                $("#searchCommissions").addClass('fontBold');
                $("#searchFundings").removeClass('fontBold');
                $("#searchBursaies").removeClass('fontBold');
                  $("#searchJobs").removeClass('fontBold');
                  $("#searchCompetitions").removeClass('fontBold');
                $("#searchScholarships").removeClass('fontBold');
                $("#searchOrganisations").removeClass('fontBold');
                  $("#searchFundedResidencies").removeClass('fontBold');
                $("#searchAll").removeClass('fontBold');
                  search(1);
                break;
               case 'Jobs':
                theCustomSearchEngineId                                    =
'003134293120551003557:x8z5xrpneai';
                $("#searchJobs").addClass('fontBold');
                $("#searchFundings").removeClass('fontBold');
                $("#searchBursaies").removeClass('fontBold');
                  $("#searchCommissions").removeClass('fontBold');
                  $("#searchCompetitions").removeClass('fontBold');
                $("#searchScholarships").removeClass('fontBold');
                $("#searchOrganisations").removeClass('fontBold');
                  $("#searchFundedResidencies").removeClass('fontBold');
                $("#searchAll").removeClass('fontBold');
                  search(1);
                break;
               case 'Competitions':
```

```
                    theCustomSearchEngineId = '003134293120551003557:mmi7xgbyli8';
                    $("#searchCompetitions").addClass('fontBold');
                    $("#searchFundings").removeClass('fontBold');
                    $("#searchBursaies").removeClass('fontBold');
                      $("#searchCommissions").removeClass('fontBold');
                      $("#searchJobs").removeClass('fontBold');
                    $("#searchScholarships").removeClass('fontBold');
                    $("#searchOrganisations").removeClass('fontBold');
                      $("#searchFundedResidencies").removeClass('fontBold');
                    $("#searchAll").removeClass('fontBold');
                      search(1);
                    break;
                   case 'Scholarships':
                    theCustomSearchEngineId = '003134293120551003557:verufrjiugs';
                    $("#searchScholarships").addClass('fontBold');
                    $("#searchFundings").removeClass('fontBold');
                    $("#searchBursaies").removeClass('fontBold');
                      $("#searchCommissions").removeClass('fontBold');
                      $("#searchJobs").removeClass('fontBold');
                    $("#searchCompetitions").removeClass('fontBold');
                    $("#searchOrganisations").removeClass('fontBold');
                      $("#searchFundedResidencies").removeClass('fontBold');
                    $("#searchAll").removeClass('fontBold');
                      search(1);
                    break;
                   case 'Organisations':
                    theCustomSearchEngineId = '003134293120551003557:g-bp_c5wvek';
                    $("#searchOrganisations").addClass('fontBold');
                    $("#searchFundings").removeClass('fontBold');
                    $("#searchBursaies").removeClass('fontBold');
                      $("#searchCommissions").removeClass('fontBold');
                      $("#searchJobs").removeClass('fontBold');
                    $("#searchCompetitions").removeClass('fontBold');
                    $("#searchScholarships").removeClass('fontBold');
                      $("#searchFundedResidencies").removeClass('fontBold');
                    $("#searchAll").removeClass('fontBold');
```

```
                search(1);
                break;
            case 'FundedResidencies':
                theCustomSearchEngineId = '003134293120551003557:_5sew958_o0';
                $("#searchFundedResidencies").addClass('fontBold');
                $("#searchFundings").removeClass('fontBold');
                $("#searchBursaies").removeClass('fontBold');
                $("#searchCommissions").removeClass('fontBold');
                $("#searchJobs").removeClass('fontBold');
                $("#searchCompetitions").removeClass('fontBold');
                $("#searchScholarships").removeClass('fontBold');
                $("#searchOrganisations").removeClass('fontBold');
                $("#searchAll").removeClass('fontBold');
                search(1);
                break;
            default:
                theCustomSearchEngineId = '003134293120551003557:2-7xccxtijg';
                $("#searchAll").addClass('fontBold');
                $("#searchFundings").removeClass('fontBold');
                $("#searchCommissions").removeClass('fontBold');
                $("#searchBursaies").removeClass('fontBold');
                $("#searchJobs").removeClass('fontBold');
                $("#searchCompetitions").removeClass('fontBold');
                $("#searchScholarships").removeClass('fontBold');
                $("#searchOrganisations").removeClass('fontBold');
                $("#searchFundedResidencies").removeClass('fontBold');
                search(1);
                break;
        }
    };
    function setSortOrder(condition) {
        switch (condition) {
            case 'relevance':
                sortCondition = '';
                $("#sortRelevance").css("font-weight", "bold");
                $("#sortDate").css("font-weight", "normal");
```

```
                $("#sortPopularity").css("font-weight", "normal");
                break;
            case 'date':
                sortCondition = '&sort=date:d';
                $("#sortRelevance").css("font-weight", "normal");
                $("#sortDate").css("font-weight", "bold");
                $("#sortPopularity").css("font-weight", "normal");
                break;
            case 'popularity':
                sortCondition = '&sort=review-rating:d:s';
                $("#sortPopularity").css("font-weight", "normal");
                $("#sortDate").css("font-weight", "normal");
                $("#sortPopularity").css("font-weight", "bold");
                break;
            default:
                sortCondition = '';
                $("#sortPopularity").css("font-weight", "normal");
                $("#sortDate").css("font-weight", "normal");
                $("#sortPopularity").css("font-weight", "normal");
                break;
        }
        if ($("#search-box").val()) {
            search(1);
        }
    }
    function search(startIndex) {
        var theQueryString = $("#search-box").val();
        if(!theQueryString) return;
        var prepared = MetaEngine.sort.prepareQuery(theQueryString);
        var theRestUrl = getUrl(prepared, startIndex);

        $.get(theRestUrl, function (data) {
            var theItems = data["items"];
            if (theItems) {
                var keywords = ['ee^1', 'e^1'];
                MetaEngine.sort.sort(theItems, keywords);
                $("#searchResult").empty();
```

```
var thetotalCount = theItems.length;
var theRowCount = 1;
var theCount = 1;
var theItem;
var theHtml = '<div class="col-m-10 results owl-carousel" id="all">';
theHtml += ' <div data-dot="' + theRowCount + '" class="wrap">';
for (var i = 0; i < thetotalCount; i++) {
    theItem = theItems[i];
    theHtml += createItemHtml(theItem["title"], theItem["snippet"], theItem["link"], theCount);
    theCount++;
};

theHtml += ' </div>' +
    '</div>';
var theQueries = data["queries"];
var theRequest = theQueries["request"][0];
var theTotalRequests = theRequest['totalResults'];
if (theTotalRequests > 8) {
    var theStartIndexInRequest = theRequest["startIndex"];
    var thePageNumber = Math.round(theTotalRequests / 8);
    if (thePageNumber > 10) {
        thePageNumber = 10;
    }
    var thePageHtml = '';
    for (i = 0; i < thePageNumber; i++) {
        var theActiveValue = '';
        if (i == Math.round(theStartIndexInRequest / 8)) {
            theActiveValue = ' active';
        }
        var theIndex = i * 8 + 1;
        var theDisplayedIndex = i + 1;
        thePageHtml += '<div onclick="search(' + theIndex + ');" class="owl-dot' + theActiveValue + '" id="page-' + theDisplayedIndex + '">' + theDisplayedIndex + '</div>';
    }
    $("#pages").html(thePageHtml);
```

```
                    }
                    theHtml += '';
                    $("#searchResult").html(theHtml);
                }
                else {
                    $("#searchResult").empty();
                    $("#pages").empty();
                }
            });
        }
        function findExictingLink(citations, link) {
            for(var i = 0; i < citations.length; i++) {
                if (citations[i].link.trim() === link.trim()) {
                    return i;
                }
            }
            return -1;
        }

        function recordCititation(link){
            if (typeof(Storage) !== "undefined") {
                var citationArray = getCitations();
                currentDateTime = new Date();
                currentDateTimeString = currentDateTime.toLocaleString('en-GB');
                //check if the link has been cited
                var locator = findExictingLink(citationArray, link);
                if (locator != -1) {
                    citationArray[locator].accessedDateTime = currentDateTimeString;
                } else {
                    var citation = {
                        link: link,
                        accessedDateTime: currentDateTimeString
                        };
                    citationArray.push(citation);
                }
                localStorage.citations = JSON.stringify(citationArray);
```

```
                $("#citationNumber").text(citationArray.length);
            } else {
                alert('Sorry! No web storage support.');
            }
        }

        function showCitations() {
            var citationArray = getCitations();

            var rows = '';

            for(var i = 0; i < citationArray.length; i++){
                rows += '<tr>' +
                '<td>' + citationArray[i].link + '</td>' +
                '<td>' + citationArray[i].accessedDateTime + '</td/>';
            }
            //$("#citationBody").html(rows);
            $('#citations tbody').html(rows);
        }
         function createItemHtml(title, snippet, link, count) {
            var fromTarget=false;
            var colors=['#BEB2A6','#6D5E51',' #C79316','#1B232A'];
            var div = '<div class="result">';
            var img = '';
            if(link.indexOf('bathspa.ac.uk') > 0)
            {
                fromTarget=true;
                color=colors[colorCount];
                div = '<div class="result" style="background-color: ' + color + '; color:
white;">';
                img = '<img class="logo" src="img/bsu-logo.png" />';
                colorCount++;
                if(colorCount>2)
                    colorCount=0;
            }
             var theHtml = div +
                 '      <div class="result-block" id="result-' + count + '">' +
```

```
                        '       <div class="title" >' +
                        '            <h3>' + title + '</h3>' +
                        '            <p>' + snippet + '</p>' +
                        '            <span class="number">' + count + '</span>' +
                        '       </div>' + img +
                        '       <a onclick="recordCititation(\'' + link + '\')" class="more view" target="_blank" href="' + link + '" id="view-1">Read More</a>' +
                        '    </div>' +
                        '</div>';
                    return theHtml;
                };
                function createTableItem(title, snippet, author, link, count) {
                    if (count<= 8)
                    {
                    var theTable= '<tr>'+
                                ' <td>'+count+'</td>'+
                                  '<td>' + title + '</td>'+
                                ' <td>' + snippet + '</td>'+
                                  '<td><a class="more view" target="_blank" href="'+link+'" id="view-1">' + link + '</a></td>'+
                                  '<td><a href="#">' + author + '</a>'+
                                '</td>'+
                            '</tr>';
                    return theTable;
                    }
                };
            </script>
        </body>
</html>
```

Appendix 3: Extended Boolean Model based Search Result Ranking

The extended Boolean model as the basis for literature ranking method is illustrated here.

a. Calculation of the similarity between the literature and query

In the extended Boolean model, consider two words and two-dimensional graph to represent the query and literature (as is shown in Figure 4). w_{xj} and w_{yj} represent the weight of domain-specific keywords and literature $d_j = (w_{xj}, w_{yj})$.

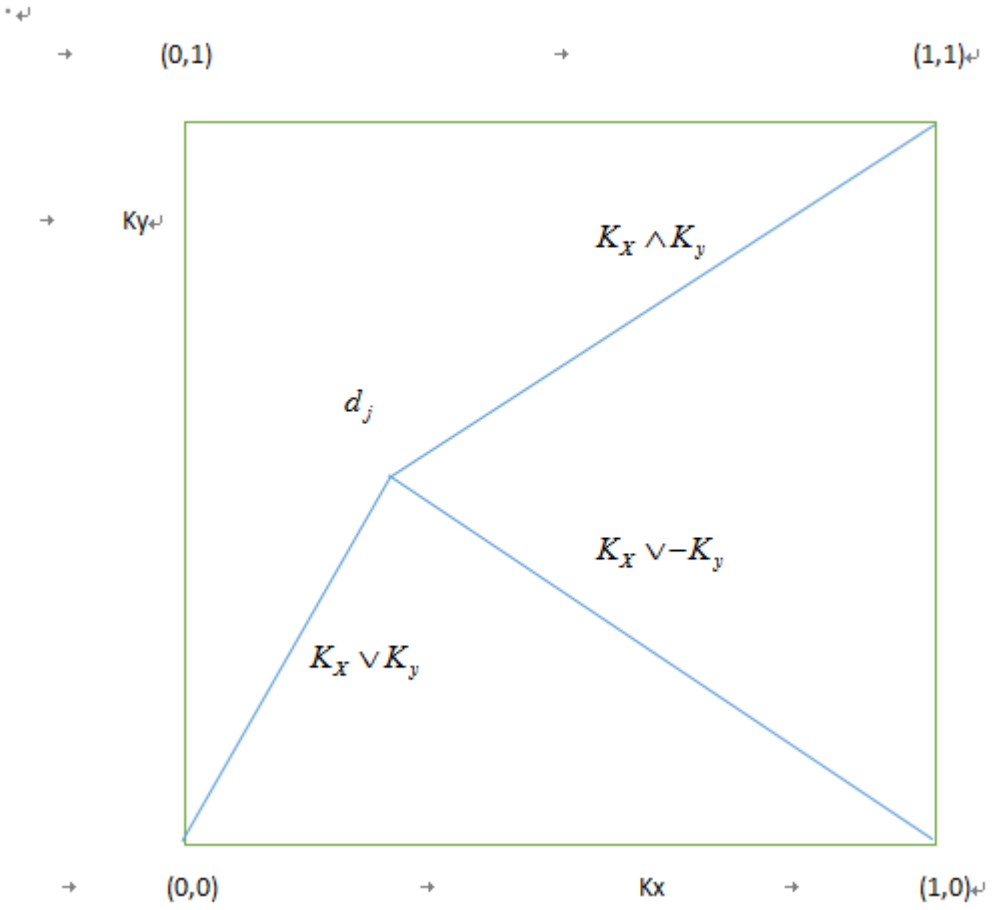


Figure 20. Relation between Literature and Query

As for the conjunction query $q_{and} = k_x \wedge k_y$,(1,1) point is the ideal point and the distance to (1,1) can used to measure the similarity between the literature and query q_{and}:

$$sim(q_{and}, d) = 1 - \sqrt{\frac{(1-w_{xj})^2 + (1-w_{yj})^2}{2}}$$

As for the disjunction query $q_{or} = k_x \vee k_y$, (0,0) is the null point and the distance to (0,0) can be used to measure the similarity between the literature and query q_{or}:

$$sim(q_{or}, d) = \sqrt{\frac{w_{xj}^2 + w_{yj}^2}{2}}$$

In the standard extended Boolean model, each querying word is required to be contained in the literature (such as $k_x \wedge k_y$ containing k_x and k_y). However, in the domain querying expression in this study, there is “no containing” form of domain-specific keyword. For example, the domain query expression $k_x \wedge \overline{k_y}$ contains k_x but does not contain k_y. In Figure 4, the (1,0) is the ideal point and the similarity between the literature and the query is defined as:

$$sim(q_{and}, d) = 1 - \sqrt{\frac{(1-w_{xj})^2 + w_{yj}^2}{2}}$$

b. Ranking Function

Supposed domain querying expression:

$$q_s = q_{cc1} \vee q_{cc2} \vee \ldots \vee q_{ccm} \quad (6)$$

Each conjunction weight can be described as:

$$q_{cc} = k_1 \wedge \ldots \wedge k_t \wedge \overline{k_{t+1}} \wedge \ldots \wedge \overline{k_n} \quad (7)$$

The similarity between each conjunction weight and the literature:

$$sim(q_{cc}, d_j) = 1 - \sqrt{\frac{\sum_{x=1}^{i}(1 - w_{xj})^2 + \sum_{y=t+1}^{n} w_{yj}^2}{n}} \quad (8)$$

The similarity between the domain querying expression and literature (the similarity between the literature and query q):

$$S_j = sim(q, d_j) = sim(q_s, d_j) = \sqrt{\frac{\sum_{x=cc1}^{ccm} sim(q_x, d_j)^2}{m}} \quad (9)$$

Considering the query-relevant domain-specific keyword weight, the formula (8) will be changed into:

$$sim(q_{cc}, d_j) = 1 - \sqrt{\frac{\sum_{x=1}^{t} w_{xq}^2 (1 - w_{xj})^2 + \sum_{y=t+1}^{n} w_{yq}^2 w_{yj}^2}{\sum_{x=1}^{t} w_{lq}^2 + \sum_{y=t+1}^{n} w_{(t+1)q}^2}} \quad (10)$$

Each conjunction weight is equal to each other so the formula (9) will not change.

Appendix 4: Publications by Candidate

[the contributions to published conference proceedings have been redacted from this digitised version due to potential copyright issues]