DESIGN OF A NOVEL SEARCH ENGINE FOR PROSPECTIVE QUESTION ANSWERING

THESIS

Submitted in fulfillment of the requirement for the award of degree of

DOCTOR OF PHILOSOPHY

to

YMCA UNIVERSITY OF SCIENCE & TECHNOLOGY

by

ROSY MADAAN

(RegistrationNo.YMCAUST/Ph06/2010**)**

Under the Supervision of

Prof. A.K.Sharma
Professor (CSE) & Dean (PG & Research),
B.S.A.I.T.M., Alampur, Faridabad

Dr. Ashutosh Dixit
Associate Professor, Department of Computer Engineering,
YMCA University of Science & Technology, Faridabad

Department of Computer Engineering
Faculty of Engineering &Technology
YMCA University of Science & Technology
Sector-6, Mathura Road, Faridabad, Haryana, India

APRIL 2016

Dedicated

to

My family

CANDIDATE'S DECLARATION

I hereby declare that this thesis entitled **"DESIGN OF A NOVEL SEARCH ENGINE FOR PROSPECTIVE QUESTION ANSWERING"** by **ROSY MADAAN,** being submitted in the fulfillment of the requirements for the degree of the Doctor of Philosophy in COMPUTER ENGINEERING under the Faculty of Engineering & Technology, YMCA University of Science & Technology Faridabad, during the academic year 2016, is a bonafide record of my original work carried out under guidance and supervision of **PROF. A. K. SHARMA, PROFESSOR (CSE) & DEAN (PG & RESEARCH), B.S.A.I.T.M., ALAMPUR, FARIDABAD & DR. ASHUTOSH DIXIT, ASSOCIATE PROFESSOR, YMCA UNIVERSITY OF SCIENCE & TECHNOLOGY, FARIDABAD** and has not been presented elsewhere.

I further declare that the thesis does not contain any part of any work which has been submitted for the award of any degree either in this university or in any other university.

(ROSY MADAAN)

Registration No.- YMCAUST/Ph06/2010

CERTIFICATE

This is to certify that this thesis entitled **"DESIGN OF A NOVEL SEARCH ENGINE FOR PROSPECTIVE QUESTION ANSWERING"** by **ROSY MADAAN,** submitted in fulfillment of the requirement for the Degree of Doctor of Philosophy in **COMPUTER ENGINEERING** under Faculty of Engineering & Technology of YMCA University of Science & Technology, Faridabad, during the academic year 2016, is a bonafide record of work carried out under our guidance and supervision.

We further declare that to the best of our knowledge, the thesis does not contain any part of any work which has been submitted for the award of any degree either in this university or in any other university.

Prof. A.K. Sharma
Professor (CSE) & Dean (PG & Research),
B.S.A.I.T.M., Alampur, Faridabad

Dr. Ashutosh Dixit
Associate Professor
Department of Computer Engineering
Faculty of Engineering and Technology
YMCA University of Science & Technology Faridabad,

Dated:

ACKNOWLEDGEMENTS

Thank you God

I would like to express thanks to my supervisor **Professor (Dr.) A.K. Sharma**, who has been a tremendous mentor for me. I would like to thank him for encouraging my research and for allowing me to grow. His advice on both research as well as on my career has been invaluable.

I am also very grateful to my co-supervisor **Dr. Ashutosh Dixit**, for his technical advice and knowledge and many insightful discussions and suggestions.

I thank my husband **Dr. Komal Kumar Bhatia** from core of my heart, for his constant support, brilliant comments and suggestions, thanks to him. He has been always there for me throughout.

I would like to thank Dr. D.K. Jha, Dr. Manoj Wadhwa, Ms. Surbhi Bhatia, Ms. Sonal Singhal, Ms. Riddle Batra, Dr. Rashima Mahajan, Ms. Manisha Saini, Mr. Kushagra Agrawal, Ms. Ila Mehta and Ms. Sonal Kukreja, for the valuable discussions done with them.

A special thanks to my family. Words cannot express how grateful I am to my parents, mother-in-law, father-in-law and my sister for all of the sacrifices that they've made on my behalf. Your prayer for me was what sustained me thus far.

Thank you all for supporting me for everything, and especially I can't thank you enough for encouraging me throughout this experience. To my beloved daughter Havi, I would like to express my thanks for being such a good girl always cheering me up.

Finally I thank God, for letting me through all the difficulties. I have experienced His guidance day by day. God is the one who let me finish my degree. Thank you, God.

CERTIFICATE

This is to certify that this Rosy Madaan (YMCAUST/Ph06/2010) has incorporated all the suggestions as well as corrections in her Ph.D. thesis as suggested by us and external examiners. It is further certified that the contents of the hardcopy and softcopy supplied to the academic section are same.

Prof. (Dr.) A.K. Sharma
Supervisor

Dr. Ashutosh Dixit
Supervisor

Dr. Naresh Chauhan
Professor & Chairman
Faculty of Engineering and Technology
YMCA University of Science & Technology Faridabad

ABSTRACT

The *World Wide Web (WWW)*, the largest and most frequently accessed public repository of information ever developed, contains large number of web pages interconnected through hyperlinks. Search Engine is an information retrieval tool that provides a search interface to the people searching for some information, where he/she can submit his/her information need by writing queries. In response, the search engine returns relevant web pages in a ranked order by searching in repository maintained using the criterion of keyword matching. The pages with higher rank appear at the top and those at the bottom of the list are lower in rank.

If the user enters a question on the interface of the search engine, the search engine performs the same keyword matching approach and provides the list of web documents in ranked form. To find the answer to the question, the user has to go through the entire document. So, need of a system is felt which is capable of taking user's question as its input and return precise answer(s) as the output. Such a system is termed as a *Question answering system.* There are some major issues of concern that are found in the field of question answering as given below:

(i).Most of the existing systems for question answering takes the user's question as input and forwards it to the search engines like Google for the search results. The search results are the list of web pages returned to the user and the user has to go through the web page to find the answer.

(ii).There are some question answering systems that puts the question submitted by the user into the discussion forum and waits for the members of the forum to respond. The answers given by the members are then presented to the user in response to his question at the interface or via email.

(iii).It has been found that none of the previous work in this field typically focuses on all aspects associated with question answering i.e. crawling, summarization, question analysis, document representation, answer extraction or ranking.

So, it has been analyzed that there is a need to design and develop a system that is user interactive and combines all the modules together for efficient question answering.

This dissertation comprises, design and development of a novel search engine for prospective question answering (PQAS) that combines all the major modules associated with question answering i.e. crawling, relevant content extraction and indexing. The work has also been done in other associated directions such as blog post ranking and predicting user's next question. The system is capable of taking user's question as its input and tries to provide precise answer(s) in return. It is also able to address the major issues of concern in the field of question answering.

It is further observed that blogs contain topical information i.e. information that is very much related to the topic on which the blog has been written, thereby may play a crucial role in providing answers corresponding to user's questions on that topic. Therefore, blogosphere has been chosen as the source of information. For downloading and extracting the blog posts from the blog pages, a novel architecture for Blog crawler has been developed. Two techniques to extract relevant content from the blog post repository have also been developed so that the irrelevant content may be filtered. The extracted relevant content is then indexed. For efficient question answering, a novel technique has been proposed for building an index on the basis on question classification.

To improve the quality of blog repository, mechanism has been proposed for scoring the blog posts on the basis of their popularity features. For improving the system's response time, a technique for predicting user's next question has been developed, so that the answers may be supplied to the user promptly.

The interface of PQAS is designed to facilitate the users to enter their question and receives answer(s) in response. The user is then asked to provide his/her feedback in terms of satisfactory or dissatisfactory response. This helps the system to improve the quality of answers in future.

The classification of the proposed work is done in such a way that a modular architecture

is developed with the expectation that new functionalities can easily be added by third parties according to their requirements. Various tests have been conducted on the system and its performance has been evaluated. High values of *Answer accuracy* for various tests conducted on the system indicate that it accurately answers the question of the user.

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

WWW:	World Wide Web
IR:	Information Retrieval
QA:	Question Answering
UC:	Unix Consultant
tf:	Term frequency
RSS:	Really Simple Syndication
Sid:	Sentence id
Pid:	Page id
P:	Precision
R:	Recall
F:	F-measure
PF:	Presence factor
PFSS:	Presence factor based sentence score generator
SS:	Sentence score
TSM:	Term-sentence matrix
PFM:	Presence-factor matrix
CSM:	Comments-sentence matrix
SRS:	Sentence relevance score
ARS:	Answer relevance score
QC:	Question class
PQAS:	Prospective question answering system
Acc:	Answer accuracy

TABLE OF CONTENTS

Chapter I

INTRODUCTION

1.1 GENERAL

World Wide Web (WWW) [1,2,3] a huge source of hyperlinked documents containing useful information for millions of users, entered the information retrieval [1,5,9,20] world in 1989. This event caused the evolution of a branch of information retrieval that is different from traditional IR in the sense that it searches the required information within new document collection.

Information retrieval (IR) is a field of study dealing with the representation, storage, organization of, and access to documents. The documents may be books, reports, pictures, videos, web pages or multimedia files. The whole point of an IR system is to provide a user easy access to documents (usually in unstructured form) containing the desired information. The best known example of an IR system is Google search engine. This is in contrast to Data Retrieval [4] that deals with structured data with well-defined semantics. Information Retrieval can be precisely defined as:

"... the IR system must somehow 'interpret' the contents of the information items (documents) in a collection and rank them according to a degree of relevance to the user query. This 'interpretation' of document content involves extracting syntactic and semantic information from the document text..." [4]

IR search engines [2] available on the Web, allow a user to submit queries and retrieve links of relevant Web pages. Virtually all IR engines work by downloading and indexing the Web document collection(s) to which retrieval is to be applied. With a relatively small collection, it may be possible to generate the index terms for a given document dynamically when the collection is being searched. On the other hand, for a huge set of collections such as the information sources available through the Web, the search engines generate indexes in advance and due to dynamic nature of the Web, the indexes are constantly updated.

1.2 WEB SEARCHING

The typical design of a search engine [2,3,4,25,26] mainly comprises of three activities: Web crawling, Indexing and Searching as shown in Fig. 1.1. It may be noted that the Web crawler is the first module that downloads Web documents to be indexed by the indexer for later use by searcher module.

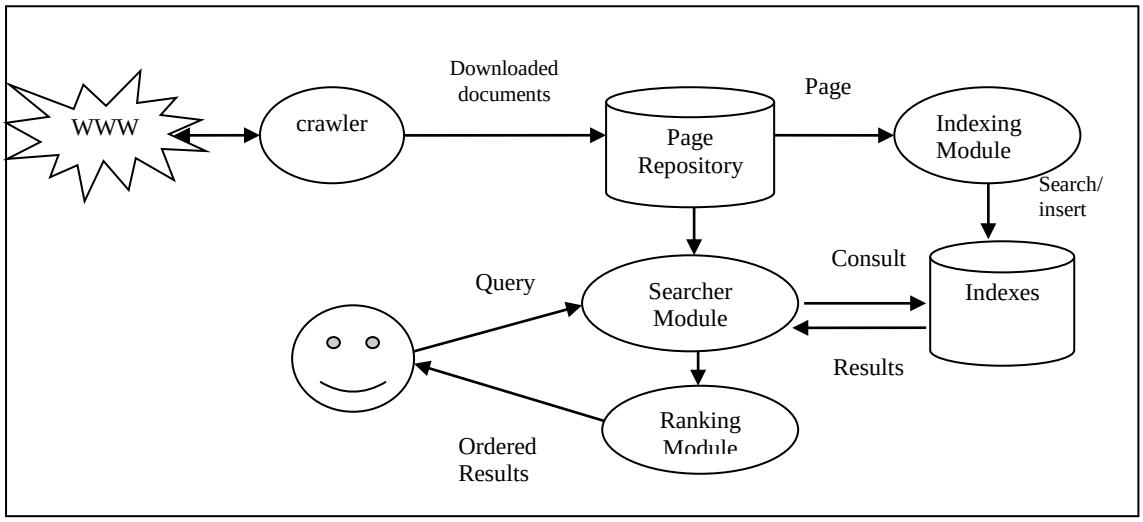


Fig. 1.1 Elements of a Search Engine

The search engine provides an interface to the user that enables him to specify what he needs to search i.e. to specify the criteria about an item of interest. Then a search is performed. The criteria specified in the search interface are referred to as a *search query* [4]. In the case of text search engines, the search query is typically expressed as a set of words separated by white spaces. A search is then performed in the locally maintained databases for the required information contained in web documents. Relevant documents are then identified and returned to the searcher in ranked order.

1.3 QUESTION ANSWERING SYSTEMS

Question Answering (QA) [45,46] is a discipline of computer science within the fields of information retrieval. It involves building a system that allows user to ask a question in natural language and get answer(s) in return. Question answering system consists of three distinct modules [45,46]: question processing, document processing, and answer processing.

First task is question processing which deals with question representation, identification of question type, derivation of expected answer type and the keywords

extraction. The relevant documents are then retrieved and the passages are identified using the keyword matching approach. Then the answer processing module then identifies the candidate answer(s) from those passages and ranks them. Answer formulation is also a task performed on the identified answers but is skipped in almost all the cases and the answer(s) are presented as it was found in the document.

1.4 MOTIVATION

The motivation behind the work is given in this section.

- If the user enters a question on the search engine interface, the search engine converts the question into the query and returns the whole document/page having terms matching to those in the query. The user has to go through the entire web page to find the answer(s) himself, which is a tedious task. So, there is a need to design a system that takes user's question as input and provide corresponding answer(s) in return.
- It has also been found that some question answering systems returns passages as their output instead of a precise answer.

The major challenges faced in the field of Question answering are given in the section below.

1.5 CHALLENGES IN DESIGNING QUESTION ANSWERING SYSTEMS

A user interested in a few words answer to his/her question would never like to waste his/her time and spend efforts in going through large collections of text. The major challenges associated with Question answering are listed as follows:

i. **To design a search engine for question answering:** When an user provides a question to the QA system, the QA system at first converts it into a query (set of keywords) and produces a list of documents relevant to the user's request. The list thus produced consists of text providing information about the topic of interest. But, the user is interested in only a few lines answer to the question, the larger length documents seem to be irrelevant to the user. So, for the purpose of finding an answer, the user needs to go through the entire document which wastes time and requires a lot of effort. Hence, there is a need

of a search engine that takes user's question as its input and provides answer(s) satisfying user's request.

Solution: *A novel search engine has been designed and developed that combines various modules of search engine eg. crawling, indexing and ranking for efficient question answering. The search engine is capable of responding to user's question with precise answer(s).*

ii. **To have topical information:** The web is a huge repository of information but is not likely to contain the information very much related to the title of the web page. For this purpose, the system must be able to collect information from some other sources that are likely to contain the information related to the topic of interest. Also, there is a need to extract only a relevant portion of the text from the pages collected from the quality sources.

Solution: *To have such information, blogosphere has been chosen as the source of information for question answering in which the blog page contains content very much related to the topic on which the blog is written. Architecture for Blog crawler has also been developed that downloads the blog pages and extract the blog posts existing in them.*

iii. **Prospective Question answering:** There is a need to design a prospective Question Answering system. For this, the system must be able to predict the user's next question which improves the system's response time.

Solution: *A technique has been proposed and implemented, to predict user's next question in Question answering system, thus improving system's response time and provides answer(s) instantly to the user.*

Solution: *A Question classification based indexing scheme for indexing the relevant content extracted from blog pages has been designed and implemented.*

iv. **To extract relevant content:** The entire content of the information source may not be relevant for the user's question, so there is a need to filter out the irrelevant content.

Solution: *Two techniques for extracting relevant content from the blog posts have been developed.*

v. **To perform efficient indexing:** There is a need to propose a scheme for efficiently indexing the relevant content existing in the repository so that search can be performed in the index thus resulting in efficient question answering.

vi. **To maintain quality of repository:** There is a need to improve the quality of the repository containing information so that the quality data can be used for question answering.

Solution: *To maintain the quality of the repository, some blog pages are filtered out and are not used for question answering. For this, a mechanism is proposed for assigning a score to each blog post based on their popularity features.*

vii. **To have accurate results:** The result returned by the question answering system is a list of answers. These answers need to be accurate. Some metric(s) needs to be introduced for measuring the accuracy of the result.

Solution: *Some performance metrics have been proposed to measure the accuracy of the question answering system.*

viii. **To update the content repository with the new information:** There is a need to gradually update the system's repository with the fresh information.

Solution: *In the proposed system, the repository is enriched with the satisfactory content using some alternate data sources.*

ix. **Answer formulation**: The result of a QA system should be presented in a way as natural as possible.

Solution: *At the PQAS interface, the answer(s) are presented to the user as found in its source page(s).*

x. **Interactive system:** There is a need of an interactive question answering system. Also, feedback of the user is important w.r.t the answers returned by the system.

Solution: *At the PQAS interface, the user is allowed to provide his feedback in terms of satisfactory or dissatisfactory response.*

1.6 ORGANIZATION OF THESIS

This research process has been linearized to present it in terms of chapters. The following is an outline of the contents of this thesis. The first chapter explores theoretical aspects of WWW, information retrieval, search engine and question answering systems and also presents related challenges.

- Chapter 1 explores theoretical aspects of Question Answering and the challenges involved.
- Chapter 2 reviews selected publications related to the Question Answering. This chapter contains the general architecture of a search engine, Question Answering system and also provides in-depth information about the Blogosphere. The strengths and weaknesses of the most commonly used Question Answering systems are also given.
- Chapter 3 proposes design of a novel search engine for prospective question answering. The architecture comprises of six functional modules, all of which along with their algorithms have been discussed in brief and their details are given in the subsequent chapters.
- Chapter 4 proposes the novel architecture of Blog crawler which crawls the blog pages on the Web. The architecture comprises of four functional modules, all of which along with their algorithms has been discussed in detail. The performance evaluation has also been given.
- Chapter 5 proposes two techniques for extracting relevant content from the blog pages. The details of modules involved and the algorithms used are also given. The snapshots of implementation have been given along with the result analysis.

- Chapter 6 proposes an indexing scheme for efficient question answering. Its two main components are discussed in detail along with the flowcharts and algorithms used. The result analysis has also been given.
- Chapter 7 proposes a mechanism to improve the quality of blog repository. The formula devised to compute the blog score has been given and explained. The snapshots of implementation have also been given.
- Chapter 8 proposes a system for improvement in system's response time. Its architecture is given and each of its components are discussed in detail. The implementation of the system is discussed.
- Chapter 9 provides implementation and results of the proposed system for question answering.
- Finally, Chapter 10 summarizes our contributions and provides guidelines for future work in this area.
- The thesis includes Appendix-1 to 9 containing the content as specified below:

Appendix 1: lists the questions asked in a survey for knowing the requirements of a user from a general search engine.

Appendix 2: lists the blog sites having RSS feed.

Appendix 3: shows the snapshots of implementation of blog crawler.

Appendix 4: lists some blog posts along with the relevant content extracted from them.

Appendix 5: lists the questions asked in a survey for indexing along with a table showing the number of participants agreeing with the options given.

Appendix 6: shows the snapshots of searching in the index for question answering.

Appendix 7: shows the snapshots of some sample blog posts and their popularity features.

Appendix 8: shows the snapshot of implementation of the system for next question prediction.

Appendix 9: shows the snapshots of the answers given by PQAS for some sample questions belonging to each question class.

Finally, the bibliography includes references to publications in this area.

A literature review of the related topics is given in next chapter i.e. chapter-2.

Chapter II

INFORMATION RETRIEVAL & QUESTION ANSWERING SYSTEMS: A REVIEW

2.1 INFORMATION RETRIEVAL

World Wide Web (*WWW* or *Web*) [1,2,3] is a largest collection of hyperlinked documents spread over the internet. The strategies for information retrieval [1,5,9,20] have been changed recently due to the increase in the size of publically indexable information [7,8,15,24] available over the WWW. Therefore, the field of information retrieval covers a broad spectrum of techniques and applications that aim to satisfy the user's information needs. An ideal information retrieval system must be able to

- determine the information [75] needs of a user,
- search the information available,
- return the relevant information that is generally compiled from multiple sources, in a language and format that can be easily understood by the users.

In IR, the information need of the user is expressed as a bag of keywords [1,5,20]. The results are returned in the form of a list of documents that contain one or more of those keywords.

Web Search Engine [2,3,4,25,26] is basically an information retrieval tool that provides search interface to the information seekers so as to allow the users to submit their information need in the form of queries and return relevant web documents from a large repository consisting of the terms in the queries. The documents are retuned in such an order that the documents appearing at the top are highly ranked and those at the bottom of the list are lower in rank. Thus, the user clicks and sifts through the documents at the beginning of the list. To decide the importance of a document, the search engine uses its

ranking mechanism. So relevance of the returned documents is generally decided by ranking mechanism that orders the set of retrieved documents.

2.2 SEARCH ENGINES

A search engine [2,3,4,25,26] is an information retrieval system designed to find information over the Web consisting of hyperlinked documents. The search engine provides an interface to the user that enables him to specify what he needs to search. i.e. to specify the criteria about an item of interest. Then a search is performed in the locally maintained databases. The criteria specified in the search interface are referred to as a *search query*. In the case of text search engines, the search query is typically expressed as a set of words separated by white spaces. The search is then performed for the required information contained in text documents, pictures files, sounds files etc.

The above discussed search model was developed in the 1960s [109] and have taken decades to grow in form of a new search model. In fact, as of June 2000, there were at least 3,500 different search engines (including the newer search engines) [109]. The components of a general web search engine are discussed in next section.

2.2.1 COMPONENTS OF A WEB SEARCH ENGINE

The various components of a general Web search engine [2,3,4,25,26] are discussed in detail as follows:

- **Crawler Module:** As compared to traditional document collections which reside in physical warehouses such as the college's library, the information available on WWW is distributed over the Internet. Because of the tremendous growth of the information available on the web, a component called crawler [4,6,25,26] is employed by the search engine which visits the Web, downloads the web pages and categorize them. Crawlers [16,17,18,19] may be defined formally as *"Software programs that traverse the World Wide Web information space by following the hypertext links extracted from*

hypertext documents". The crawler traverses the web [22] and downloads the web pages. From the downloaded web pages, the crawler extracts the hyperlinks which are queued and traversed later on by the crawler. The downloaded Web pages are temporarily stored in a local storage of search engine, called page repository.

- **Page Repository:** It provides storage to the pages downloaded by the crawler and those new pages remain in the repository until they are sent to the indexing module, for the purpose of creating an index for information search [21].
- **Indexing Module.** The indexing module takes each new page from the page repository and indexes it. The index holds the valuable information for each web page. *Indexing* [4,25,26] is the process by which a vocabulary of keywords is assigned to documents in which they appear, thus creating an index and such index is generally termed as *inverted index* [4,25,26].
- **Query Module:** The query module takes a user's query as input and search in the various indexes in order to respond to the query. For example, the query module consults the inverted index to find which pages contain the query terms. The pages given as output are the relevant pages, which are then passed to the ranking module for the purpose of ranking [100].
- **Ranking Module.** The ranking module [4,25,26] takes the set of relevant pages as input and ranks them according to a given criterion. The generally used criterion are popularity score, content score etc. The output of this module is an ordered list of web pages such that the pages appearing on the top of the list are the pages with the highest rank. The ranking module is the most important component of the search process because the output of the query module often results in thousands of relevant pages that the user otherwise must sift through. There are two types of scores used for ranking of a page: the *content score* and the *popularity score*. These two scores are calculated for each web page and then these are combined for the overall score. The *popularity score* is determined from analysis of the Web's

hyperlink structure. Many web search engines give pages, using the query word in the title, as a higher content score as compared to the pages containing the query word in the body of the page. The set of relevant pages resulting from the query module are then presented to the user in order of their overall scores.

2.2.2 TYPE OF DATA RETRIEVED BY SEARCH ENGINE

The type of data retrieved by the Search engine [2,3,4,25,26] is listed as follows:

- **Large volume:** WWW contains huge collection of data. Also, the growth of data over the WWW is exponential. Increase in the amount, poses scaling issues that are difficult to cope with.
- **Distributed data:** Data is distributed widely over the WWW. It is located at different sites and platforms. The communication links between computers vary widely.
- **Unstructured and redundant data:** The data on the Web is highly unstructured [48]. It is impossible to organize and add consistency to the data and the hyperlinks. Also, there exists semantic redundancy that can increase traffic.
- **High percentage of volatile data:** The data on the Web is highly volatile. Documents can be added or removed easily in the World Wide Web. These changes to the documents are usually unnoticed by users.
- **Quality of data:** The data available on the Web is not of high quality. A lot of Web pages do not involve any editorial process. That means data can be false, inaccurate, outdated, or poorly written.
- **Heterogeneous data:** Data on the Web is heterogeneous. They are written in different formats, media types, and natural languages.
- **Dynamic data:** The content of Web document changes dynamically [10,11,12]. Some web pages are highly dynamic and some are less. The web pages that changes dynamically need to be updated [13,14], so that the user gets an updated page on visit.

Web is massive, much less coherent; it changes more rapidly, and is spread over geographically distributed computers. This requires new information retrieval techniques, or extensions to the old ones, to deal with the gathering of the information, to make index structures scalable and efficiently updateable, and to improve the discriminating ability of search engines.

Given a query, a set of keywords, the search engines [2,3,4,25,26] retrieve the information in the form of documents. In case, if the user needs an answer to a question [101,103], the user has to go through the entire document to extract the answer to his question, which is a time consuming process. There may occur a situation in which user is interested only a portion of the web page rather than whole document. For example "What is the height of Eiffel Tower? ". Search Engines fails in such cases. Thus, there is a need to build a system which is capable of taking user's question as input and return answer(s) as the result. Hence, need of Question answering systems [45, 46, 76, 79] arises. To validate this, a survey has been conducted in this work in which a questionnaire has been prepared and distributed to 60 persons, some of which are teachers, students, technical and non-teaching staff. As the result of the survey, responses have been collected and critically analyzed. The analysis of responses of survey conducted that for efficient retrieving of information, there is need of a Question Answering system.

Question answering [45,46] offers a more intuitive approach to information processing than search engines. Given a collection of documents and a natural language query posed by user in form of question, a question answering system attempts to find the precise answer or at least a portion of the text in which the answer appears. The next section discusses a general Question Answering System in detail.

2.3 QUESTION ANSWERING SYSTEMS:AN INTRODUCTION

Question Answering [45,46] is a discipline of computer science within the fields of information retrieval that involves building a system which allows user to ask a natural language question and get answer(s) in return. A Question Answering system can be implemented using a computer program which generates answers by querying an

unstructured collection of natural language documents. The goal of question answering (QA) is to provide a relevant answer to a question posed in natural language.

A standard QA system is made up of three modules: *question processing, document processing, and answer processing* [45,46]. The main contribution of the question processing module is to identify the question type (who, when, where,. . .), and the expected answer type which predicts the type of entity that question requires as an answer. Document processing is concerned with retrieving relevant documents and extraction of passages that contain answer(s) and their ordering, in a very similar way to an IR system, as described above, and is often implemented as such. Answer processing uses the information provided by the other modules to pinpoint the answer within a passage and is also concerned with their ranking.

The basic form of question answering takes a question such as "*Who is the President of India"* and returns an answer in the form of a name. These sorts of answers are known as named entities and might be, for example, the name of a person, a country or a type of animal, or else perhaps a date, a time or an amount. More complex questions are also possible. The other types of questions that a user may ask are like those starting with 'wh" like who, why, when, which, where. "How, how much and how many" type questions are also possible.

If the computer system continues to treat the question as a bag of words, nothing is gained in using a question. Thereafter, there is a need of natural language processing techniques, such as syntactic and semantic parsing and named entity extraction to analyze the question type and to retrieve an accurate answer.

The next section discusses a brief history of Question answering systems.

2.3.1 BRIEF HISTORY OF QUESTION ANSWERING SYSTEM

BASEBALL[45,47] and LUNAR[45,47] were two early Question Answering Systems (QA systems) that have been introduces in the literature. BASEBALL answered questions about the US baseball league over a period of one year. LUNAR in turn, answered questions about the geological analysis of rocks returned by the Apollo moon

missions. Both QA systems were very effective in their chosen domains. In fact, LUNAR was demonstrated at a lunar science convention in 1971 and it was able to answer 90% of the questions in its domain posed by people untrained on the system. Further restricted-domain QA systems were developed in the following years. The common feature of all these systems is that they had a core database or knowledge system that was hand-written by experts of the chosen domain. The language abilities of BASEBALL and LUNAR used techniques that is similar to ELIZA[45,47] and DOCTOR[45,47], the first chatterbot programs.

SHRDLU[45,47] was a highly successful question-answering program developed by Terry Winograd in the late 60s and early 70s. It simulated the operation of a robot in a toy world (the "blocks world"), and it offered the possibility to ask the robot questions about the state of the world. Again, the strength of this system was the choice of a very specific domain and a very simple world with rules of physics that were easy to encode in a computer program.

In 1970s, knowledge bases were developed that targeted narrower domains of knowledge. The QA systems developed to interface with these expert systems produced more repeatable and valid responses to questions within an area of knowledge. These expert systems closely resembled modern QA systems except in their internal architecture. Expert systems rely heavily on expert-constructed and organized knowledge bases, whereas many modern QA systems rely on statistical processing of a large, unstructured, natural language text corpus.

The comprehensive theories in computational linguistics were developed in 1970 and 1980, which led to the development of ambitious projects in question answering. One example of such a system was the Unix Consultant (UC), developed by Robert Wilensky at U.C. Berkeley [45,47] in the late 1980s. The system answered questions pertaining to the UNIX operating system. It had a comprehensive hand-crafted knowledge base of its domain, and it aimed at phrasing the answer to accommodate various types of users.

A text-understanding system that operated on the domain of tourism information in a German city was developed. The systems developed in the UC and LILOG projects [45,47] never went past the stage of simple demonstrations, but they helped the development of theories on computational linguistics and reasoning. Recently, specialized natural language QA systems have been developed, such as EAGLi [45,47] for health and life scientists.

2.3.2 THE ANATOMY OF QA SYSTEMS

When a user asks a question, the first task of utmost importance is to catch the inflection of the words and to understand the need of the user that is a difficult task to perform by a machine. However, several approaches on how to design such a system have been suggested and implemented in the literature related to design of such systems. In this section, the general architecture of a Question Answering System is presented, and a detailed description of each of its module is given as follows:

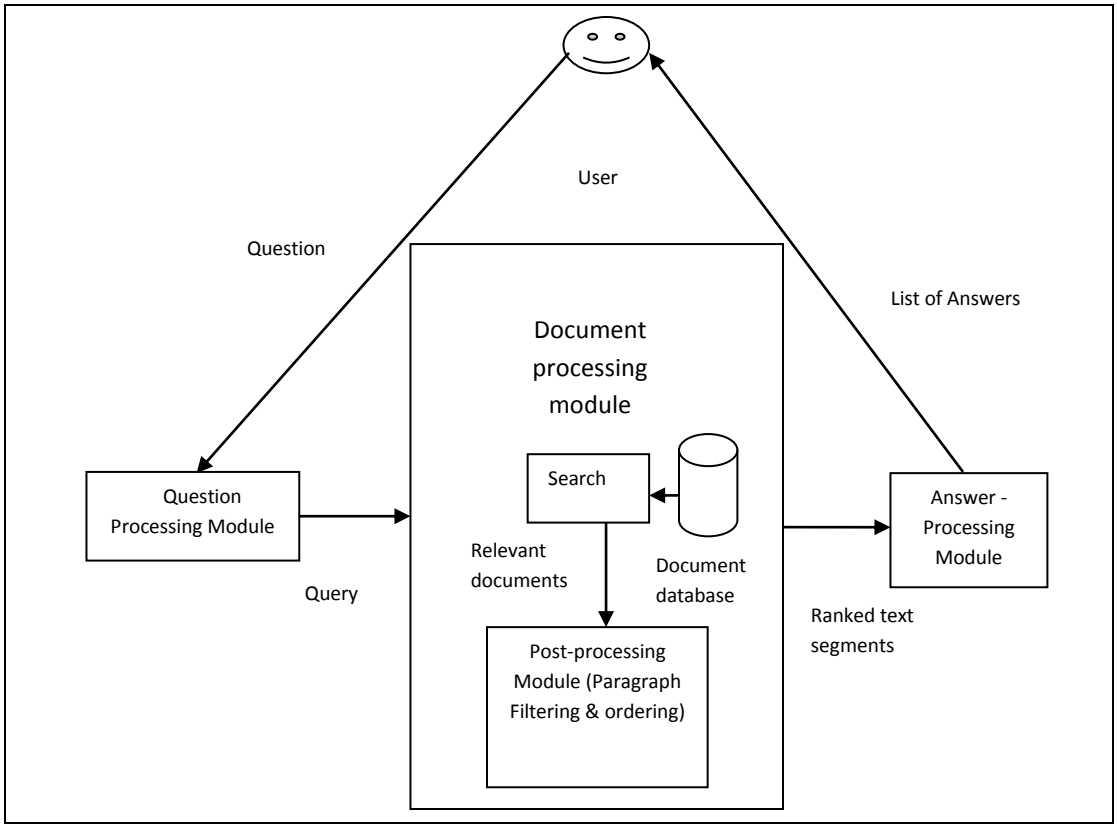


Fig. 2.1 Prototype of question answering system

Question answering system consists of three distinct modules: question processing, document processing, and answer processing. An illustration of this system architecture is given in Fig. 2.1.

- **Question processing:** Question processing deals with the question representation, identification of question type, finding expected answer type and the keywords extraction. The keywords that have been extracted are then used to fetch relevant documents. The question type is used to identify the expected answer type for finding a correct answer.

- **Document processing**: Document processing typically includes document retrieval, and passage identification. For this purpose, the approach of Keyword based matching coupled with the technique of keyword expansion is used that typically involves taking the keywords extracted in the question processing stage and looking them up in a thesaurus, or other resource, and adding similar search terms in order to fetch the relevant documents. A term such as "fight" might be expanded to "quarrel" for instance. Here, the expanded query is passed to a standard search engine (e.g. Google) and the documents with higher ranks are retrieved as a result. In passage retrieval, within each document the paragraph or section containing the possible answer is identified.

- **Answer processing**: This module consists of candidate answer identification, answer ranking, and answer formulation. Identifying the candidate answers means taking the results from the identified passages and further processing it. For this purpose, parsing of complete passage is committed. This results in a set of candidate answers that are then ranked according to a ranking algorithm or set of heuristics. Answer formulation is in most cases skipped completely and the answer is presented as it was found in the document. Answer Ranking is a major step in case if the answer extraction results in more than one answer. These answers are ranked based on relevance with those with higher rank appearing at the beginning of the list.

2.4 QUESTION ANSWERING SYSTEMS

A detailed discussion on the question answering systems is given in this section.

2.4.1 ANSWERBUS QA SYSTEM

AnswerBus [28] is an open-domain question answering system based on sentence level Web information retrieval. It accepts user's questions in six languages namely English, German, French, Spanish, Italian and Portuguese and provides answers in English. To respond to user's questions, five search engines and directories are used to retrieve relevant Web pages. The sentences that are determined to contain answers are then extracted from these Web pages. The working process of AnswerBus has been described in Fig. 2.2. To determine the language in which the user's question is posed, a language recognition module is used. If the language of the question is other than English, then the question is send to BabelFish [69], the translation tool of Alta Vista.

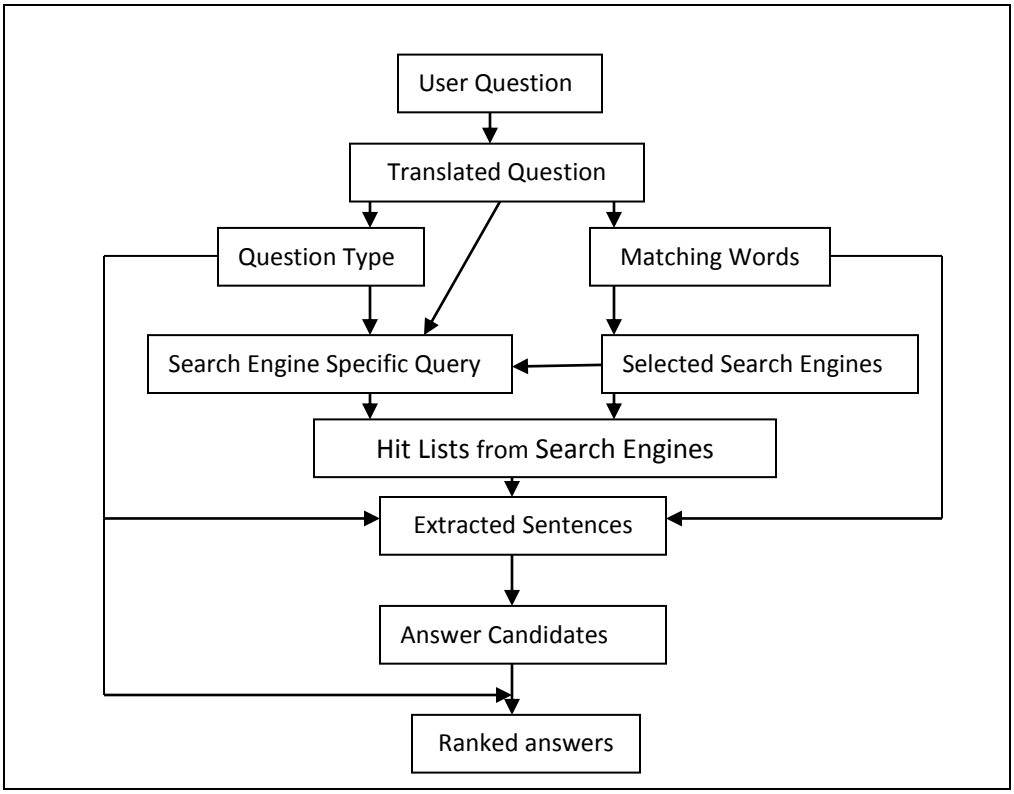


Fig. 2.2 Working process of AnswerBus

The rest of the process is completed into four steps:

i. Selection of two or three search engines among five for question answering and conversion of question into search engine specific queries.
ii. Retrieve the documents appearing at the top as the result of providing the queries to the selected search engines.
iii. Extraction of sentences from the documents that is likely to contain answers.
iv. Ranking the answers and return the choices at the top with the contextual URL links to the user.

2.4.1.1 RELEVANT DOCUMENT RETRIEVAL

AnswerBus aims to retrieve enough relevant documents from search engines within a response time that is acceptable to users. The main tasks involved in relevant document retrieval are as follows:

- Search engine selection: For answering a specific question, AnswerBus chooses to use two or more specific search engines among the five. For this, it collects 2000 sample questions and sends the queries to all of the five search engines. The answers are then recorded. All the words used in the queries are indexed. For example, for query q1, Google returns 8 answers, AltaVista returns 4 answers and Yahoo returns 7 answers. For query q2, Google returned 6 answers, AltaVista returned 6 answers and Yahoo returned 5 answers. For query q1 and q2, AnswerBus returns the results by merging the results of Google and Yahoo.
- Search engine specific query formation: These queries refer to the queries formed from a user's natural language question that are given to particular search engines and produce optimal search outcomes. Optimal means the best outcomes in terms of both recall of documents and time to retrieve the documents for a QA system. If a question like "How tall is Mount Everest", is sent to the search engine like Google, the results given as the result are more likely to be irrelevant to the user, hence lower precision and longer will be used to retrieve and process the documents. Some approaches use query expansion like ORing the synonyms,

however this leads to increase in the complexity and search response time will prolong. So, here focus has been laid on generating one simple query instead of expanded one. Here, several approaches are combined to form queries including functional word deletion (prepositions, conjunctions, interjections etc. and others like “kind of” are treated as functional words), deletion of frequently used words and word form modification etc.

2.4.1.2 CANDIDATE ANSWER EXTRACTION

At this stage, AnswerBus downloads and processes the documents referred at the top of search results returned by different search engines. It uses sentence segmentation tool that deletes HTML tags, excludes non-contextual content, and regards some special HTML tags as sentence boundary indications to parse the document into sentences and then separate sentences that are answer candidates by the process of word matching. Each sentence gets a primary score on the basis of matching words in the sentence. The sentences with a score of “0” are generally discarded.

2.4.1.3 ANSWER RANKING

AnswerBus uses several techniques to refine the primary score assigned to each sentence which then decides the overall rank of a sentence. The techniques used for answer ranking are follows.

- Question type and QA specific dictionary: The question type is classified on the basis of the type of answer the user is expecting like “who is……” is assigned a “person/organization” type. Along with this, some other parameters are also used like “how far” is most likely will be a “mile, km, light year” and “how close” is most likely will be a “inch, cm”. The system uses a QA specific dictionary, a database containing this kind of information about the relationship of words between questions and answers. This piece of information is used to judge whether a sentence can be an answer to a question.

- Dynamic basic named entities extraction: This involves pre-tagging of the corpus or knowledge base on which a QA is based. For a question “how much money”, a sentence with the entity of “CURRENCY” receives higher rank.
- Coreference resolution: A sentence may contain words, such as “he”, “they”, that coreference to other objects described in the document. AnswerBus solves the coreferences in the adjacent sentences. After this, the later sentence receives part of score from its previous sentence.
- Hit position and search engine confidence: The rank of a sentence can also be related to the position that its source document is located in the hit list returned by a search engine. A sentence extracted from the first hit receives the highest score associated with hit positions and the score decreases as the position moves down.
- Redundancy: Different search engines can retrieve document for a question; one or multiple documents may contain same or very similar sentences. This leads to candidate answer sentence redundancy. For checking this redundancy, the system compares highly ranked sentences against one another. However, spaces, punctuation marks, special characters and the words with high frequency are not considered while making comparisons among the sentences.

2.4.2 AskMSR QUESTION ANSWERING SYSTEM

In [29,30,31], a question answering system is described that is designed to make use of the tremendous amount of data available on WWW. Unlike most of the QA systems that use linguistic resources, here the focus is on the redundancy available in large corpora as an important resource. This redundancy is used to simplify the query rewrites for mining answers from returned snippets. Also, some strategies are explored in this work for determining the incorrect answers given by the system.

2.4.2.1 SYSTEM OVERVIEW

A flow diagram of the system is shown in Fig. 2.3. There are four main components contained in the system discussed as follows:

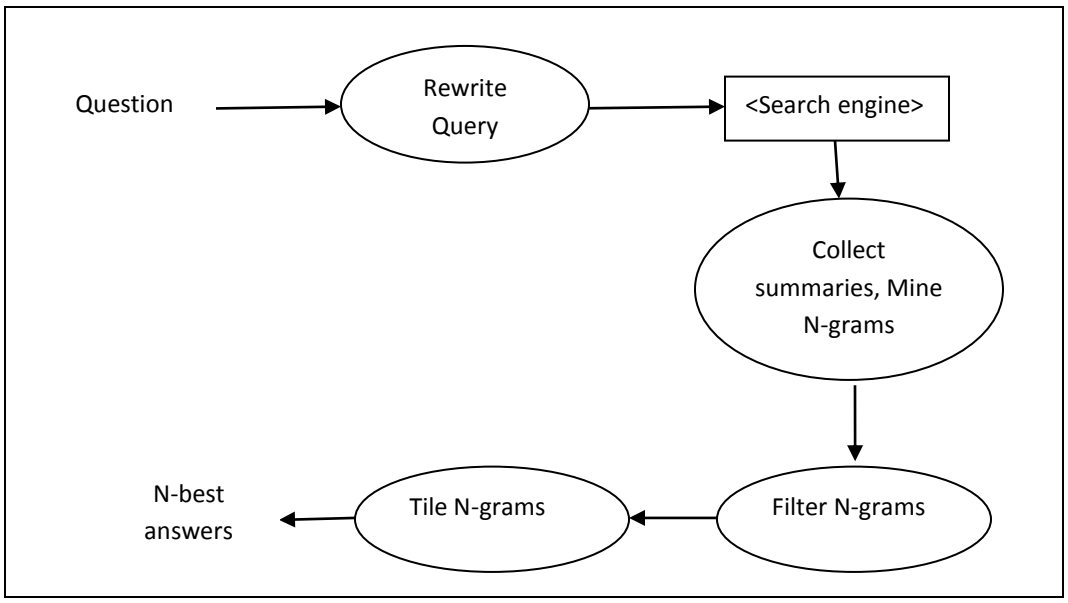


Fig. 2.3 AskMSR System architecture

- Rewrite Query: The first component takes a question as input; and generates a number of rewrite strings. These strings are likely substrings of declarative answers to the question. Let us take a question "When was Abraham Lincoln born?" For this question, it is known that a likely answer formulation takes the form "Abraham Lincoln was born on <DATE>". Therefore, a search can be performed in the collection of documents, in search for such a pattern. At first, it is important to classify the question into one of seven categories and each of which then needs to be mapped to a particular set of rewrite rules. As the output of this module, a set of 3-tuples is formed. The set takes the form [string, L/R/-, weight], where "string" is the query for which search is being performed, "L/R/-" indicates where we expect to find the answer with respect to the query string i.e. the position of the string (to the left, right or anywhere) and "weight" reflects the weight assigned to the answers found with this particular query.

- The answers are weighted according to the weights assigned to the rewrite query. The query may be a high precision or a low precision query. The idea behind using a weight is that answers found using a high precision query are more likely to be correct than those found using a lower precision query. Like the query "Abraham Lincoln was born on" is of high precision than the query "Abraham"

AND “Lincoln” AND “born”. So, the answers associated with the first query are given higher weightage than those associated with the second one. The rewrite rules and associated weights are created manually for the current system. The query rewrites generated by our system are simple string-based manipulations.

Consider how a verb “is” is moved while making a query rewrite for the question “Where is the Louvre Museum located?” The rewrite is “The Louvre Museum is located in”. This determination for where to move a verb can be done by analyzing the sentence syntactically. Given a query such as “Where is w1 w2 … wn”, where each of the wi is a word, for each possible move of the verb, a rewrite is generated, like “w1 is w2 … wn”, “w1 w2 is … wn”, etc.

For each query, a final rewrite which is a backoff to a simple ANDing of the non-stop words in the query is generated. Let’s take an example for the query “Who created the character of Scrooge?”, the rewrites are as follows:

a). LEFT_5_”created +the character +of Scrooge”

b). RIGHT_5_”+the character +of Scrooge +was created +by”

c). AND_2_”created” AND “+the character” AND “+of Scrooge”

d). AND_1_”created” AND “character” AND “Scrooge”

In this approach, the stop words like “in” and “the” need to be matched, like in the above example. So, here the stop words are important indicators of likely answers. The next step is to fire the query rewrites as search engine queries and the snippets provided by the search engine as the result are used as the page summaries are then collected and analyzed.

- Mine N-Grams: The next step is to mine n-grams from the page summaries returned by the search engine. For reasons of efficiency, only the returned summaries and not the full-text of the corresponding web page are used. The summaries returned by the search engine contain the query terms, usually with a few words of surrounding context. The summary text is then processed. The only

strings that are extracted are either to the left or right of the query string. The extraction is as specified in the rewrite triple. 1-, 2-, and 3-grams are extracted from the summaries. Then the extracted N-gram is scored according the weight of the query rewrite that retrieved it. The summaries that contain the particular n-gram are searched and these scores are summed across all those summaries. This is the opposite of the usual inverse document frequency component of document. The usual term frequency (tf) component used in ranking schemes i.e. the frequency of occurrence of n-gram within a summary is not used. Thus, the final score for an n-gram is based on the rewrite rules that generated it and the number of unique summaries in which it occurred. For the query "Who created the character of Scrooge? discussed above, the following are the top-ranked n-grams:

a). Dickens 117

b). Christmas Carol 78Charles Dickens 75

c). Disney 72

d). Carl Banks 54

e). A Christmas 41

f). uncle 31

- Filter/Reweight N-Grams: The next step is to filter the n-grams. After the step of mining n-grams, it is found that how well each candidate matches the expected answer-type. This is the criterion used for filtering the n-grams. The system uses the filters for this purpose. The filtering takes place in the following manner:

  First, the query is analyzed to find out what we expect as an answer in response to the query. For this the query is assigned one of seven question types, such as who, why, or how-many i.e. one question type is assigned to the query. Since, there are a number of filters for different question types with specific features, so on the basis of the query type that has been assigned, the system determines the filter to apply to the set of potential answers found earlier. The answers are then rescored

according to the presence of desired features. The system uses a collection of approximately 15 filters. After the application of filters to a pool of candidate answers, the score of the string is then readjusted. The score assigned to a potential answer is boosted using the filters. Some answer candidates may be removed in some cases like in case when the set of correct answers was determined to be a closed set (e.g. “Which continent….?”) or definable by a set of closed properties (e.g. “How many…?”).

- Tile N-Grams: Next step is to tile the n-grams that have been filtered for a question by applying an answer tiling algorithm. The algorithm works on the objective of merging similar answers and then assembles longer answers out of small answer fragments. Tiling forms longer n-grams from sequences of overlapping shorter n-grams.

  As an example, two n-grams filtered say "A B C" and "B C D" are tiled into "A B C D." The n-grams selected for the purpose are the top-scoring candidates. Also, a cut-off is decided for picking up the subsequent candidates that satisfy the criterion. These candidates are checked to see if they can be tiled with the current candidate answer. If so, longer tiled n-gram is used to replace the higher scoring candidate and the lower scoring candidate is removed. The algorithm proceeds until no further n-grams can be tiled.

  For the Scrooge query are, the top-ranked n-grams after tiling are as follows:

  a). Charles Dickens 117

  b). A Christmas Carol 78

  c). Walt Disney’s uncle 72

  d). Carl Banks 54

  e). uncle 31

2.4.3 QUESTION ANSWERING BASED ON SEMANTIC GRAPHS

The system discussed in [32] makes use of semantic graphs. Along with providing answers to questions in natural language, the system also provides explanations for the answers via a visual representation of documents and uses subject-verb-object triplets and their summaries. The system is not restricted to a specific domain; however it restricts the grammatical structure of the question to a predefined template. These facts are then used to retrieve answers. The triplets are then enhanced and on the basis of the enhanced triplets the semantic graph for the document is constructed. To generate the document summary, the semantic description of the document and the extracted facts is used.

2.4.3.1 SYSTEM OVERVIEW

The system combines three major functionalities: question answering, summarization and document visualization. The user poses a question to the system and the system responds with the answers. Each answer is linked to the sentence that contain it and the documents that contain these sentences. Also, a document overview is provided by the following:

- document semantic graph
- list of subject-verb-object facts and
- document summary of variable length

The description of the main components of the system is given in the next section.

2.4.3.2 QUESTION ANSWERING

The system at first extracts facts from the text and then the facts are represented using subject-verb-object triplets. The triplets are then indexed and a search is performed for any of their elements that are left undetermined. Like, in Fig. 2.4, answer for the question "where do tigers live" is identified using triplets. In general, a tree structure is used to organize a query in which the leaves of the tree are triplets with one or more elements undetermined. A database is prepared which stores the triplets constructed as explained above and then a search is performed for the query triplet formed as the result of analysis

of the question. For the example above, let the query triplet are (tigers, live, ?) and (tigers, inhabit, ?) and the stored triplets are (tiger, live, Sumatra) and (Indonesia, protect, tiger). Wordnet is used to find out the synonyms. As the answer to the question, "Sumatra" is returned.

- **Triplets:** The information contained in a sentence is represented using the triplet. It contains the subject, the verb and the object of the sentence, it represents. This is the basic unit of data on which the question answering system is built. This information is indexed for the fast retrieval. Thus, an index is maintained that consists of the triplets formed. The system uses a query structure that is formed for easy to reuse and extend.

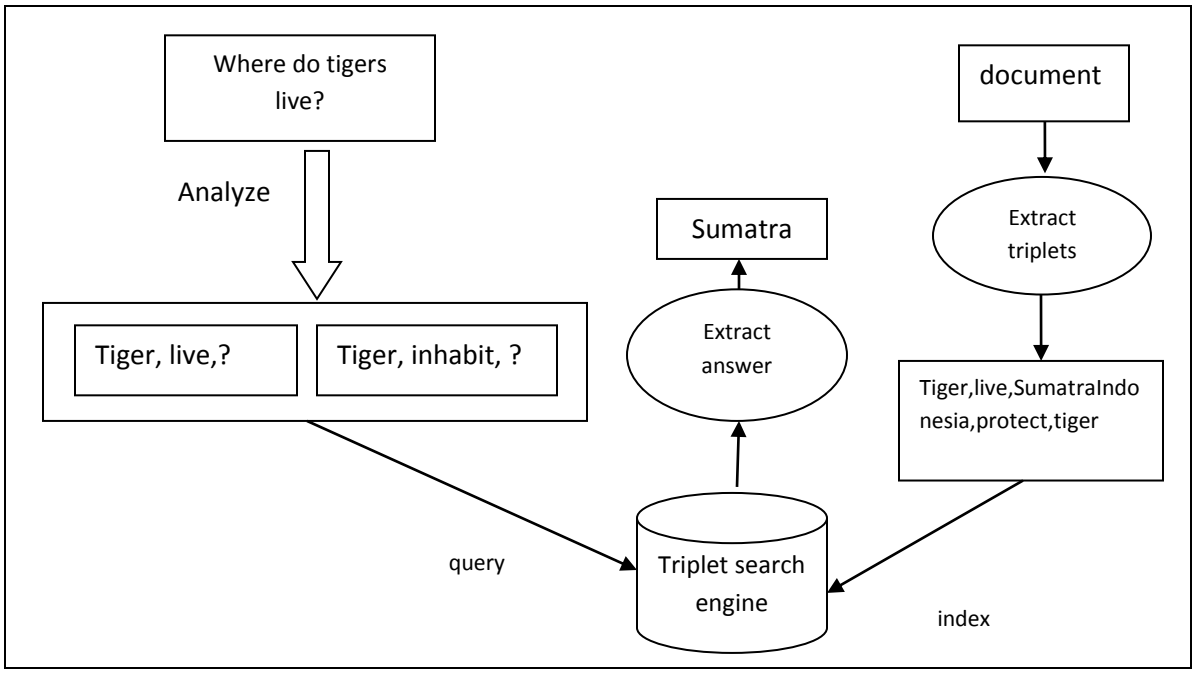


Fig. 2.4 Process of answering a question

- **Question Analysis:** The major task of this module is to determine the type of question and to form a query for answering questions. The system supports various question types like yes/no, list type questions, those starting with why, how much, where, when etc. The question is parsed using OpenNLP parser to obtain a parse tree.

- **Answer Generation:** The result of a query is a set of triplets as follows.

    i. If the question is of type yes/no, then the resulting set of triplets is split into two groups: Triplets in which the polarity of the verb matches the polarity of the verb in the question: the group that supports answer "yes" and the Triplets in which the polarity of the two doesn't match: the group that supports answer "no".

    ii. If the question is a list question, a quantity question or a location question, then the answer is the collection of items.

    iii. For a reason question or a time question, the system gives no clear answer. Instead the sentences that contain the triplets returned by the query are given as answer.

2.4.3.3 SEMANTIC GRAPH

Semantic graph provides an overview of the content contained in the document. The input document is processed and after the processing, it is then passed through following series of sequential operations explained below composing a pipeline, to obtain the graph:

- Text preprocessing: The original document is split into sentences.

- Named Entity Extraction: It refers to the names of people, locations and organizations, for retrieving semantic information from the input text. For this a toolkit for NLP named *General Architecture for Text Engineering* is used. Like, for the people, the information about their gender is stored. Similarly, for locations, the names of cities and countries are stored separately. This enables co reference resolution that implies identifying terms refering to the same entity. Also, entity matching is performed like in case of inclusion of one surface form into another ("Anna Maria" is same as "Anna Maria Smith"), when one surface form is the abbreviation

of the other ("NLP" stands for "Natural language processing") or when a combination of both of these ("A. Smith" and "Anna Smith") occur.

- Triple extraction: For triplet extraction, each sentence is taken into account without considering the surrounding text. For this, a Penn Treebank parser is applied and also the statistical Stanford Parser and the OpenNLP parser is employed for question answering. For the triplet extraction, the pure syntactic analysis of the sentences needs to be performed.

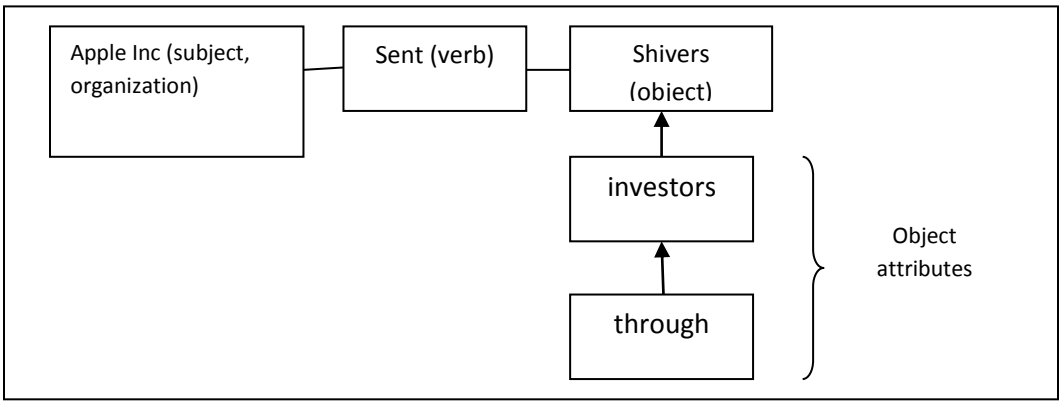


Fig. 2.5 A triplet extracted from sample sentences

Fig. 2.5 shows an example of a triplet extracted. It shows the subject, verb, and object and also the object attributes.

- Triplet Enhancement and Semantic Graph Generation: Triplets are enhanced by first resolving anaphors for a subset of pronouns: {I, he, she, it, they}, their objective, reflexive and passive form and the relative pronoun "who". For this, the triplets are linked to their corresponding co referenced named entity. Also, the subject like "Apple" would be linked to co referenced named entity "Apple Inc"). Further all the pronouns in the document are searched for replacement. The triplet elements that share the same meaning are merged using Wordnet.

- Finally a directed Semantic Graph is obtained in which the arrows move from the subject node to the object node and the connecting link is a verb. Fig. 2.6 presents a semantic sub-graph of a text portion.

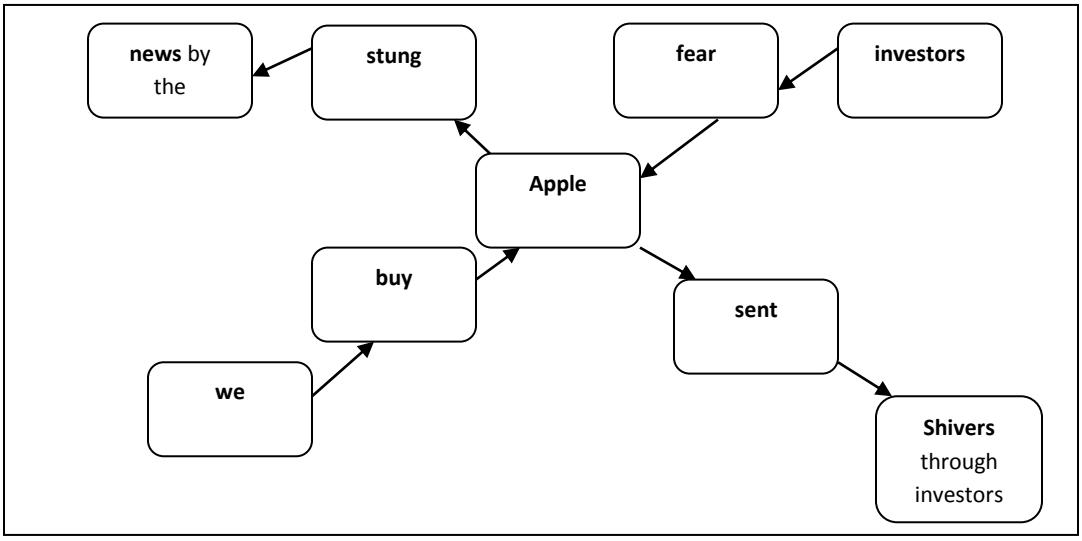


Fig. 2.6 An example semantic sub-graph

2.4.3.4 DOCUMENT SUMMARIES

The summary of the document consists of sentences from the text, with the sentences in the same order as in the original text. The summarization technique used involves training a linear SVM classifier to determine those triplets that are useful for extracting sentences for summarization by considering some features. As an input to the SVM classifier, the initial document and the semantic graph are used and as the output, it provides a score to each sentence termed as SVM score as the output. The sentences are then ordered based on these scores in descending order.

The work done so far on QA systems [28,29,30,31,32] uses general web pages as their source of information. But it has been observed that for question answering, there is a need of the sources comprising of topical information. The blogosphere [61,113] has been found comprising of such kind of information. A brief introduction to blogs and blogosphere is given in the sections below.

2.5 BLOGS: AN INTRODUCTION

An introduction to blogs and the related content is given in this section.

2.5.1 BLOG

A blog [33] is an online journal which is generally composed of media-rich articles called "posts". Blogs are also referred to as "weblogs". A weblog [34] is defined as a web document which doesn't require any external editing. It is composed of posts which are updated on periodic basis and are presented in reverse order of their occurrence in time i.e. in reverse chronological order. A blog may consist of a number of hyperlinks to other online sources. Blogs can be treated as a space to write about oneself in form of personal diaries, as columns for technical advice, for chat, political commentary, educational purposes, opinion sharing, discussions etc. Blogs can be published on any matter that can be thought of. Also, any person can easily publish a blog without expenses. It has been found that the majority of blogs are written in English.

In general, a blog can be thought of as a website with a number of pages. A person can visit a blog page and can read a blog page in the same way as (s)he can read a web page. Some of the popular blog sites that facilitate a user to write and publish his blog are technorati, wordpress, blogger, livejournal, typepad, travelpod etc.

Since, a blog page is written by either a single person or a small group, so a consistent style of writing is used across the whole text. Blogs can be classified according to their purpose: *personal blogs* (documenting one's life), *issues* blogs (expressing opinions, writing comments, debating current events) and *topical blogs* (serve as community forums allowing users share their ideas with each other). Blogs have a wide coverage and thus cover most of the topics like education, entertainment, sports, music, health, business, agriculture etc. It has been found that blogs written on a topic are likely to contain details on the topic being discussed. So, these contain the relevant in depth information on the topic of interest. It shows that the information on blogs is very much relevant as compared to that present on remaining part of WWW. Blogs Pages are the richest source of information where people express their opinion on various topics and

situations. Blog written by one may be shared by others. Blogs are search friendly and attract more traffic for its fresh and dynamic contents. Blog posts are written by experienced person called Blogger [12] on various topics. The content on the Blog pages is likely to be related to the topic on which the Blog is being written.

2.5.2 COMPONENTS OF A BLOG

In terms of content organization, a typical blog is composed of three main components [35], depicted in Fig 2.7:

- One or more units of content- Each unit of content is said to be a blog post written by the blogger using HTML. A blog usually covers a single topic and includes one or more comments added by the readers of the blog. A blog is assigned a permanent URL (known as a permalink) for its unique identification.
- A syndicated XML feed: Since, the blogs are updated on periodic basis, there is an XML feed added in the blog. Whenever there is an update in the blog content, the digital content that is frequently updated is published and organized in the feed. The feed gives the ability to the readers to subscribe to the blogs. For subscription there are some client applications that run, known as aggregators, feed readers or news readers. When a reader subscribes to RSS Feed, the recent blog post is delivered to his mailbox automatically. The updates appear just like the emails appearing in the mailbox, latest content on the top, with the headline and the first few lines of the post. There are two XML standards used in common for blog feeds, namely Really Simple Syndication (RSS) [100] and Atom [100]. RSS is in commonly used. In addition, some blogs provide feeds for also retrieving comments.

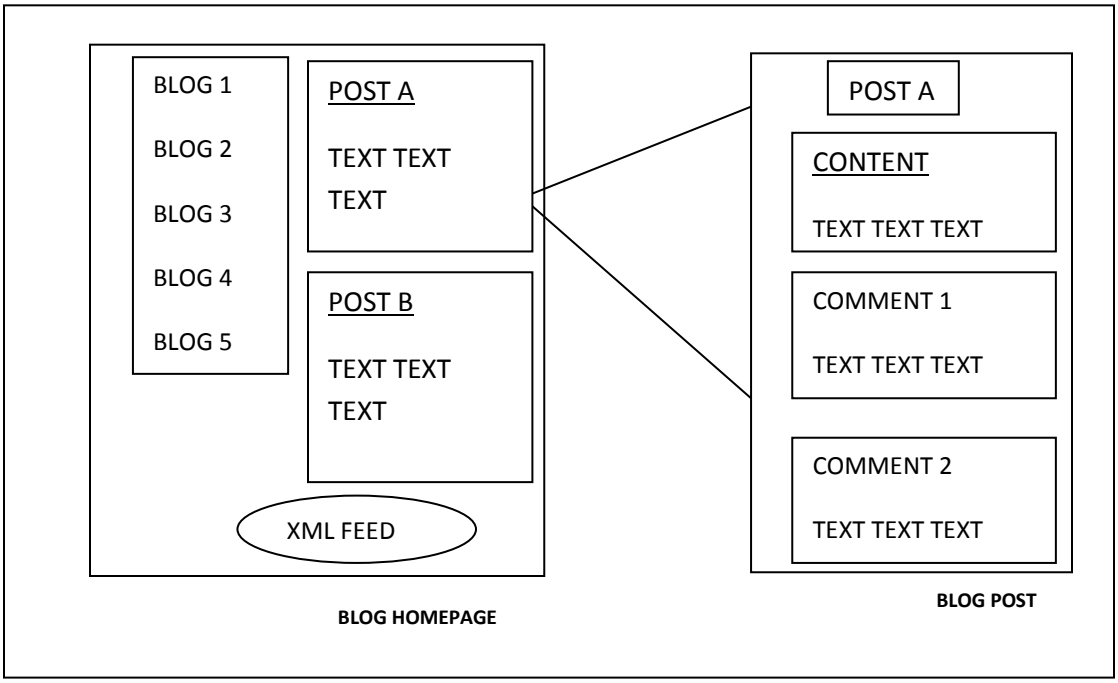


Fig. 2.7 A typical Blog

- An HTML homepage: A blog can be considered as an HTML page with the posts organized in a reverse chronological order. The content on the blog content is dynamic because it can be expanded, modified, or removed at any time. The content on the blog is not necessarily text. Besides text, a blog may consist of specific types of data. A blog may consist of audio (podcasts), images (photoblogs), video (vlogs), etc. A user may be interested in publishing very short content (e.g., a 140-character long post) which focuses on his up-to-the-minute thoughts. These are known as Microblogs (eg. Twitter). A blog page may consist of a number of comments given by the readers of the blog. There is a list of "friend" blogs i.e. those blogs that are somehow related to the current blog or in which the blogger is interested. This list is known as a blogroll. The blogrolls facilitate the user with the blogs that cater to his topic and thus save his time and effort.

2.5.3 THE NETWORKED STRUCTURE OF THE BLOGOSPHERE

The universe of blogs is conventionally referred to as the blogosphere [34,40]. The structure of the blogosphere is networked as it relies on hyperlinks and may contain links to other blogs. The bloggers maintain a blogroll on their website containing a list of blogs that the blogger frequently read or admire. The list also includes clickable URLs to link the blog pages that are added in the list. Blogrolls is one of the important means of finding out bloggers interest and preferences within the blogosphere. Also, the posts contained in the blogs written by the bloggers may point to other blogs. A blog may contain comments to the blog posts of others. Posts that contain comments on other's posts are a way of information exchange in the blogosphere. In this way a chain on bloggers commenting on blogs is maintained. The links and page views are very important for blogosphere. A blogger is always keen to have wide readership. Linking on another weblog is the most reliable way to have readership. The basis of this is the hypertext. Finding a link to another blog, the reader reads the blog that is being linked by the current blog. Because of this, the reader may find the second blog more useful and may become regular reader of the second blog.

Also, a blogger is interested in knowing who links to him i.e. all the incoming links. Also, when the reader finds important source of information on another blogs, they are likely to credit the sources in their comments and may provide links to them also for other readers. So, blogs are linked through the use of hyperlinks such that they form a network. Each blog is treated as a node or vertex and each hyperlink is treated as an edge or an arc connecting blogs. The number of incoming and outgoing edges is called indegree and outdegree respectively.

2.5.4 WHY DO PEOPLE BLOG?

The bloggers [35] have been classified into two categories: specialists and generalists. Specialists are those bloggers who write their blogs on some specific topics like sports, gaming, politics, medicine or technology etc. These bloggers receive a large number of visits and comments also on their blogs. The other category is generalists, the people who

write in the blogs as if they are writing in their personal diaries-about themselves and their activities carried out by them on daily basis. These write for a small number of people. Their blogs are generally read by a small number of people. The studies [41] have shown that half of the blogs available online are written by males and half are written by females. So, there is an equal distribution between the both. Also, only 7% of bloggers are the people over 50 years in age, about 50%bloggers are those aged 21-35 years and 20% are aged 20 or under. So, most of the bloggers are teenagers. The people may write blogs individually or in a group. Some people write their blogs individually while some of them prefer to write in groups. Later type is referred to as group blogging. One example where this type of blogging is used is corporate blogging. Like in an organization, employees can communicate with each other in an effective way through these types of blogs. Because of higher link popularity and longer post lengths in group blogs, they are likely to be regarded as of high quality.

2.5.5 THE BLOGOSPHERE

The collection of blogs [35,40,41] on the web, has given a new direction not only to the way of information consumption but also to the way the information is produced in an intelligent manner. There is no external entity to manage the content on the blogosphere but they are the independent bloggers who manage the content. The major feature is that the blogs enable interaction among the people i.e. among those who write and those who read the blogs written by others. Also, the facility to provide reviews to the blog's content in form of comments enables interaction and effective communication. Also, the bloggers can themselves be the readers of other blogs, so the role of information producer and consumer may be performed by both.

The blog linking is another way to communicate in the Blogosphere. There are following three categories of inter-blog links that have been found in the blogosphere.

- Blogroll links: These links are generally placed on the blog's homepage and are the links to the related blogs or the friend blogs. This relationship is presented using the concept of Blogroll links.

- Citation links: These links are just like the hyperlinks present on the general web pages. These show the relationship of the current blog or blog post to the blog page or post with which it connects.

- Linkbacks: These types of links are also called trackbacks. It is a way used to find the people who are linking to your post i.e. to find the incoming links.

Due to these types of links, there is huge interaction among the bloggers in the blogosphere. Thus, the blogosphere can be thought of as a network of interconnected bloggers. The blogs that are written by experienced and authoritative bloggers are read and followed by large number of people.

2.5.6 BLOG SEARCH

A detailed discussion on blog search is given in this section.

2.5.6.1 BLOG DIRECTORIES

With the increase in the size of Blogosphere, there is a need of tools to find the blogs to which the reader can subscribe. One way to find the blogs which are related somehow to the current blog of the blogger is a list of blogs present on the right hand side of the blog page. The list helps the readers of a blog to find the other interesting blogs. But the process is not useful for the searching of the blogs [43] that are not mentioned in the blogs being read by the reader. Blog directories such as Blogflux[35, 70] and Topblogarea[35, 71] provide a search area in which the user can write for the blog to search. A snapshot of Blogflux is shown in fig. 2.8.

The interface of Blogflux provides a space to the user to enter the topic of the blog he/she is looking for. Also, the facility of blog tagging is provided in which the user can search for a particular category of blog.

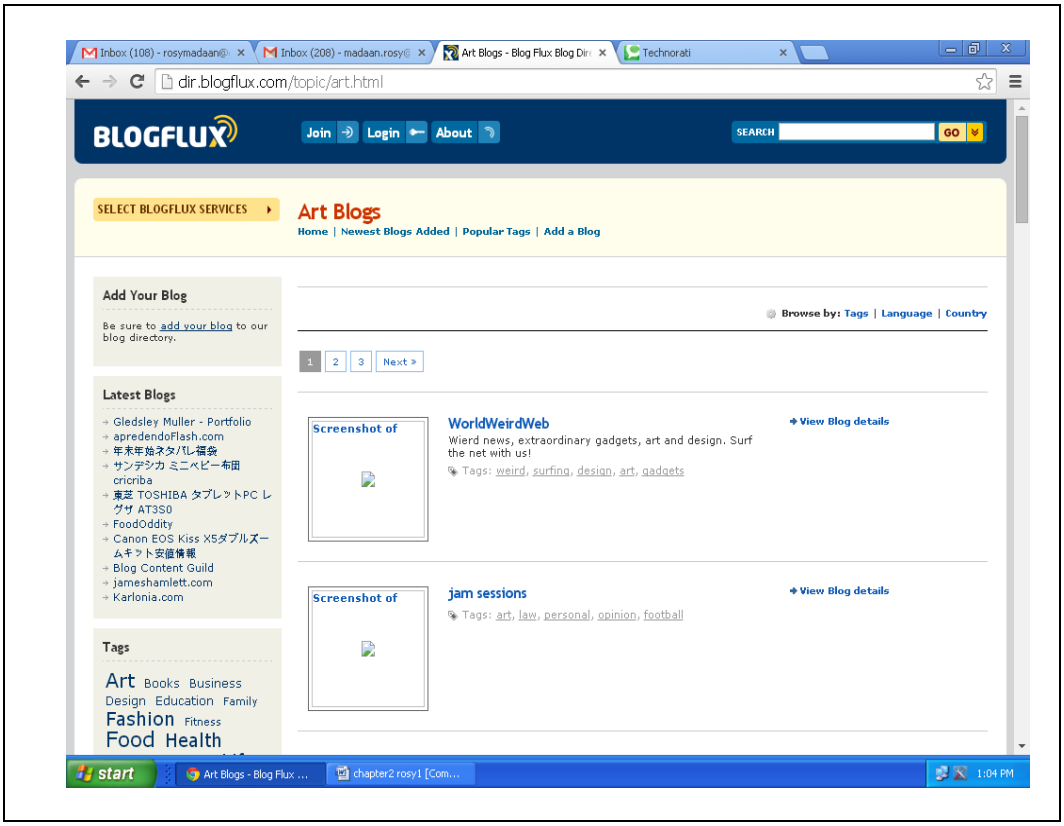


Fig. 2.8 An example Blog directory: Blogflux.com

The introduction to the blog search engines is given in the next sections.

2.5.6.2 EXISTING BLOG SEARCH ENGINES

The emergence, growth and popularity of the blogosphere, lead to the introduction of the blog search engines. These special types of search engines allow the user to enter a query for blog post search. As a response to the query, a list of relevant blog posts is returned to the user. Two major issues associated with most of the blog search engines are freshness and recency. Therefore, the blog search engine needs to provide the user with the up-to-date information with the most recent blog posts on the top.

- Technorati [35,88] is one of the most definitive blog search engines, since 2002. Technorati (see Fig. 2.9) supports the searching of blogs or their individual posts by providing two choices to the user in form of slider switch interface. It has also been found that the Google search engine give more importance to the fresh results than the Technorati for initial relevance ranking[63,64,65]. In contrast,

popularity is a major factor used to rank [66,67,68] the blogs in Technorati and gives more importance to the deemed authority. Technorati is a real-time search engine dedicated to the blogosphere. Its database consists of blogs and it only searches through its database to find what the user is looking for. It is very easy to search for blogs on Technorati. Technorati's home page provides main search query bar to write query. Technorati provides more advanced search options to the user i.e. the user can search for more search parameters.

There are a number of tags that it provides to the user. These are basically subjects or topics that bloggers assign to their blogs. The tags shown on the Technorati Tag page are organized in alphabetical order.

Technorati has a blog directory known as Technorati Blog Finder organized by topic. There are number of categories in the directory and the user can browse through for the blog of interest. It also contains the most recently added blogs. Technorati has a popular list that shows people searching and there is an option of *What's Popular* with main categories like News, books, movies etc.

Another type of blog search engine comes into picture that monitors the blogosphere from a temporal perspective. As blogosphere is a repository of huge subjective content and deals with full discussions, mining for insights and opinions about products or a company can be very useful. Technorati can also be used to track various trends and topics on the Web on day to day basis. Therefore, Technorati is recommended as a great way to search the blogosphere.

Fig. 2.9 Searching for blogs in Tecnorati.com

- In September 2005, Google [36] introduced the search of blog content on its website. Google introduced its Blog Search to expand their vertical search. In this, Google focuses on a specific sphere, namely the blogosphere. Because of good reputation of Google and its large index of web search, it soon became the main competitor of Technorati.

  .

2.5.7 SIGNIFICANCE OF BLOGS

Since, Blogs [37] provide an opportunity for sharing knowledge, sharing ideas, debate etc., which attract a large number of audiences. These facilitate open discussions, thus making blogs an ideal place for communication and large distant discussions on new and emerging topics and trends. Blogs also bring up the communities and allow collaborative sharing and recommendations. Blogs are growing at an exponential rate and occupy a large amount of space on the web. Blogs are easily approachable and they exist in close association to other blogs. Also, the blogs serve as an educational tool for the building, sharing and reflection of knowledge. The value of blogs is increasing day to day in the

field of education. The content on the blogs can be shared among individuals by using RSS feeds. Blogs allow students, faculty, staff and the other people associated to have peer to peer interaction and the students can learn a lot from one another as from the teachers and/or books and can explore further.Thus, these provide an excellent mechanism for knowledge sharing and acquisition.

Blogs [38], if seen from a teacher's point of view, can be seen as a class notice board. It can serve as a discussion tool with the students. From the student's point of view, it can also be used as a learning tool. Blogs encourage students to write and also to read on a topic they wish to comment on. Blogs allow the students and teachers to continually search and filter for the posts. Also these allow the users to post ideas and information which engage higher order thinking skills. Since, commenting is an important feature of the blogs, so the comments can be treated as feedback and the students can use them to improve on their work. Also, like websites, the blogs allow the bloggers to embed the other Multimedia elements other than text like video, audio or flash movies. One can also attach word processing, spreadsheets and pdf files into a blog. Blogging helps to enhance the following skills [39]:

- Sharing — thoughts, concepts, experiences, knowledge
- Analyzing
- Reflecting — Critiquing, Writing, Questioning, Reacting
- Communication
- Record keeping — thoughts, concepts, and experiences
- Collaboration — with peers, people (experts, students) around the world

A component based search engine for blogs is given in the next section.

2.6 COMPONENT BASED SEARCH ENGINE FOR BLOGS

A component based architecture has been proposed in [44] for searching in blogs for the user's query. There is a major step involved in search engines termed as Information Extraction. A blog page consists of text and a lot of other kinds of information like links to advertisements, copyright notices, navigational links, links to archives and new

content, website menu etc. This all is regarded as *noise* in the blog page. Thus there is a need to extract the relevant information to improve the search accuracy. The blogs that are written by the blog authors contain useful and valuable information. Because of the presence of a large amount of noise in the blog page, the extraction of the useful content within the page is harmed. This is really a big problem that needs to be overcome for efficiency of blog search engine. The task of separating the noise from the rest of the blog's content is really a difficult task. The reason behind this is that the blogs are written by different authors in their way and style. Also, the template used may differ from blog to blog. Some author may also use design their own template as per their need and preference. Thus, the problem of separating the noise and data arises. If these aren't separated, then each of the word in the blog page is indexed, thus reducing the accuracy of the search results. In [44], instead of separating the content and other parts of a blog page, the authors have assigned a score to each of the component. By assigning a score, the content within the page is given a high score as compared to the text in the other parts of the page. As the result, the blog with the rich content appear at the top of the search results. Consider the snapshot of an article in a tourism blog as shown in Fig. 2.10.

The snapshot consists of five parts as explained below:

Part-1: This part contains the navigational links of the site.

Part-2: This part contains the archive links.

Part-3: This part contains the links to the latest contents.

Part-4: This part contains the title of the article.

Part-5: This part contains the main contents of the article.

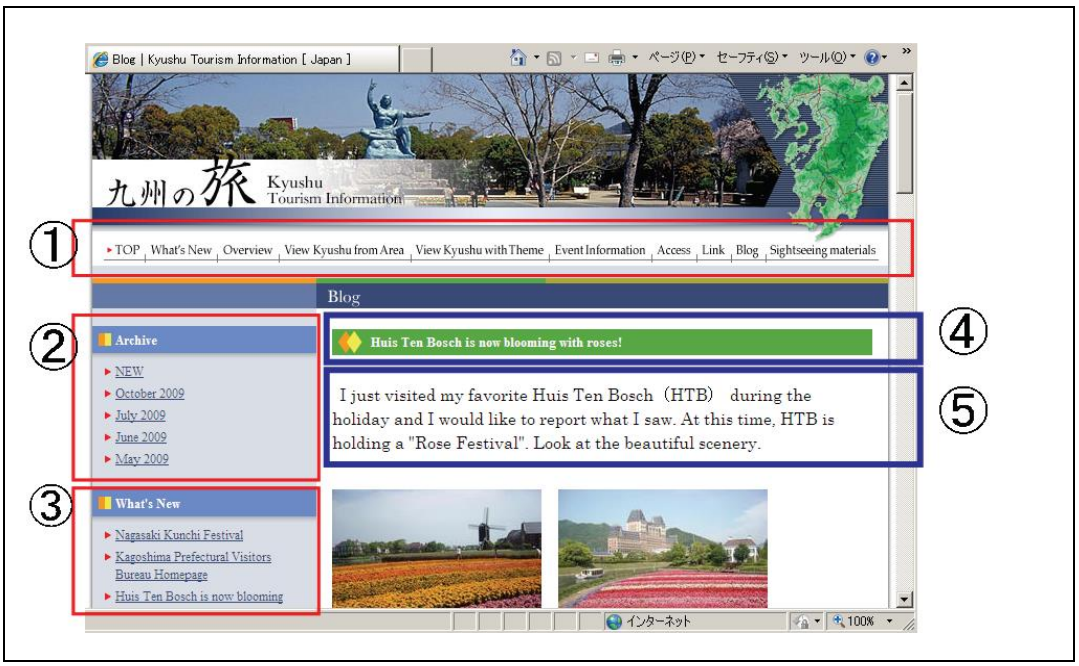


Fig. 2.10 A Tourism blog

The HTML tag tree of the Tourism blog is shown in Fig. 2.11.

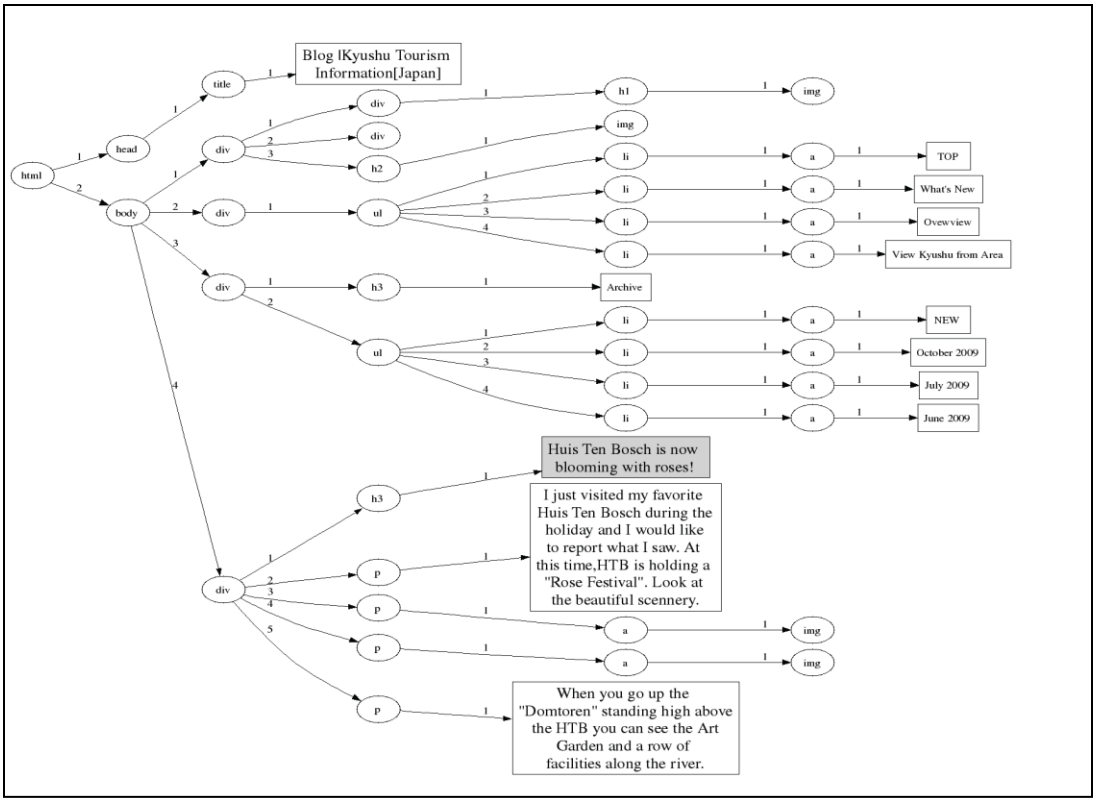


Fig. 2.11 HTML tag tree of the Tourism blog

The square boxes in the Fig. 2.11 show the text areas in the page. These are termed as "components". These components are then indexed for searching in the search engine. The index file of the search engine is shown in Fig. 2.12. In the component index shown in Fig. 2.12, the lines that start with @ symbol show the components that have been indexed. The components used in the index are explained as follows: "@1-1" and "@1-2" represents the 1st and 2nd text areas of the 1st HTML file. Also, the other lines of the index display the index keywords and their frequencies in the text area. Id of the HTML file starts with h: and the path from the root of the HTML tag tree to the component starts with p: as shown. It has been found that the linked text is usually relevant to the other web pages and not for the search engine. Thus, it is recommended to filter out the links under the anchor tag from the components. There may be some links under the anchor tag that may point inside the page. Therefore, eliminating these links doesn't lose any information from the page.

```
@1-1
1 h:1
1 p:/html[1]/head[1]/title[1]
1 Blog
1 Kyushu
1 Tourism
1 . . .
@1-2
1 h:1
1 p:/html[2]/body[4]/div[1]/div[1]/h3[1]/
1 Houis
1 Ten
2 Bosh
2 is
1 . . .
@1-3
1 h:1
1 p:/html[2]/body[4]/div[1]/div[2]/p[1]/
3 I
1 just
1 visit
1 . . .
...
```

Fig. 2.12 Component index of the Tourism blog

There are a number of methods that have been proposed to evaluate the importance of a word in a document. For finding the score of the entire document, the score of each of its components need to be found. Then the summation of the scores yields the score of the entire document. The formula used for the calculation is:

$$\text{Score } (C_i) = \text{NW } (C_i) * \text{depth } (C_i) \quad \text{.....eq. 2.1}$$

In eq. 2.1, NW (C_i) is the number of the distinct words present in the component C_i and *depth* (C_i) is the length of the path from the component to the HTML tree's root.

Finding that the content which is detailed exists in the web page deep inside whereas the content treated as noise like links to ads etc. exists at the shallow nodes. Therefore the formula uses the function *depth* for the calculation. By summing up the score assigned to the individual components, the score of the whole document is obtained.

2.7 ISSUES IN EXISTING APPROACHES OF QUESTION ANSWEING

As the result of the literature review on the existing Question Answering Systems, it has been found that the issues shown in Table 2.1 need to be addressed while designing an efficient Question Answering System for answering user's question belonging to different domains:

Table 2.1 Issues in existing approaches for question answering

S.NO.	APPROACH USED	ISSUES
1	ANSWERBUS QA SYSTEM	(i). In this system, the relevant web pages are retrieved from the five search engines and directories and which further are used to extract the sentences that contain answers but it is not specified that how many search engines are used for searching in case of a particular query.

		(ii). It is not clear from the available literature that for which type of questions the system works. (iii). The module of Question Analysis is not discussed in much detail and needs to be explored.
2	ASKMSR QA SYSTEM	(i). The answers are extracted and ranked on the basis of query rewrites but much details on how to write different kinds of user's query is not given. Also, the criterion for scoring the answer candidates on the basis of the query rewrites is not clear. (ii). The strategy for finding the inaccurate answers is not given but stated. (iii). Much details on how to mine, filter and tile n-grams is not given.
3	QA BASED ON SEMANTIC GRAPH	(i). One major drawback of the system based on Semantic Graphs is that it restricts that the question must be asked in a predetermined format.

There are some issues of concern that are found common in all the above discussed literature as follows:

- None of the previous work in this field have focused on all the aspects associated with Question answering like summarization, question analysis, document

representation, answer extraction or ranking. Thus, it has being analyzed that there is a need to design and develop a system that is user interactive and combines all the modules together for efficient question answering.

- The existing systems have not much focussed on the indexing of the Web pages for answering the questions. So, there is a need to design an efficient indexing scheme for designing and developing a fast QA system.

- The referred work in this section collected the pages from the Web. However, it has been found that the information contained in web pages is not of high quality and also, the major portion of the web pages contain information that is not of much importance to the user. Therefore, the web is a huge repository of information but is not likely to contain the information that is focussed. For this purpose, the system to be developed must be able to collect information from some other sources that are likely to be containing focussed information. Also, there is a need to extract only a relevant portion of the text from the pages collected from the quality sources.

A design of a search engine for prospective question answering is being proposed in Chapter-3 that not only addresses the problems prevailing in the recent QA systems but also uses the blogosphere as the major source of the information for improving the quality of question answering.

Chapter II

INFORMATION RETRIEVAL & QUESTION ANSWERING SYSTEMS: A REVIEW

2.1 INFORMATION RETRIEVAL

World Wide Web (*WWW* or *Web*) [1,2,3] is a largest collection of hyperlinked documents spread over the internet. The strategies for information retrieval [1,5,9,20] have been changed recently due to the increase in the size of publically indexable information [7,8,15,24] available over the WWW. Therefore, the field of information retrieval covers a broad spectrum of techniques and applications that aim to satisfy the user's information needs. An ideal information retrieval system must be able to

- determine the information [75] needs of a user,
- search the information available,
- return the relevant information that is generally compiled from multiple sources, in a language and format that can be easily understood by the users.

In IR, the information need of the user is expressed as a bag of keywords [1,5,20]. The results are returned in the form of a list of documents that contain one or more of those keywords.

Web Search Engine [2,3,4,25,26] is basically an information retrieval tool that provides search interface to the information seekers so as to allow the users to submit their information need in the form of queries and return relevant web documents from a large repository consisting of the terms in the queries. The documents are retuned in such an order that the documents appearing at the top are highly ranked and those at the bottom of the list are lower in rank. Thus, the user clicks and sifts through the documents at the beginning of the list. To decide the importance of a document, the search engine uses its

ranking mechanism. So relevance of the returned documents is generally decided by ranking mechanism that orders the set of retrieved documents.

2.2 SEARCH ENGINES

A search engine [2,3,4,25,26] is an information retrieval system designed to find information over the Web consisting of hyperlinked documents. The search engine provides an interface to the user that enables him to specify what he needs to search. i.e. to specify the criteria about an item of interest. Then a search is performed in the locally maintained databases. The criteria specified in the search interface are referred to as a *search query*. In the case of text search engines, the search query is typically expressed as a set of words separated by white spaces. The search is then performed for the required information contained in text documents, pictures files, sounds files etc.

The above discussed search model was developed in the 1960s [109] and have taken decades to grow in form of a new search model. In fact, as of June 2000, there were at least 3,500 different search engines (including the newer search engines) [109]. The components of a general web search engine are discussed in next section.

2.2.1 COMPONENTS OF A WEB SEARCH ENGINE

The various components of a general Web search engine [2,3,4,25,26] are discussed in detail as follows:

- **Crawler Module:** As compared to traditional document collections which reside in physical warehouses such as the college's library, the information available on WWW is distributed over the Internet. Because of the tremendous growth of the information available on the web, a component called crawler [4,6,25,26] is employed by the search engine which visits the Web, downloads the web pages and categorize them. Crawlers [16,17,18,19] may be defined formally as *"Software programs that traverse the World Wide Web information space by following the hypertext links extracted from*

hypertext documents". The crawler traverses the web [22] and downloads the web pages. From the downloaded web pages, the crawler extracts the hyperlinks which are queued and traversed later on by the crawler. The downloaded Web pages are temporarily stored in a local storage of search engine, called page repository.

- **Page Repository:** It provides storage to the pages downloaded by the crawler and those new pages remain in the repository until they are sent to the indexing module, for the purpose of creating an index for information search [21].
- **Indexing Module.** The indexing module takes each new page from the page repository and indexes it. The index holds the valuable information for each web page. *Indexing* [4,25,26] is the process by which a vocabulary of keywords is assigned to documents in which they appear, thus creating an index and such index is generally termed as *inverted index* [4,25,26].
- **Query Module:** The query module takes a user's query as input and search in the various indexes in order to respond to the query. For example, the query module consults the inverted index to find which pages contain the query terms. The pages given as output are the relevant pages, which are then passed to the ranking module for the purpose of ranking [100].
- **Ranking Module.** The ranking module [4,25,26] takes the set of relevant pages as input and ranks them according to a given criterion. The generally used criterion are popularity score, content score etc. The output of this module is an ordered list of web pages such that the pages appearing on the top of the list are the pages with the highest rank. The ranking module is the most important component of the search process because the output of the query module often results in thousands of relevant pages that the user otherwise must sift through. There are two types of scores used for ranking of a page: the *content score* and the *popularity score*. These two scores are calculated for each web page and then these are combined for the overall score. The *popularity score* is determined from analysis of the Web's

hyperlink structure. Many web search engines give pages, using the query word in the title, as a higher content score as compared to the pages containing the query word in the body of the page. The set of relevant pages resulting from the query module are then presented to the user in order of their overall scores.

2.2.2 TYPE OF DATA RETRIEVED BY SEARCH ENGINE

The type of data retrieved by the Search engine [2,3,4,25,26] is listed as follows:

- **Large volume:** WWW contains huge collection of data. Also, the growth of data over the WWW is exponential. Increase in the amount, poses scaling issues that are difficult to cope with.
- **Distributed data:** Data is distributed widely over the WWW. It is located at different sites and platforms. The communication links between computers vary widely.
- **Unstructured and redundant data:** The data on the Web is highly unstructured [48]. It is impossible to organize and add consistency to the data and the hyperlinks. Also, there exists semantic redundancy that can increase traffic.
- **High percentage of volatile data:** The data on the Web is highly volatile. Documents can be added or removed easily in the World Wide Web. These changes to the documents are usually unnoticed by users.
- **Quality of data:** The data available on the Web is not of high quality. A lot of Web pages do not involve any editorial process. That means data can be false, inaccurate, outdated, or poorly written.
- **Heterogeneous data:** Data on the Web is heterogeneous. They are written in different formats, media types, and natural languages.
- **Dynamic data:** The content of Web document changes dynamically [10,11,12]. Some web pages are highly dynamic and some are less. The web pages that changes dynamically need to be updated [13,14], so that the user gets an updated page on visit.

Web is massive, much less coherent; it changes more rapidly, and is spread over geographically distributed computers. This requires new information retrieval techniques, or extensions to the old ones, to deal with the gathering of the information, to make index structures scalable and efficiently updateable, and to improve the discriminating ability of search engines.

Given a query, a set of keywords, the search engines [2,3,4,25,26] retrieve the information in the form of documents. In case, if the user needs an answer to a question [101,103], the user has to go through the entire document to extract the answer to his question, which is a time consuming process. There may occur a situation in which user is interested only a portion of the web page rather than whole document. For example "What is the height of Eiffel Tower? ". Search Engines fails in such cases. Thus, there is a need to build a system which is capable of taking user's question as input and return answer(s) as the result. Hence, need of Question answering systems [45, 46, 76, 79] arises. To validate this, a survey has been conducted in this work in which a questionnaire has been prepared and distributed to 60 persons, some of which are teachers, students, technical and non-teaching staff. As the result of the survey, responses have been collected and critically analyzed. The analysis of responses of survey conducted that for efficient retrieving of information, there is need of a Question Answering system.

Question answering [45,46] offers a more intuitive approach to information processing than search engines. Given a collection of documents and a natural language query posed by user in form of question, a question answering system attempts to find the precise answer or at least a portion of the text in which the answer appears. The next section discusses a general Question Answering System in detail.

2.3 QUESTION ANSWERING SYSTEMS:AN INTRODUCTION

Question Answering [45,46] is a discipline of computer science within the fields of information retrieval that involves building a system which allows user to ask a natural language question and get answer(s) in return. A Question Answering system can be implemented using a computer program which generates answers by querying an

unstructured collection of natural language documents. The goal of question answering (QA) is to provide a relevant answer to a question posed in natural language.

A standard QA system is made up of three modules: *question processing, document processing, and answer processing* [45,46]. The main contribution of the question processing module is to identify the question type (who, when, where,. . .), and the expected answer type which predicts the type of entity that question requires as an answer. Document processing is concerned with retrieving relevant documents and extraction of passages that contain answer(s) and their ordering, in a very similar way to an IR system, as described above, and is often implemented as such. Answer processing uses the information provided by the other modules to pinpoint the answer within a passage and is also concerned with their ranking.

The basic form of question answering takes a question such as "*Who is the President of India"* and returns an answer in the form of a name. These sorts of answers are known as named entities and might be, for example, the name of a person, a country or a type of animal, or else perhaps a date, a time or an amount. More complex questions are also possible. The other types of questions that a user may ask are like those starting with 'wh" like who, why, when, which, where. "How, how much and how many" type questions are also possible.

If the computer system continues to treat the question as a bag of words, nothing is gained in using a question. Thereafter, there is a need of natural language processing techniques, such as syntactic and semantic parsing and named entity extraction to analyze the question type and to retrieve an accurate answer.

The next section discusses a brief history of Question answering systems.

2.3.1 BRIEF HISTORY OF QUESTION ANSWERING SYSTEM

BASEBALL[45,47] and LUNAR[45,47] were two early Question Answering Systems (QA systems) that have been introduces in the literature. BASEBALL answered questions about the US baseball league over a period of one year. LUNAR in turn, answered questions about the geological analysis of rocks returned by the Apollo moon

missions. Both QA systems were very effective in their chosen domains. In fact, LUNAR was demonstrated at a lunar science convention in 1971 and it was able to answer 90% of the questions in its domain posed by people untrained on the system. Further restricted-domain QA systems were developed in the following years. The common feature of all these systems is that they had a core database or knowledge system that was hand-written by experts of the chosen domain. The language abilities of BASEBALL and LUNAR used techniques that is similar to ELIZA[45,47] and DOCTOR[45,47], the first chatterbot programs.

SHRDLU[45,47] was a highly successful question-answering program developed by Terry Winograd in the late 60s and early 70s. It simulated the operation of a robot in a toy world (the "blocks world"), and it offered the possibility to ask the robot questions about the state of the world. Again, the strength of this system was the choice of a very specific domain and a very simple world with rules of physics that were easy to encode in a computer program.

In 1970s, knowledge bases were developed that targeted narrower domains of knowledge. The QA systems developed to interface with these expert systems produced more repeatable and valid responses to questions within an area of knowledge. These expert systems closely resembled modern QA systems except in their internal architecture. Expert systems rely heavily on expert-constructed and organized knowledge bases, whereas many modern QA systems rely on statistical processing of a large, unstructured, natural language text corpus.

The comprehensive theories in computational linguistics were developed in 1970 and 1980, which led to the development of ambitious projects in question answering. One example of such a system was the Unix Consultant (UC), developed by Robert Wilensky at U.C. Berkeley [45,47] in the late 1980s. The system answered questions pertaining to the UNIX operating system. It had a comprehensive hand-crafted knowledge base of its domain, and it aimed at phrasing the answer to accommodate various types of users.

A text-understanding system that operated on the domain of tourism information in a German city was developed. The systems developed in the UC and LILOG projects [45,47] never went past the stage of simple demonstrations, but they helped the development of theories on computational linguistics and reasoning. Recently, specialized natural language QA systems have been developed, such as EAGLi [45,47] for health and life scientists.

2.3.2 THE ANATOMY OF QA SYSTEMS

When a user asks a question, the first task of utmost importance is to catch the inflection of the words and to understand the need of the user that is a difficult task to perform by a machine. However, several approaches on how to design such a system have been suggested and implemented in the literature related to design of such systems. In this section, the general architecture of a Question Answering System is presented, and a detailed description of each of its module is given as follows:

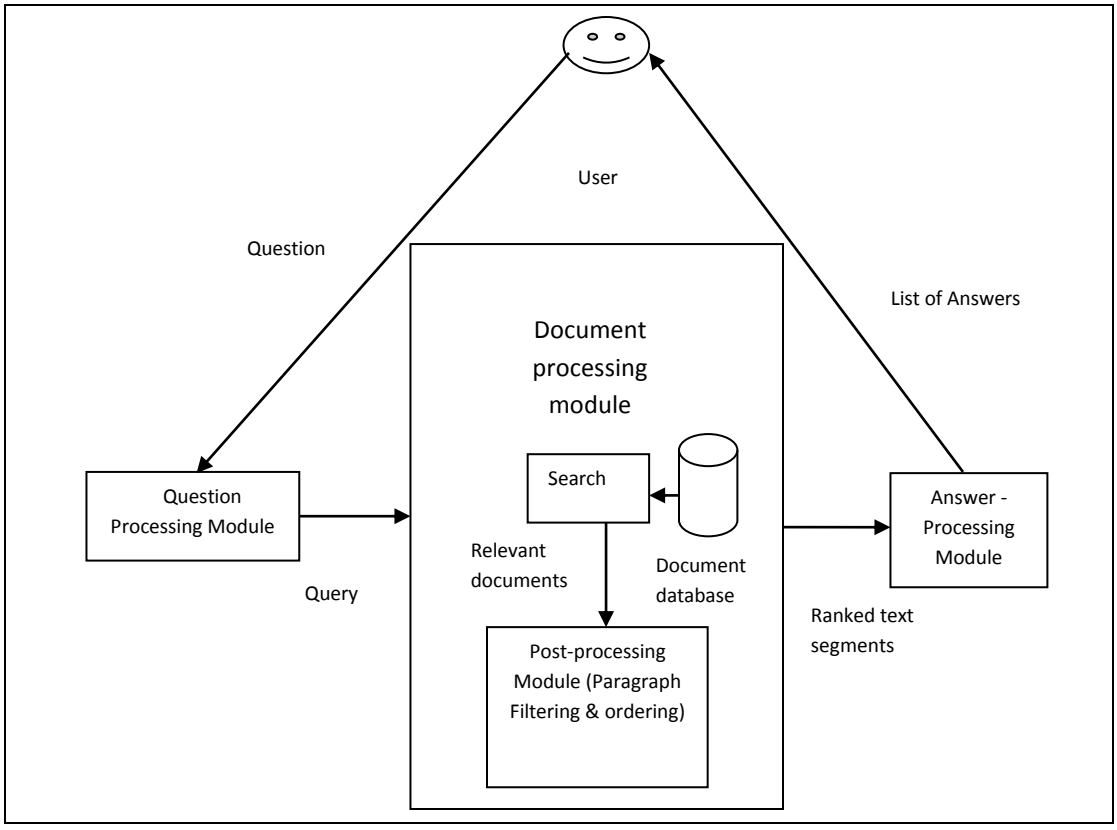


Fig. 2.1 Prototype of question answering system

Question answering system consists of three distinct modules: question processing, document processing, and answer processing. An illustration of this system architecture is given in Fig. 2.1.

- **Question processing:** Question processing deals with the question representation, identification of question type, finding expected answer type and the keywords extraction. The keywords that have been extracted are then used to fetch relevant documents. The question type is used to identify the expected answer type for finding a correct answer.

- **Document processing**: Document processing typically includes document retrieval, and passage identification. For this purpose, the approach of Keyword based matching coupled with the technique of keyword expansion is used that typically involves taking the keywords extracted in the question processing stage and looking them up in a thesaurus, or other resource, and adding similar search terms in order to fetch the relevant documents. A term such as "fight" might be expanded to "quarrel" for instance. Here, the expanded query is passed to a standard search engine (e.g. Google) and the documents with higher ranks are retrieved as a result. In passage retrieval, within each document the paragraph or section containing the possible answer is identified.

- **Answer processing**: This module consists of candidate answer identification, answer ranking, and answer formulation. Identifying the candidate answers means taking the results from the identified passages and further processing it. For this purpose, parsing of complete passage is committed. This results in a set of candidate answers that are then ranked according to a ranking algorithm or set of heuristics. Answer formulation is in most cases skipped completely and the answer is presented as it was found in the document. Answer Ranking is a major step in case if the answer extraction results in more than one answer. These answers are ranked based on relevance with those with higher rank appearing at the beginning of the list.

2.4 QUESTION ANSWERING SYSTEMS

A detailed discussion on the question answering systems is given in this section.

2.4.1 ANSWERBUS QA SYSTEM

AnswerBus [28] is an open-domain question answering system based on sentence level Web information retrieval. It accepts user's questions in six languages namely English, German, French, Spanish, Italian and Portuguese and provides answers in English. To respond to user's questions, five search engines and directories are used to retrieve relevant Web pages. The sentences that are determined to contain answers are then extracted from these Web pages. The working process of AnswerBus has been described in Fig. 2.2. To determine the language in which the user's question is posed, a language recognition module is used. If the language of the question is other than English, then the question is send to BabelFish [69], the translation tool of Alta Vista.

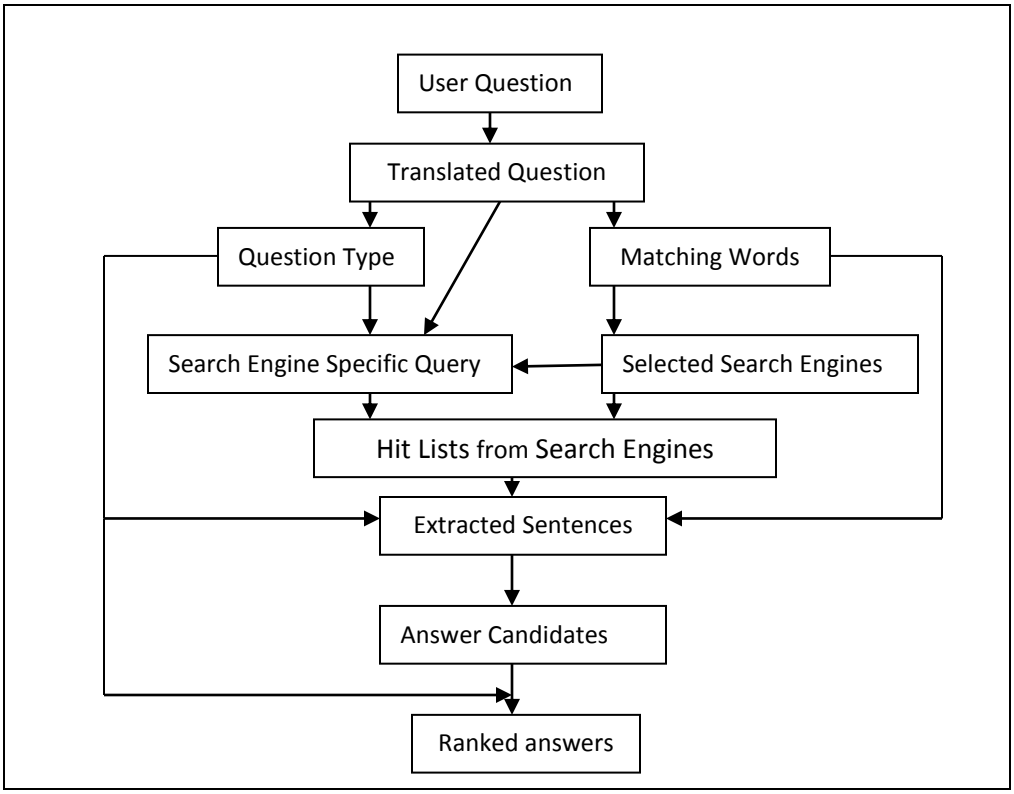


Fig. 2.2 Working process of AnswerBus

The rest of the process is completed into four steps:

i. Selection of two or three search engines among five for question answering and conversion of question into search engine specific queries.
ii. Retrieve the documents appearing at the top as the result of providing the queries to the selected search engines.
iii. Extraction of sentences from the documents that is likely to contain answers.
iv. Ranking the answers and return the choices at the top with the contextual URL links to the user.

2.4.1.1 RELEVANT DOCUMENT RETRIEVAL

AnswerBus aims to retrieve enough relevant documents from search engines within a response time that is acceptable to users. The main tasks involved in relevant document retrieval are as follows:

- Search engine selection: For answering a specific question, AnswerBus chooses to use two or more specific search engines among the five. For this, it collects 2000 sample questions and sends the queries to all of the five search engines. The answers are then recorded. All the words used in the queries are indexed. For example, for query q1, Google returns 8 answers, AltaVista returns 4 answers and Yahoo returns 7 answers. For query q2, Google returned 6 answers, AltaVista returned 6 answers and Yahoo returned 5 answers. For query q1 and q2, AnswerBus returns the results by merging the results of Google and Yahoo.
- Search engine specific query formation: These queries refer to the queries formed from a user's natural language question that are given to particular search engines and produce optimal search outcomes. Optimal means the best outcomes in terms of both recall of documents and time to retrieve the documents for a QA system. If a question like "How tall is Mount Everest", is sent to the search engine like Google, the results given as the result are more likely to be irrelevant to the user, hence lower precision and longer will be used to retrieve and process the documents. Some approaches use query expansion like ORing the synonyms,

however this leads to increase in the complexity and search response time will prolong. So, here focus has been laid on generating one simple query instead of expanded one. Here, several approaches are combined to form queries including functional word deletion (prepositions, conjunctions, interjections etc. and others like “kind of” are treated as functional words), deletion of frequently used words and word form modification etc.

2.4.1.2 CANDIDATE ANSWER EXTRACTION

At this stage, AnswerBus downloads and processes the documents referred at the top of search results returned by different search engines. It uses sentence segmentation tool that deletes HTML tags, excludes non-contextual content, and regards some special HTML tags as sentence boundary indications to parse the document into sentences and then separate sentences that are answer candidates by the process of word matching. Each sentence gets a primary score on the basis of matching words in the sentence. The sentences with a score of “0” are generally discarded.

2.4.1.3 ANSWER RANKING

AnswerBus uses several techniques to refine the primary score assigned to each sentence which then decides the overall rank of a sentence. The techniques used for answer ranking are follows.

- Question type and QA specific dictionary: The question type is classified on the basis of the type of answer the user is expecting like “who is……” is assigned a “person/organization” type. Along with this, some other parameters are also used like “how far” is most likely will be a “mile, km, light year” and “how close” is most likely will be a “inch, cm”. The system uses a QA specific dictionary, a database containing this kind of information about the relationship of words between questions and answers. This piece of information is used to judge whether a sentence can be an answer to a question.

- Dynamic basic named entities extraction: This involves pre-tagging of the corpus or knowledge base on which a QA is based. For a question “how much money”, a sentence with the entity of “CURRENCY” receives higher rank.
- Coreference resolution: A sentence may contain words, such as “he”, “they”, that coreference to other objects described in the document. AnswerBus solves the coreferences in the adjacent sentences. After this, the later sentence receives part of score from its previous sentence.
- Hit position and search engine confidence: The rank of a sentence can also be related to the position that its source document is located in the hit list returned by a search engine. A sentence extracted from the first hit receives the highest score associated with hit positions and the score decreases as the position moves down.
- Redundancy: Different search engines can retrieve document for a question; one or multiple documents may contain same or very similar sentences. This leads to candidate answer sentence redundancy. For checking this redundancy, the system compares highly ranked sentences against one another. However, spaces, punctuation marks, special characters and the words with high frequency are not considered while making comparisons among the sentences.

2.4.2 AskMSR QUESTION ANSWERING SYSTEM

In [29,30,31], a question answering system is described that is designed to make use of the tremendous amount of data available on WWW. Unlike most of the QA systems that use linguistic resources, here the focus is on the redundancy available in large corpora as an important resource. This redundancy is used to simplify the query rewrites for mining answers from returned snippets. Also, some strategies are explored in this work for determining the incorrect answers given by the system.

2.4.2.1 SYSTEM OVERVIEW

A flow diagram of the system is shown in Fig. 2.3. There are four main components contained in the system discussed as follows:

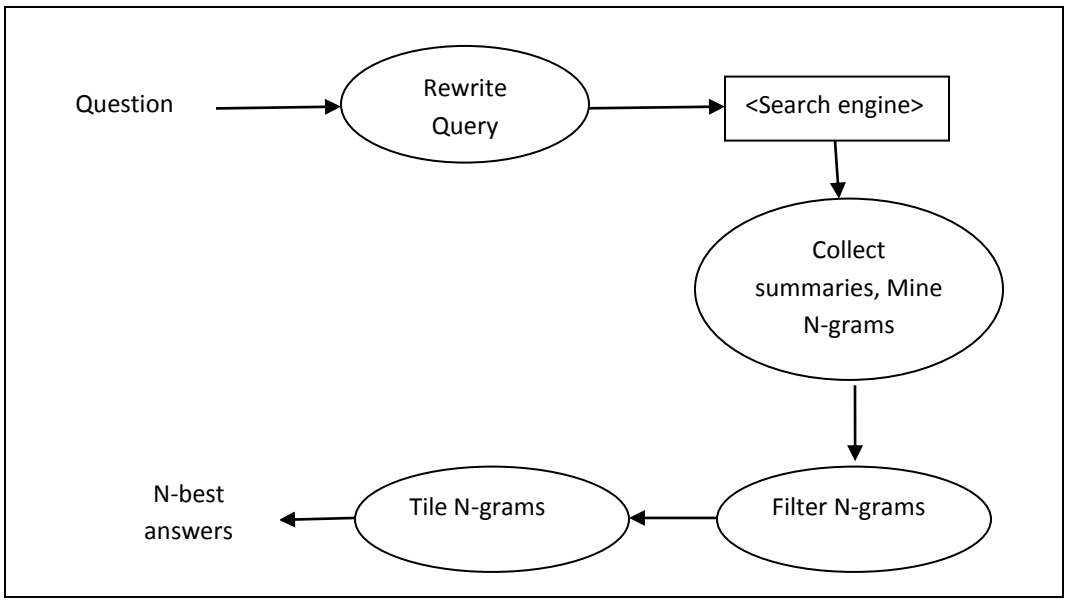


Fig. 2.3 AskMSR System architecture

- Rewrite Query: The first component takes a question as input; and generates a number of rewrite strings. These strings are likely substrings of declarative answers to the question. Let us take a question "When was Abraham Lincoln born?" For this question, it is known that a likely answer formulation takes the form "Abraham Lincoln was born on <DATE>". Therefore, a search can be performed in the collection of documents, in search for such a pattern. At first, it is important to classify the question into one of seven categories and each of which then needs to be mapped to a particular set of rewrite rules. As the output of this module, a set of 3-tuples is formed. The set takes the form [string, L/R/-, weight], where "string" is the query for which search is being performed, "L/R/-" indicates where we expect to find the answer with respect to the query string i.e. the position of the string (to the left, right or anywhere) and "weight" reflects the weight assigned to the answers found with this particular query.

- The answers are weighted according to the weights assigned to the rewrite query. The query may be a high precision or a low precision query. The idea behind using a weight is that answers found using a high precision query are more likely to be correct than those found using a lower precision query. Like the query "Abraham Lincoln was born on" is of high precision than the query "Abraham"

AND “Lincoln” AND “born”. So, the answers associated with the first query are given higher weightage than those associated with the second one. The rewrite rules and associated weights are created manually for the current system. The query rewrites generated by our system are simple string-based manipulations.

Consider how a verb “is” is moved while making a query rewrite for the question “Where is the Louvre Museum located?” The rewrite is “The Louvre Museum is located in”. This determination for where to move a verb can be done by analyzing the sentence syntactically. Given a query such as “Where is w1 w2 … wn”, where each of the wi is a word, for each possible move of the verb, a rewrite is generated, like “w1 is w2 … wn”, “w1 w2 is … wn”, etc.

For each query, a final rewrite which is a backoff to a simple ANDing of the non-stop words in the query is generated. Let’s take an example for the query “Who created the character of Scrooge?”, the rewrites are as follows:

a). LEFT_5_”created +the character +of Scrooge”

b). RIGHT_5_”+the character +of Scrooge +was created +by”

c). AND_2_”created” AND “+the character” AND “+of Scrooge”

d). AND_1_”created” AND “character” AND “Scrooge”

In this approach, the stop words like “in” and “the” need to be matched, like in the above example. So, here the stop words are important indicators of likely answers. The next step is to fire the query rewrites as search engine queries and the snippets provided by the search engine as the result are used as the page summaries are then collected and analyzed.

- Mine N-Grams: The next step is to mine n-grams from the page summaries returned by the search engine. For reasons of efficiency, only the returned summaries and not the full-text of the corresponding web page are used. The summaries returned by the search engine contain the query terms, usually with a few words of surrounding context. The summary text is then processed. The only

strings that are extracted are either to the left or right of the query string. The extraction is as specified in the rewrite triple. 1-, 2-, and 3-grams are extracted from the summaries. Then the extracted N-gram is scored according the weight of the query rewrite that retrieved it. The summaries that contain the particular n-gram are searched and these scores are summed across all those summaries. This is the opposite of the usual inverse document frequency component of document. The usual term frequency (tf) component used in ranking schemes i.e. the frequency of occurrence of n-gram within a summary is not used. Thus, the final score for an n-gram is based on the rewrite rules that generated it and the number of unique summaries in which it occurred. For the query "Who created the character of Scrooge? discussed above, the following are the top-ranked n-grams:

a). Dickens 117

b). Christmas Carol 78Charles Dickens 75

c). Disney 72

d). Carl Banks 54

e). A Christmas 41

f). uncle 31

- Filter/Reweight N-Grams: The next step is to filter the n-grams. After the step of mining n-grams, it is found that how well each candidate matches the expected answer-type. This is the criterion used for filtering the n-grams. The system uses the filters for this purpose. The filtering takes place in the following manner:

  First, the query is analyzed to find out what we expect as an answer in response to the query. For this the query is assigned one of seven question types, such as who, why, or how-many i.e. one question type is assigned to the query. Since, there are a number of filters for different question types with specific features, so on the basis of the query type that has been assigned, the system determines the filter to apply to the set of potential answers found earlier. The answers are then rescored

according to the presence of desired features. The system uses a collection of approximately 15 filters. After the application of filters to a pool of candidate answers, the score of the string is then readjusted. The score assigned to a potential answer is boosted using the filters. Some answer candidates may be removed in some cases like in case when the set of correct answers was determined to be a closed set (e.g. “Which continent….?”) or definable by a set of closed properties (e.g. “How many…?”).

- Tile N-Grams: Next step is to tile the n-grams that have been filtered for a question by applying an answer tiling algorithm. The algorithm works on the objective of merging similar answers and then assembles longer answers out of small answer fragments. Tiling forms longer n-grams from sequences of overlapping shorter n-grams.

  As an example, two n-grams filtered say "A B C" and "B C D" are tiled into "A B C D." The n-grams selected for the purpose are the top-scoring candidates. Also, a cut-off is decided for picking up the subsequent candidates that satisfy the criterion. These candidates are checked to see if they can be tiled with the current candidate answer. If so, longer tiled n-gram is used to replace the higher scoring candidate and the lower scoring candidate is removed. The algorithm proceeds until no further n-grams can be tiled.

  For the Scrooge query are, the top-ranked n-grams after tiling are as follows:

  a). Charles Dickens 117

  b). A Christmas Carol 78

  c). Walt Disney’s uncle 72

  d). Carl Banks 54

  e). uncle 31

2.4.3 QUESTION ANSWERING BASED ON SEMANTIC GRAPHS

The system discussed in [32] makes use of semantic graphs. Along with providing answers to questions in natural language, the system also provides explanations for the answers via a visual representation of documents and uses subject-verb-object triplets and their summaries. The system is not restricted to a specific domain; however it restricts the grammatical structure of the question to a predefined template. These facts are then used to retrieve answers. The triplets are then enhanced and on the basis of the enhanced triplets the semantic graph for the document is constructed. To generate the document summary, the semantic description of the document and the extracted facts is used.

2.4.3.1 SYSTEM OVERVIEW

The system combines three major functionalities: question answering, summarization and document visualization. The user poses a question to the system and the system responds with the answers. Each answer is linked to the sentence that contain it and the documents that contain these sentences. Also, a document overview is provided by the following:

- document semantic graph
- list of subject-verb-object facts and
- document summary of variable length

The description of the main components of the system is given in the next section.

2.4.3.2 QUESTION ANSWERING

The system at first extracts facts from the text and then the facts are represented using subject-verb-object triplets. The triplets are then indexed and a search is performed for any of their elements that are left undetermined. Like, in Fig. 2.4, answer for the question "where do tigers live" is identified using triplets. In general, a tree structure is used to organize a query in which the leaves of the tree are triplets with one or more elements undetermined. A database is prepared which stores the triplets constructed as explained above and then a search is performed for the query triplet formed as the result of analysis

of the question. For the example above, let the query triplet are (tigers, live, ?) and (tigers, inhabit, ?) and the stored triplets are (tiger, live, Sumatra) and (Indonesia, protect, tiger). Wordnet is used to find out the synonyms. As the answer to the question, "Sumatra" is returned.

- **Triplets:** The information contained in a sentence is represented using the triplet. It contains the subject, the verb and the object of the sentence, it represents. This is the basic unit of data on which the question answering system is built. This information is indexed for the fast retrieval. Thus, an index is maintained that consists of the triplets formed. The system uses a query structure that is formed for easy to reuse and extend.

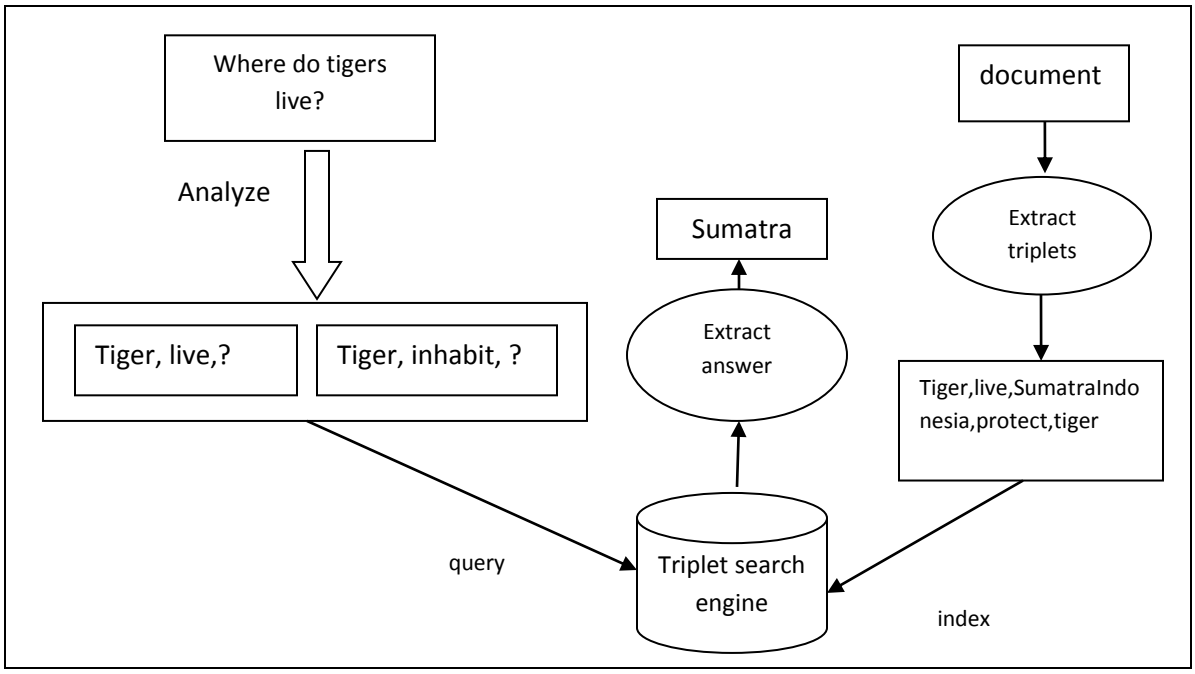


Fig. 2.4 Process of answering a question

- **Question Analysis:** The major task of this module is to determine the type of question and to form a query for answering questions. The system supports various question types like yes/no, list type questions, those starting with why, how much, where, when etc. The question is parsed using OpenNLP parser to obtain a parse tree.

- **Answer Generation:** The result of a query is a set of triplets as follows.

    i. If the question is of type yes/no, then the resulting set of triplets is split into two groups: Triplets in which the polarity of the verb matches the polarity of the verb in the question: the group that supports answer "yes" and the Triplets in which the polarity of the two doesn't match: the group that supports answer "no".

    ii. If the question is a list question, a quantity question or a location question, then the answer is the collection of items.

    iii. For a reason question or a time question, the system gives no clear answer. Instead the sentences that contain the triplets returned by the query are given as answer.

2.4.3.3 SEMANTIC GRAPH

Semantic graph provides an overview of the content contained in the document. The input document is processed and after the processing, it is then passed through following series of sequential operations explained below composing a pipeline, to obtain the graph:

- Text preprocessing: The original document is split into sentences.

- Named Entity Extraction: It refers to the names of people, locations and organizations, for retrieving semantic information from the input text. For this a toolkit for NLP named *General Architecture for Text Engineering* is used. Like, for the people, the information about their gender is stored. Similarly, for locations, the names of cities and countries are stored separately. This enables co reference resolution that implies identifying terms refering to the same entity. Also, entity matching is performed like in case of inclusion of one surface form into another ("Anna Maria" is same as "Anna Maria Smith"), when one surface form is the abbreviation

of the other ("NLP" stands for "Natural language processing") or when a combination of both of these ("A. Smith" and "Anna Smith") occur.

- Triple extraction: For triplet extraction, each sentence is taken into account without considering the surrounding text. For this, a Penn Treebank parser is applied and also the statistical Stanford Parser and the OpenNLP parser is employed for question answering. For the triplet extraction, the pure syntactic analysis of the sentences needs to be performed.

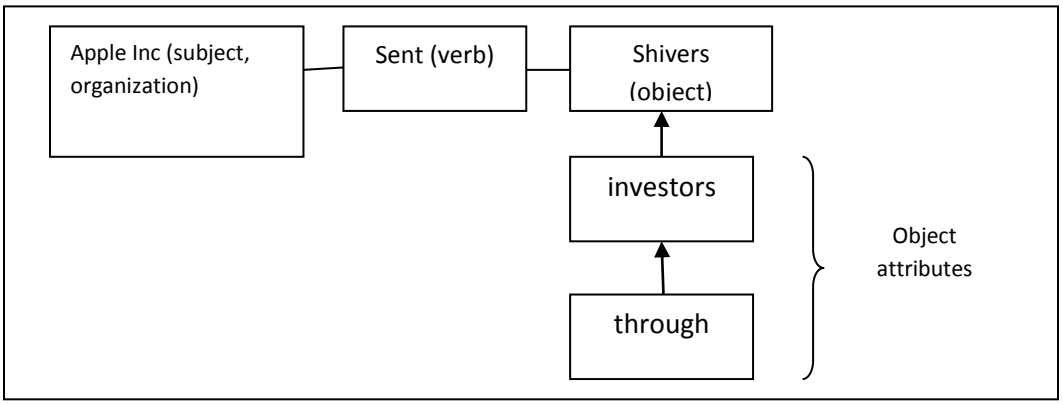


Fig. 2.5 A triplet extracted from sample sentences

Fig. 2.5 shows an example of a triplet extracted. It shows the subject, verb, and object and also the object attributes.

- Triplet Enhancement and Semantic Graph Generation: Triplets are enhanced by first resolving anaphors for a subset of pronouns: {I, he, she, it, they}, their objective, reflexive and passive form and the relative pronoun "who". For this, the triplets are linked to their corresponding co referenced named entity. Also, the subject like "Apple" would be linked to co referenced named entity "Apple Inc"). Further all the pronouns in the document are searched for replacement. The triplet elements that share the same meaning are merged using Wordnet.

- Finally a directed Semantic Graph is obtained in which the arrows move from the subject node to the object node and the connecting link is a verb. Fig. 2.6 presents a semantic sub-graph of a text portion.

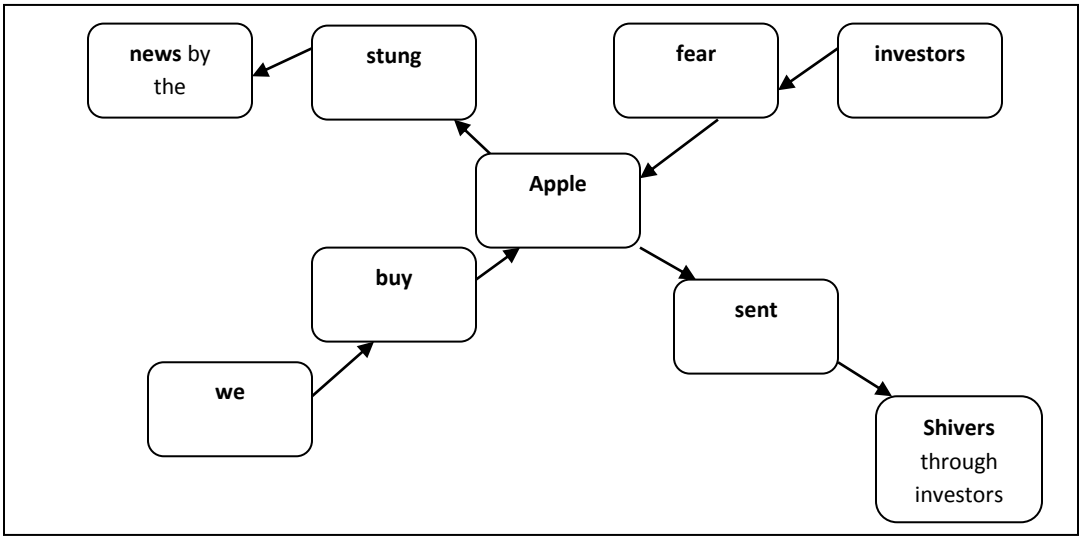


Fig. 2.6 An example semantic sub-graph

2.4.3.4 DOCUMENT SUMMARIES

The summary of the document consists of sentences from the text, with the sentences in the same order as in the original text. The summarization technique used involves training a linear SVM classifier to determine those triplets that are useful for extracting sentences for summarization by considering some features. As an input to the SVM classifier, the initial document and the semantic graph are used and as the output, it provides a score to each sentence termed as SVM score as the output. The sentences are then ordered based on these scores in descending order.

The work done so far on QA systems [28,29,30,31,32] uses general web pages as their source of information. But it has been observed that for question answering, there is a need of the sources comprising of topical information. The blogosphere [61,113] has been found comprising of such kind of information. A brief introduction to blogs and blogosphere is given in the sections below.

2.5 BLOGS: AN INTRODUCTION

An introduction to blogs and the related content is given in this section.

2.5.1 BLOG

A blog [33] is an online journal which is generally composed of media-rich articles called "posts". Blogs are also referred to as "weblogs". A weblog [34] is defined as a web document which doesn't require any external editing. It is composed of posts which are updated on periodic basis and are presented in reverse order of their occurrence in time i.e. in reverse chronological order. A blog may consist of a number of hyperlinks to other online sources. Blogs can be treated as a space to write about oneself in form of personal diaries, as columns for technical advice, for chat, political commentary, educational purposes, opinion sharing, discussions etc. Blogs can be published on any matter that can be thought of. Also, any person can easily publish a blog without expenses. It has been found that the majority of blogs are written in English.

In general, a blog can be thought of as a website with a number of pages. A person can visit a blog page and can read a blog page in the same way as (s)he can read a web page. Some of the popular blog sites that facilitate a user to write and publish his blog are technorati, wordpress, blogger, livejournal, typepad, travelpod etc.

Since, a blog page is written by either a single person or a small group, so a consistent style of writing is used across the whole text. Blogs can be classified according to their purpose: *personal blogs* (documenting one's life), *issues* blogs (expressing opinions, writing comments, debating current events) and *topical blogs* (serve as community forums allowing users share their ideas with each other). Blogs have a wide coverage and thus cover most of the topics like education, entertainment, sports, music, health, business, agriculture etc. It has been found that blogs written on a topic are likely to contain details on the topic being discussed. So, these contain the relevant in depth information on the topic of interest. It shows that the information on blogs is very much relevant as compared to that present on remaining part of WWW. Blogs Pages are the richest source of information where people express their opinion on various topics and

situations. Blog written by one may be shared by others. Blogs are search friendly and attract more traffic for its fresh and dynamic contents. Blog posts are written by experienced person called Blogger [12] on various topics. The content on the Blog pages is likely to be related to the topic on which the Blog is being written.

2.5.2 COMPONENTS OF A BLOG

In terms of content organization, a typical blog is composed of three main components [35], depicted in Fig 2.7:

- One or more units of content- Each unit of content is said to be a blog post written by the blogger using HTML. A blog usually covers a single topic and includes one or more comments added by the readers of the blog. A blog is assigned a permanent URL (known as a permalink) for its unique identification.
- A syndicated XML feed: Since, the blogs are updated on periodic basis, there is an XML feed added in the blog. Whenever there is an update in the blog content, the digital content that is frequently updated is published and organized in the feed. The feed gives the ability to the readers to subscribe to the blogs. For subscription there are some client applications that run, known as aggregators, feed readers or news readers. When a reader subscribes to RSS Feed, the recent blog post is delivered to his mailbox automatically. The updates appear just like the emails appearing in the mailbox, latest content on the top, with the headline and the first few lines of the post. There are two XML standards used in common for blog feeds, namely Really Simple Syndication (RSS) [100] and Atom [100]. RSS is in commonly used. In addition, some blogs provide feeds for also retrieving comments.

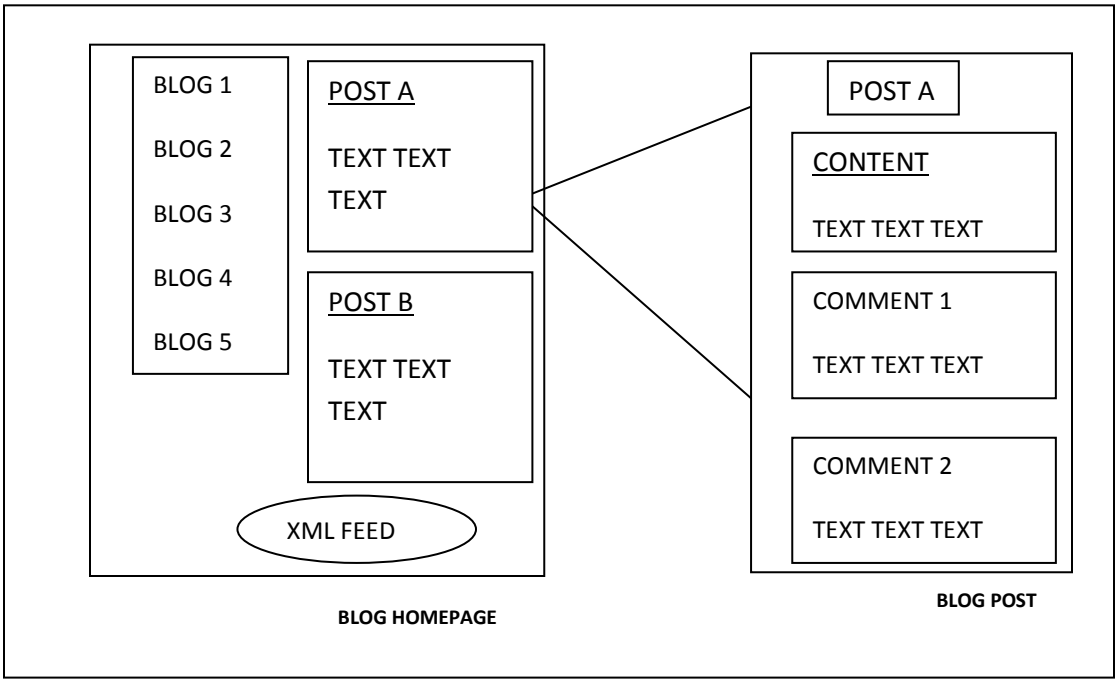


Fig. 2.7 A typical Blog

- An HTML homepage: A blog can be considered as an HTML page with the posts organized in a reverse chronological order. The content on the blog content is dynamic because it can be expanded, modified, or removed at any time. The content on the blog is not necessarily text. Besides text, a blog may consist of specific types of data. A blog may consist of audio (podcasts), images (photoblogs), video (vlogs), etc. A user may be interested in publishing very short content (e.g., a 140-character long post) which focuses on his up-to-the-minute thoughts. These are known as Microblogs (eg. Twitter). A blog page may consist of a number of comments given by the readers of the blog. There is a list of "friend" blogs i.e. those blogs that are somehow related to the current blog or in which the blogger is interested. This list is known as a blogroll. The blogrolls facilitate the user with the blogs that cater to his topic and thus save his time and effort.

2.5.3 THE NETWORKED STRUCTURE OF THE BLOGOSPHERE

The universe of blogs is conventionally referred to as the blogosphere [34,40]. The structure of the blogosphere is networked as it relies on hyperlinks and may contain links to other blogs. The bloggers maintain a blogroll on their website containing a list of blogs that the blogger frequently read or admire. The list also includes clickable URLs to link the blog pages that are added in the list. Blogrolls is one of the important means of finding out bloggers interest and preferences within the blogosphere. Also, the posts contained in the blogs written by the bloggers may point to other blogs. A blog may contain comments to the blog posts of others. Posts that contain comments on other's posts are a way of information exchange in the blogosphere. In this way a chain on bloggers commenting on blogs is maintained. The links and page views are very important for blogosphere. A blogger is always keen to have wide readership. Linking on another weblog is the most reliable way to have readership. The basis of this is the hypertext. Finding a link to another blog, the reader reads the blog that is being linked by the current blog. Because of this, the reader may find the second blog more useful and may become regular reader of the second blog.

Also, a blogger is interested in knowing who links to him i.e. all the incoming links. Also, when the reader finds important source of information on another blogs, they are likely to credit the sources in their comments and may provide links to them also for other readers. So, blogs are linked through the use of hyperlinks such that they form a network. Each blog is treated as a node or vertex and each hyperlink is treated as an edge or an arc connecting blogs. The number of incoming and outgoing edges is called indegree and outdegree respectively.

2.5.4 WHY DO PEOPLE BLOG?

The bloggers [35] have been classified into two categories: specialists and generalists. Specialists are those bloggers who write their blogs on some specific topics like sports, gaming, politics, medicine or technology etc. These bloggers receive a large number of visits and comments also on their blogs. The other category is generalists, the people who

write in the blogs as if they are writing in their personal diaries-about themselves and their activities carried out by them on daily basis. These write for a small number of people. Their blogs are generally read by a small number of people. The studies [41] have shown that half of the blogs available online are written by males and half are written by females. So, there is an equal distribution between the both. Also, only 7% of bloggers are the people over 50 years in age, about 50%bloggers are those aged 21-35 years and 20% are aged 20 or under. So, most of the bloggers are teenagers. The people may write blogs individually or in a group. Some people write their blogs individually while some of them prefer to write in groups. Later type is referred to as group blogging. One example where this type of blogging is used is corporate blogging. Like in an organization, employees can communicate with each other in an effective way through these types of blogs. Because of higher link popularity and longer post lengths in group blogs, they are likely to be regarded as of high quality.

2.5.5 THE BLOGOSPHERE

The collection of blogs [35,40,41] on the web, has given a new direction not only to the way of information consumption but also to the way the information is produced in an intelligent manner. There is no external entity to manage the content on the blogosphere but they are the independent bloggers who manage the content. The major feature is that the blogs enable interaction among the people i.e. among those who write and those who read the blogs written by others. Also, the facility to provide reviews to the blog's content in form of comments enables interaction and effective communication. Also, the bloggers can themselves be the readers of other blogs, so the role of information producer and consumer may be performed by both.

The blog linking is another way to communicate in the Blogosphere. There are following three categories of inter-blog links that have been found in the blogosphere.

- Blogroll links: These links are generally placed on the blog's homepage and are the links to the related blogs or the friend blogs. This relationship is presented using the concept of Blogroll links.

- Citation links: These links are just like the hyperlinks present on the general web pages. These show the relationship of the current blog or blog post to the blog page or post with which it connects.

- Linkbacks: These types of links are also called trackbacks. It is a way used to find the people who are linking to your post i.e. to find the incoming links.

Due to these types of links, there is huge interaction among the bloggers in the blogosphere. Thus, the blogosphere can be thought of as a network of interconnected bloggers. The blogs that are written by experienced and authoritative bloggers are read and followed by large number of people.

2.5.6 BLOG SEARCH

A detailed discussion on blog search is given in this section.

2.5.6.1 BLOG DIRECTORIES

With the increase in the size of Blogosphere, there is a need of tools to find the blogs to which the reader can subscribe. One way to find the blogs which are related somehow to the current blog of the blogger is a list of blogs present on the right hand side of the blog page. The list helps the readers of a blog to find the other interesting blogs. But the process is not useful for the searching of the blogs [43] that are not mentioned in the blogs being read by the reader. Blog directories such as Blogflux[35, 70] and Topblogarea[35, 71] provide a search area in which the user can write for the blog to search. A snapshot of Blogflux is shown in fig. 2.8.

The interface of Blogflux provides a space to the user to enter the topic of the blog he/she is looking for. Also, the facility of blog tagging is provided in which the user can search for a particular category of blog.

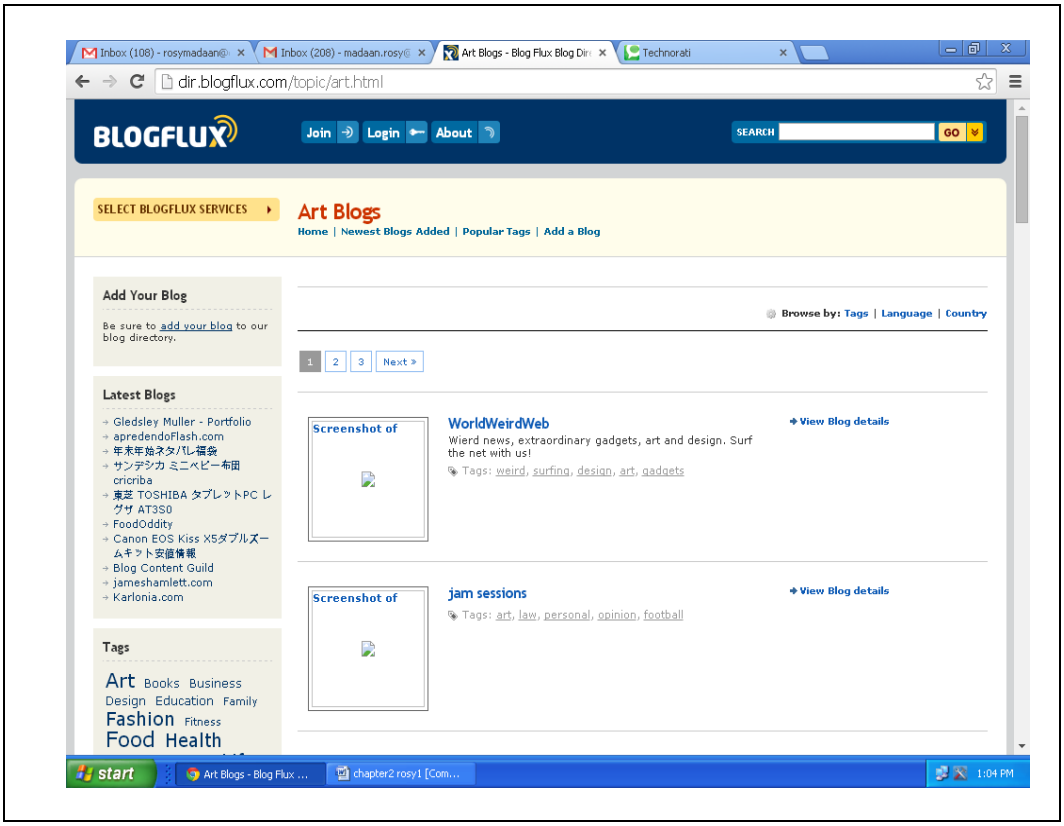


Fig. 2.8 An example Blog directory: Blogflux.com

The introduction to the blog search engines is given in the next sections.

2.5.6.2 EXISTING BLOG SEARCH ENGINES

The emergence, growth and popularity of the blogosphere, lead to the introduction of the blog search engines. These special types of search engines allow the user to enter a query for blog post search. As a response to the query, a list of relevant blog posts is returned to the user. Two major issues associated with most of the blog search engines are freshness and recency. Therefore, the blog search engine needs to provide the user with the up-to-date information with the most recent blog posts on the top.

- Technorati [35,88] is one of the most definitive blog search engines, since 2002. Technorati (see Fig. 2.9) supports the searching of blogs or their individual posts by providing two choices to the user in form of slider switch interface. It has also been found that the Google search engine give more importance to the fresh results than the Technorati for initial relevance ranking[63,64,65]. In contrast,

popularity is a major factor used to rank [66,67,68] the blogs in Technorati and gives more importance to the deemed authority. Technorati is a real-time search engine dedicated to the blogosphere. Its database consists of blogs and it only searches through its database to find what the user is looking for. It is very easy to search for blogs on Technorati. Technorati's home page provides main search query bar to write query. Technorati provides more advanced search options to the user i.e. the user can search for more search parameters.

There are a number of tags that it provides to the user. These are basically subjects or topics that bloggers assign to their blogs. The tags shown on the Technorati Tag page are organized in alphabetical order.

Technorati has a blog directory known as Technorati Blog Finder organized by topic. There are number of categories in the directory and the user can browse through for the blog of interest. It also contains the most recently added blogs. Technorati has a popular list that shows people searching and there is an option of *What's Popular* with main categories like News, books, movies etc.

Another type of blog search engine comes into picture that monitors the blogosphere from a temporal perspective. As blogosphere is a repository of huge subjective content and deals with full discussions, mining for insights and opinions about products or a company can be very useful. Technorati can also be used to track various trends and topics on the Web on day to day basis. Therefore, Technorati is recommended as a great way to search the blogosphere.

Fig. 2.9 Searching for blogs in Tecnorati.com

- In September 2005, Google [36] introduced the search of blog content on its website. Google introduced its Blog Search to expand their vertical search. In this, Google focuses on a specific sphere, namely the blogosphere. Because of good reputation of Google and its large index of web search, it soon became the main competitor of Technorati.

  .

2.5.7 SIGNIFICANCE OF BLOGS

Since, Blogs [37] provide an opportunity for sharing knowledge, sharing ideas, debate etc., which attract a large number of audiences. These facilitate open discussions, thus making blogs an ideal place for communication and large distant discussions on new and emerging topics and trends. Blogs also bring up the communities and allow collaborative sharing and recommendations. Blogs are growing at an exponential rate and occupy a large amount of space on the web. Blogs are easily approachable and they exist in close association to other blogs. Also, the blogs serve as an educational tool for the building, sharing and reflection of knowledge. The value of blogs is increasing day to day in the

field of education. The content on the blogs can be shared among individuals by using RSS feeds. Blogs allow students, faculty, staff and the other people associated to have peer to peer interaction and the students can learn a lot from one another as from the teachers and/or books and can explore further.Thus, these provide an excellent mechanism for knowledge sharing and acquisition.

Blogs [38], if seen from a teacher's point of view, can be seen as a class notice board. It can serve as a discussion tool with the students. From the student's point of view, it can also be used as a learning tool. Blogs encourage students to write and also to read on a topic they wish to comment on. Blogs allow the students and teachers to continually search and filter for the posts. Also these allow the users to post ideas and information which engage higher order thinking skills. Since, commenting is an important feature of the blogs, so the comments can be treated as feedback and the students can use them to improve on their work. Also, like websites, the blogs allow the bloggers to embed the other Multimedia elements other than text like video, audio or flash movies. One can also attach word processing, spreadsheets and pdf files into a blog. Blogging helps to enhance the following skills [39]:

- Sharing — thoughts, concepts, experiences, knowledge
- Analyzing
- Reflecting — Critiquing, Writing, Questioning, Reacting
- Communication
- Record keeping — thoughts, concepts, and experiences
- Collaboration — with peers, people (experts, students) around the world

A component based search engine for blogs is given in the next section.

2.6 COMPONENT BASED SEARCH ENGINE FOR BLOGS

A component based architecture has been proposed in [44] for searching in blogs for the user's query. There is a major step involved in search engines termed as Information Extraction. A blog page consists of text and a lot of other kinds of information like links to advertisements, copyright notices, navigational links, links to archives and new

content, website menu etc. This all is regarded as *noise* in the blog page. Thus there is a need to extract the relevant information to improve the search accuracy. The blogs that are written by the blog authors contain useful and valuable information. Because of the presence of a large amount of noise in the blog page, the extraction of the useful content within the page is harmed. This is really a big problem that needs to be overcome for efficiency of blog search engine. The task of separating the noise from the rest of the blog's content is really a difficult task. The reason behind this is that the blogs are written by different authors in their way and style. Also, the template used may differ from blog to blog. Some author may also use design their own template as per their need and preference. Thus, the problem of separating the noise and data arises. If these aren't separated, then each of the word in the blog page is indexed, thus reducing the accuracy of the search results. In [44], instead of separating the content and other parts of a blog page, the authors have assigned a score to each of the component. By assigning a score, the content within the page is given a high score as compared to the text in the other parts of the page. As the result, the blog with the rich content appear at the top of the search results. Consider the snapshot of an article in a tourism blog as shown in Fig. 2.10.

The snapshot consists of five parts as explained below:

Part-1: This part contains the navigational links of the site.

Part-2: This part contains the archive links.

Part-3: This part contains the links to the latest contents.

Part-4: This part contains the title of the article.

Part-5: This part contains the main contents of the article.

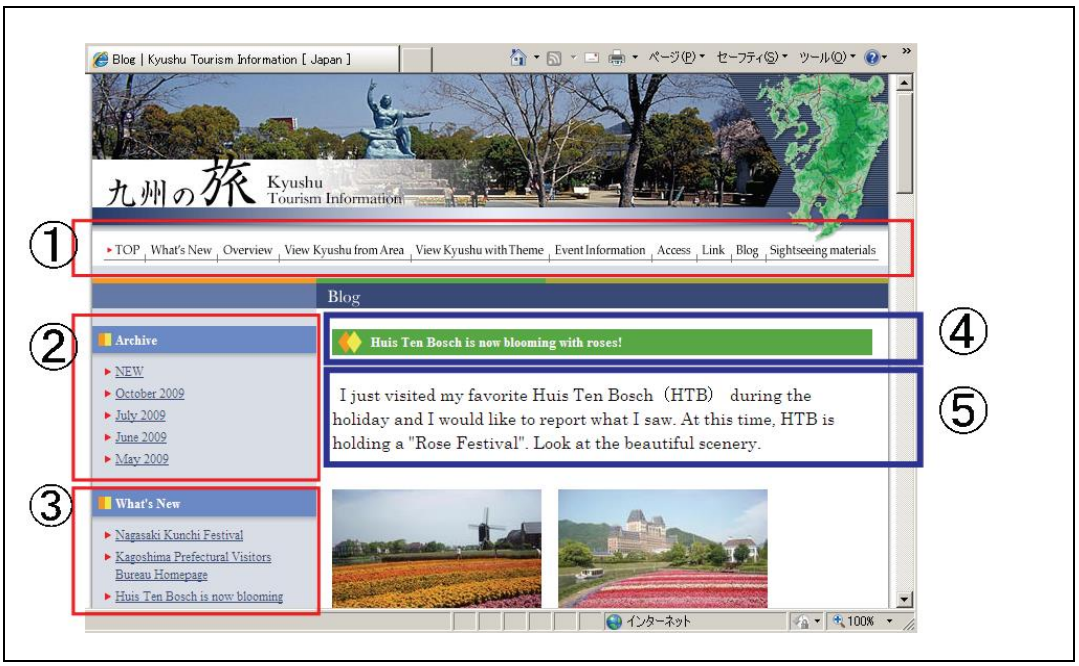


Fig. 2.10 A Tourism blog

The HTML tag tree of the Tourism blog is shown in Fig. 2.11.

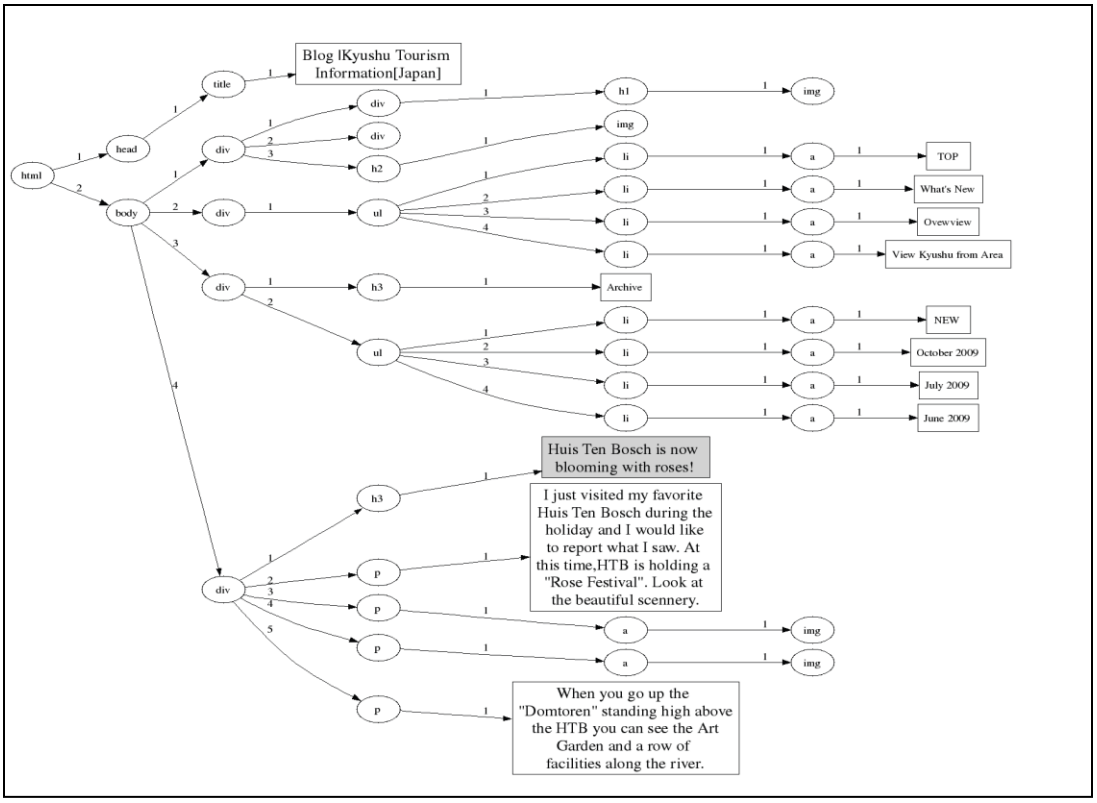


Fig. 2.11 HTML tag tree of the Tourism blog

The square boxes in the Fig. 2.11 show the text areas in the page. These are termed as "components". These components are then indexed for searching in the search engine. The index file of the search engine is shown in Fig. 2.12. In the component index shown in Fig. 2.12, the lines that start with @ symbol show the components that have been indexed. The components used in the index are explained as follows: "@1-1" and "@1-2" represents the 1st and 2nd text areas of the 1st HTML file. Also, the other lines of the index display the index keywords and their frequencies in the text area. Id of the HTML file starts with h: and the path from the root of the HTML tag tree to the component starts with p: as shown. It has been found that the linked text is usually relevant to the other web pages and not for the search engine. Thus, it is recommended to filter out the links under the anchor tag from the components. There may be some links under the anchor tag that may point inside the page. Therefore, eliminating these links doesn't lose any information from the page.

```
@1-1
1 h:1
1 p:/html[1]/head[1]/title[1]
1 Blog
1 Kyushu
1 Tourism
1 . . .
@1-2
1 h:1
1 p:/html[2]/body[4]/div[1]/div[1]/h3[1]/
1 Houis
1 Ten
2 Bosh
2 is
1 . . .
@1-3
1 h:1
1 p:/html[2]/body[4]/div[1]/div[2]/p[1]/
3 I
1 just
1 visit
1 . . .
...
```

Fig. 2.12 Component index of the Tourism blog

There are a number of methods that have been proposed to evaluate the importance of a word in a document. For finding the score of the entire document, the score of each of its components need to be found. Then the summation of the scores yields the score of the entire document. The formula used for the calculation is:

$$\text{Score } (C_i) = \text{NW } (C_i) * \text{depth } (C_i) \quad \text{.....eq. 2.1}$$

In eq. 2.1, NW (C_i) is the number of the distinct words present in the component C_i and *depth* (C_i) is the length of the path from the component to the HTML tree's root.

Finding that the content which is detailed exists in the web page deep inside whereas the content treated as noise like links to ads etc. exists at the shallow nodes. Therefore the formula uses the function *depth* for the calculation. By summing up the score assigned to the individual components, the score of the whole document is obtained.

2.7 ISSUES IN EXISTING APPROACHES OF QUESTION ANSWEING

As the result of the literature review on the existing Question Answering Systems, it has been found that the issues shown in Table 2.1 need to be addressed while designing an efficient Question Answering System for answering user's question belonging to different domains:

Table 2.1 Issues in existing approaches for question answering

S.NO.	APPROACH USED	ISSUES
1	ANSWERBUS QA SYSTEM	(i). In this system, the relevant web pages are retrieved from the five search engines and directories and which further are used to extract the sentences that contain answers but it is not specified that how many search engines are used for searching in case of a particular query.

		(ii). It is not clear from the available literature that for which type of questions the system works. (iii). The module of Question Analysis is not discussed in much detail and needs to be explored.
2	ASKMSR QA SYSTEM	(i). The answers are extracted and ranked on the basis of query rewrites but much details on how to write different kinds of user's query is not given. Also, the criterion for scoring the answer candidates on the basis of the query rewrites is not clear. (ii). The strategy for finding the inaccurate answers is not given but stated. (iii). Much details on how to mine, filter and tile n-grams is not given.
3	QA BASED ON SEMANTIC GRAPH	(i). One major drawback of the system based on Semantic Graphs is that it restricts that the question must be asked in a predetermined format.

There are some issues of concern that are found common in all the above discussed literature as follows:

- None of the previous work in this field have focused on all the aspects associated with Question answering like summarization, question analysis, document

representation, answer extraction or ranking. Thus, it has being analyzed that there is a need to design and develop a system that is user interactive and combines all the modules together for efficient question answering.

- The existing systems have not much focussed on the indexing of the Web pages for answering the questions. So, there is a need to design an efficient indexing scheme for designing and developing a fast QA system.

- The referred work in this section collected the pages from the Web. However, it has been found that the information contained in web pages is not of high quality and also, the major portion of the web pages contain information that is not of much importance to the user. Therefore, the web is a huge repository of information but is not likely to contain the information that is focussed. For this purpose, the system to be developed must be able to collect information from some other sources that are likely to be containing focussed information. Also, there is a need to extract only a relevant portion of the text from the pages collected from the quality sources.

A design of a search engine for prospective question answering is being proposed in Chapter-3 that not only addresses the problems prevailing in the recent QA systems but also uses the blogosphere as the major source of the information for improving the quality of question answering.

Chapter III

DESIGN OF A NOVEL SEARCH ENGINE FOR PROSPECTIVE QUESTION ANSWERING

3.1 INTRODUCTION

Search Engine responds to the user's query with the set of web pages returned as the search results. A critical look at the available literature and keeping the observations made thereof in mind, it is felt that a survey needs to be conducted to find the requirements of the users from a general search engine like Google, Yahoo, AltaVista etc and to study whether general purpose search engines are able to fulfil the requirements of users or not. A survey was conducted in two engineering colleges namely YMCA University of Science & Technology, Faridabad and Echelon Institute of Technology, Faridabad among 60 people in total comprising of students, teachers, technical and non-teaching staff. The survey consisted of 21 questions as given in Appendix-1. For each question, a number of choices were given for selection. After conducting the survey, all the responses were collected as shown in Table 3.1, where Qid is the Question in the survey and the options provided are given by a, b, c, d and e.

On analyzing the responses, the following observations have been made:

1. It has been observed that most of the users are of the opinion that, for finding the precise answer to a particular question, the traditional search engine is not a correct choice but a QA system offers better results.
2. It is further observed that blogs contain topical information i.e. information that is very much related to the topic on which the blog has been written, thereby may play a crucial role in providing answers corresponding to user's questions on that topic.
3. It has also been observed that a large number of blogs have been written by experienced bloggers, and the content in blogs is of good quality.

Table 3.1 Survey responses

Qid	a	B	c	c	e
1	57	3			
2	20	19	21		
3	45	15			
4	40	10	7	2	1
5	35	10	15		
6	40	10	10		
7	45	15			
8	10	43	7		
9	8	42	9		
10	40	20			
11	38	5	10	7	
12	38	10	12		
13	47	13			
14	45	15			
15	10	50			
16	48	12			
17	47	10	3		
18	45	15			
19	45	10	5		

A graph has been plotted for the responses received by the participants as shown in Fig. 3.1.

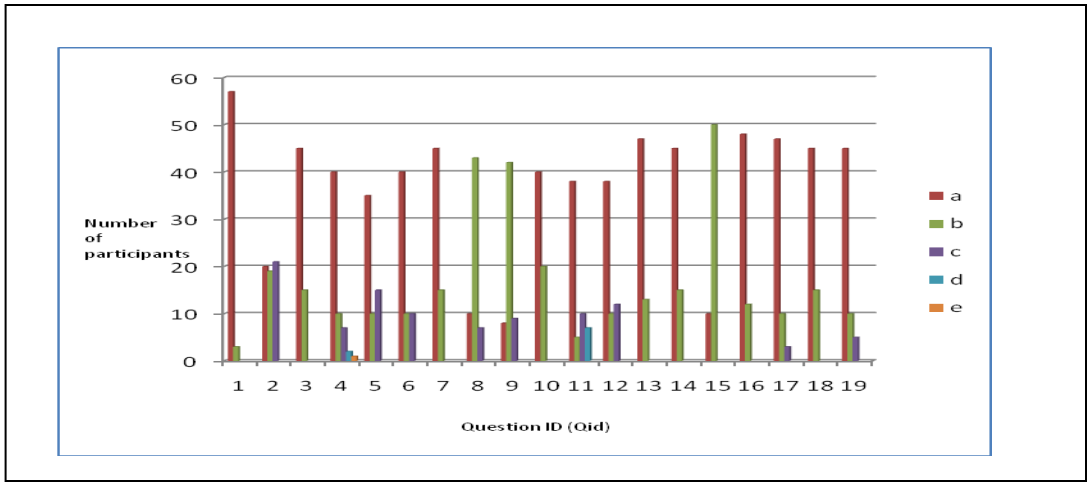


Fig. 3.1 Graph showing the responses of survey conducted

Keeping in mind the need of the question answering system, a framework for question answering system is being proposed that can provide answers to user's questions by searching in the contents maintained in the blogs. The system deals with the four major functionalities- crawling blog pages, extracting relevant content from blog pages, maintaining index of the relevant blog content and then searching in the index for answer(s). The user asks a question and gets answer(s) in response. The detailed architecture of the same has been discussed in the next section.

3.2 PROPOSED DESIGN OF A NOVEL SEARCH ENGINE FOR PROSPECTIVE QUESTION ANSWERING

The proposed design of a novel search engine for prospective question answering [89] is shown in Fig. 3.2. It takes the input from the user in form of questions and returns the answer(s) from its resources. The system consists of the following six functional components:

(i) crawl blogs
(ii) extract relevant content
(iii) index blogs
(iv) classify question
(v) searcher
(vi) look up for alternate data sources

A brief discussion on each of these functional components is given below:

3.2.1 crawl blogs

The *crawl blogs* is one of the major components of the proposed architecture that downloads the blog pages from the WWW. The functionality of the *crawl blogs* is similar to the general crawler with a major difference that it downloads the blog pages only, unlike the general crawler that downloads all types of web pages. Since the blogs update at a greater frequency than the general web pages, the *crawl blogs* needs to recrawl the blogs more frequently to maintain the freshness of the repository. It is possible by

applying more accurate time interval between two successive crawls of the same blog [81]. The process of crawling starts with an initial set of URLs possibly the top N amongst the recent blogs and follows the steps given below repeatedly:

i. Extracts the URLs from the list of seed URLs.
ii. Check whether the URL is that of a blog page.
iii. If yes, download the blog page from WWW and store it in a buffer.
iv. Extract the links existing in the blog pages and add then in the list of seed URLs for further crawling.
v. Extract the title, content and comments existing in each post of the blog page and store them in *page repository* in form of text pages.

It is worth noting that there are some features of blog pages that are different from the features of the general web pages, as listed below:

- There is an ordering among the blog posts in a blog page such that the recent posts appear at the top of the other blog posts.
- The URLs of the blog pages usually contain a *Blog* mark i.e. a string or substring "blog" appears in the URL.
- There exists a RSS tag (Really Simple Syndication), a special feature of blog page which allows sending a notification to the user whenever blog gets updated.
- The majority of the hyperlinks existing in the blog page point to the blog itself or to the other blogs.

Keeping these features in mind, architecture for the blog crawler is designed and implemented. The detailed discussion with experimental analysis of the proposed architecture is given in Chapter-4.

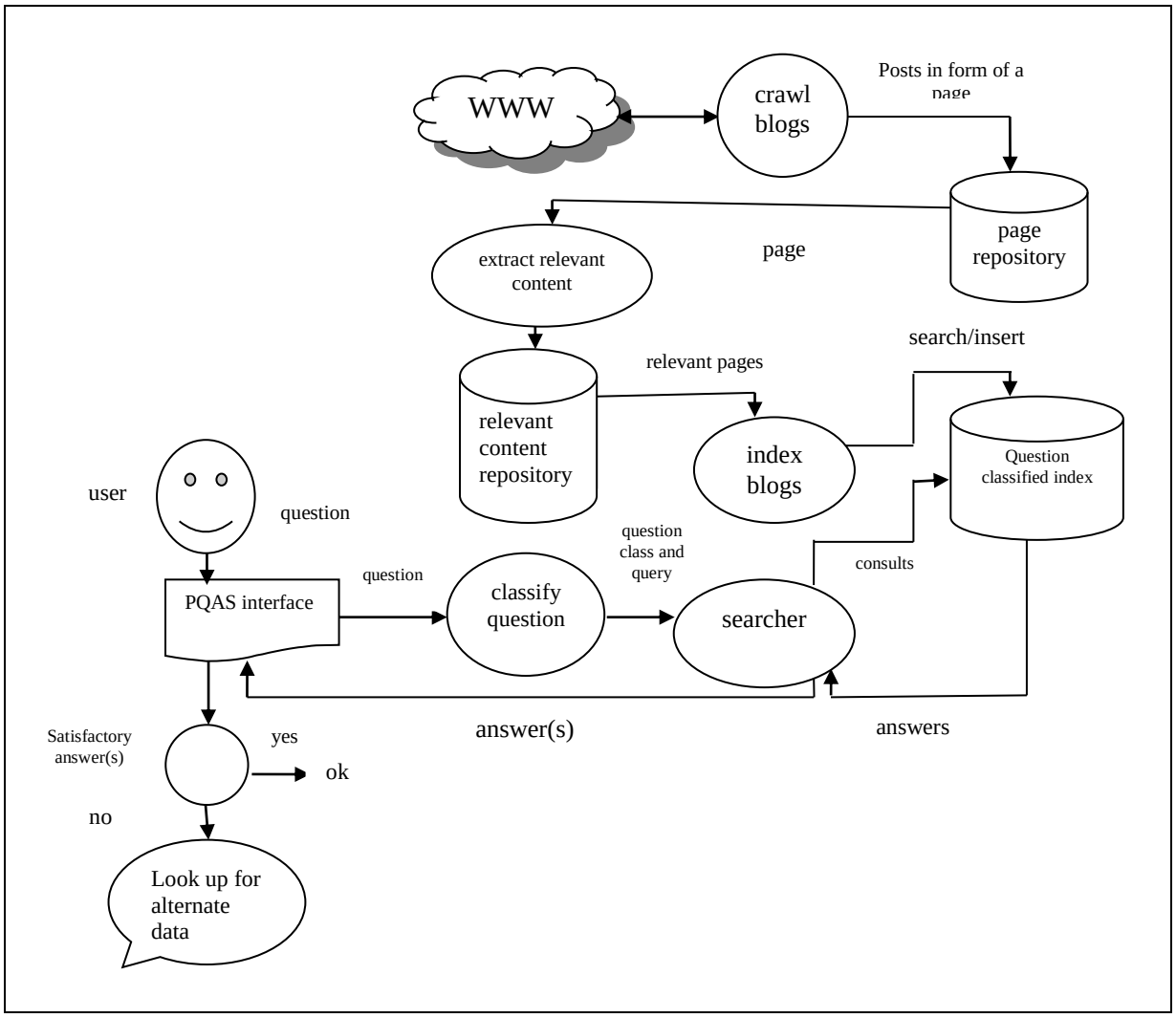


Fig.3.2 Overall system architecture

The discussion on *extract relevant content* module is given in the section below.

3.2.2 extract relevant content

The module reads the page repository and retrieves the blog posts stored in the form of text pages. It then extracts the relevant content from the posts. The task of extracting

relevant content is important because the entire text contained in the blog page may not be useful and relevant as far as user's perspective is concerned. Thus, there is a need to filter the relevant contents from the blog pages. The relevant contents extracted from the blog posts are stored in form of individual text pages in a repository called *relevant content repository*.

A technique has been proposed for extracting the relevant contents from the blog pages. The technique utilizes the title of the blog page for determining the relevancy of the text contained in the blog. One more parameter has been focussed for the relevant content extraction termed as *Presence factor*. This feature considers those sentences as more relevant in which most of the title terms are present. The sentences that contain lesser number of title terms are considered to be less important.

An alternate method has also been proposed for extracting the relevant contents from blog pages. The method best utilizes one additional feature of a blog page namely title of the blog and comments of the blog readers. Since, the comments of the blog readers besides the title are also very important from the point of view of relevant content extraction from the blog post. Therefore, the proposed system also focuses on the valid comments of the blog post, for the purpose of extracting the relevant contents. The detailed system design along with the working of the related modules of both the techniques is given in Chapter-5.

3.2.3 index blogs

The relevant content stored in the form of text pages in the *relevant content repository* is taken as the input by the *index blogs* module to generate *Question classified index*. A system of *Question classification based indexing* has been proposed for efficient question answering. The system maintains an index of the relevant content stored in relevant content repository. The system uses the following steps for indexing:

i. Identify the *termset* of each sentence of the pages stored in the *relevant content repository*.

ii. Taking each of the term in the termset as input, the system then uses the Web definitions [90] for the purpose of obtaining the description of the term through the online sources e.g. dictionaries [91] and/or WordNet [92].

iii. By analysing the web definition, the terms are categorized under an appropriate answer type. For example, the terms mother, father, leader etc. are categorized under the type "person". Table 3.2 provides the categorization of some example terms under appropriate answer type. Like for the terms descriptions-government, agency, company, airline; the appropriate answer type identified is "organization".

iv. The term is then indexed under the identified answer type along with the Sentence id (Sid) and Page id (Pid) in which it exists (see Table 3.3).

Table 3.2 Categorization of terms under appropriate answer type

Term description for the tems obtained using Web definitions	**Answer type**
Government-Agency, Agency, Company, Airline, University, Institute, Sports-Team	Organization
Leader, Father, Mother , Sister, Brother, King, Queen, Emperor, Name	Person
City, Country, State, Territory, Mountain, Island, Street, Land, planet, river, ocean	Location
Full form/short form	Abbreviation
Quantity, Distance, weight, size, temperature, speed, percent, code, count, period, money, currency	Number
Procedure, Method, process	Process
Day, Days of the week	Day
AM, PM	Time
Year	Year
Months of the year like January etc.	Month
Date	Date

Concept, object, protocol, definition	Description
Animal, body, color, disease, medicine, event, food, instrument, language, letter, plant, product, religion, sport, substance, symbol, technique, vehicle	Entity
Explanation	Reason

Table 3.3 shows how the terms are indexed under appropriate answer type(s) along with Sid and Pid.

Table 3.3 Index based on Answer Type(s)

Answer type	Terms	Sid and Pid
Person	Batsman	s5,p7
Person	President	s1,p9
Location	USA	s4,p7
Process	Scheduling	s3,p4
Day	Sunday	s1,p7
Month	December	(s4,p7), (s1,p9)
Abbreviation	ADT	(s5,p1), (s1,p2), (s4,p2)
Abbreviation	WHO	(s5,p4), (s1,p3)
Organization	Corporate	s5,p5
Description	Concept	(s2,p9), (s4,p9)

This index is then used to respond to the user's question. The detailed architecture for *Question classification based indexing* is given in Chapter-6.

3.2.4 classify question

The question provided by the user to the PQAS interface is given to the *classify question* module for classification. To classify the question, at first the module identifies the question class. Since, the proposed system restricts that the question can start with: *Who,*

What, When, Where, Which, Why and How, so the *classify question* module identifies, the first word of the question as the question class.

After identifying the question class, the rest of the question is converted into query that consists of a set of terms. The conversion is done by performing some main tasks involved in pre-processing [20] as given below:

- Tokenization: It is the task of splitting sentences into pieces, called *tokens*, perhaps at the same time throwing away certain characters, such as punctuation.
- Stop word elimination: It is a process of eliminating the commonly occurring or rarely occurring words existing in a sentence like a, an, the, are etc.
- Stemming: It usually refers to a crude heuristic process that cuts off the ends of words, which may often includes the removal of derivational affixes.
- Lemmatization: It usually refers to doing things properly with the use of a vocabulary and morphological analysis of words, normally aiming to remove inflections only and to return the dictionary form of a word, which is known as the *lemma*. If confronted with the token *saw*, stemming might return just *s*, whereas lemmatization would attempt to return either *see* or *saw* depending on whether the use of the token was as a verb or a noun.
- Normalization: It is a process of canonicalizing tokens so that matches occur despite superficial differences in the character sequences of the tokens. For instance, if the tokens *anti-discriminatory* and *antidiscriminatory* are both mapped onto the term *antidiscriminatory.*

For example, if the user asks “What is ADT”, the module identifies “What” as the Question Class and “ADT” as a term of the query. Similarly, for the user’s question “Who is prime minister of India”, the module identifies “Who” as the Question class and “prime”, “minister” and “India” as the query terms.

3.2.5 searcher

Searcher is the component that takes the question class and the terms in the query as input from the *classify question* module and then searches in the index. For the question class, the searcher looks in the second column of the table for finding the Answer type(s) (see Table 3.4) and then searches in the *Question classified index* (see Table 3.3) for the terms in the query under the corresponding answer type.

It then extracts the sentence Id(Sid) and its corresponding page Id (Pid) of the page containing relevant content in which the sentence exists. The sentences are then given as answers to the user on PQAS interface.

Table 3.4 Answer type(s) for Question classes

Question class	**Answer type**
Who	Person, Organization
Where	Location
What	Money, Number, Definition, Procedure, Abbreviation, Organization, Person, Year, Month, Day, Time, Location, entity, date
When	Time, Year, Day, Month, date
Which	Person, Location, Month, Time, Year, Day, organization, entity, date
Why	Reason
How	Process

3.2.6 look up for alternate data sources

Some alternate data sources are used by PQAS to respond the user's question, if it is not able to provide the satisfactory answer(s) to the user. If the user is satisfied with the answer(s) given by alternate data sources, then PQAS updates its *page repository* with the data obtained from the alternate data sources. If the user is not satisfied with the answer given by the alternate data source, then the answer is simply discarded and PQAS doesn't update its database. Also, the PQAS displays an appropriate message in this case. The flowchart of the whole process is shown in Fig. 3.3.

The updation of the *page repository*, with a satisfactory answer(s) returned by the alternate data source is a major step taken for enriching the database maintained by the proposed system. It may be worth noting that this step takes place in increments and enriches the PQAS database gradually. Some sample snapshots showing the answer(s) given for few questions by PQAS and for the same question by the alternate data sources are given in Appendix-9.

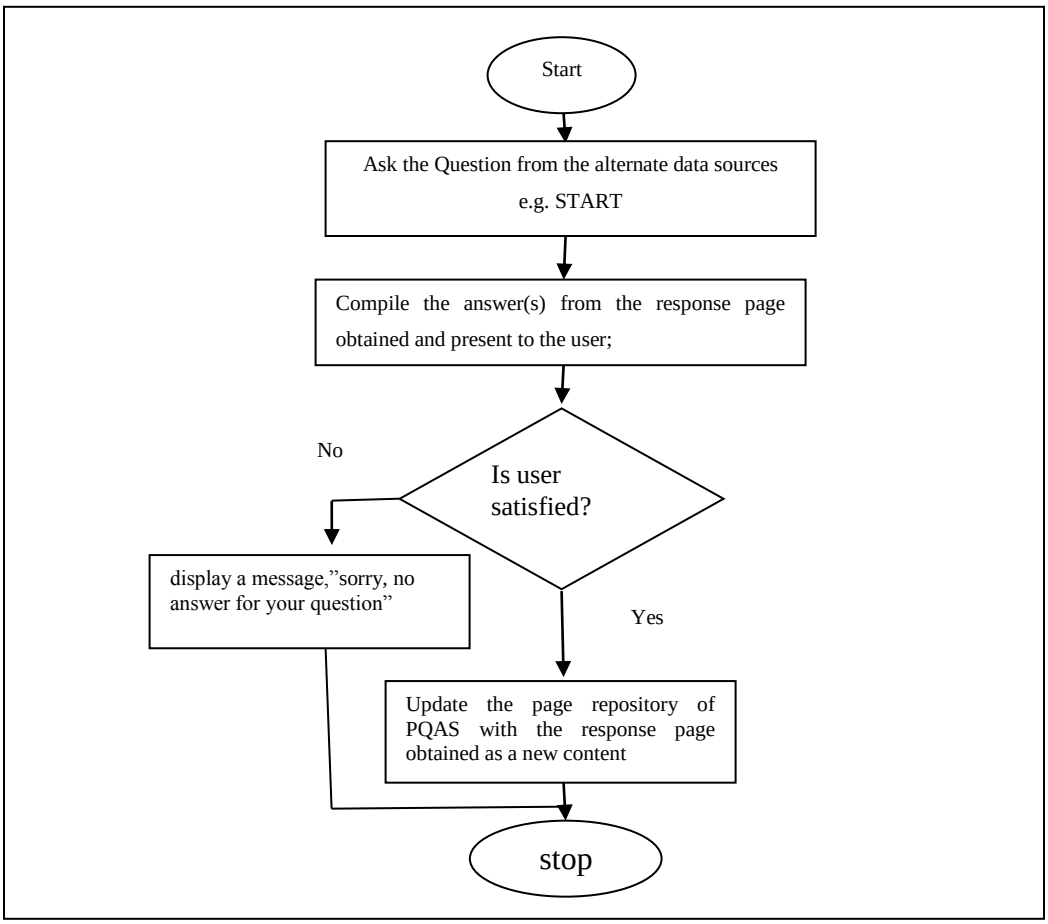


Fig. 3.3 Process: look up for alternate data sources

The algorithm for *Look up for alternate data sources* module is given below in Fig. 3.4.

```
Algorithm Look up for alternate data sources ( ){
Step 1. ask the Question from the alternate data sources
     2. compile the answer(s) from the response page obtained and present the answer(s)
        to the user on PQAS interface;
   3.   if (the user is satisfied with the answer given)
        update the page repository of PQAS with the response page obtained as a new content;
        else
        display a message. ”sorry, no answer for your question”}
```

Fig. 3.4 Algorithm: look up for alternate data sources

The architecture along with the detailed description of the first module of the proposed system is given in Chapter-4.

Chapter IV

A NOVEL ARCHITECTURE FOR A BLOG CRAWLER

4.1 GENERAL

Blogging [39,60] is a rising trend and serves as a very useful source of information. For retrieving content from the blogs, a blog crawler is needed. Unlike a general web crawler that downloads the web pages, blog crawler downloads the pages from blogosphere. Hence, a blog crawler [42,58,59] is similar to a general crawler with a difference that its crawl boundary is restricted to the blogosphere only.

4.2 PROPOSED ARCHITECTURE OF BLOG CRAWLER

The blog crawler [89] is a major module of the proposed architecture for Question Answering system. It downloads the blog pages from the blogosphere. While designing a blog crawler, some of the features that make a blog page different from a general web page are considered, as given below:

- There is an ordering among the blog posts, the latest posts appearing on the top of other blog posts.
- The URL of the blog pages usually contain a "blog" mark i.e. a string or a substring "blog" appears in the URL.
- There exists a RSS tag (Really Simple Syndication) [100], a special feature of blog page which allows the user to subscribe to blogs and receive notification on updates.
- Majority of the hyperlinks existing in a blog page points to itself or to the other blogs.

Also, a blog page is different from other web pages in the sense that it contains the author's information, blog likes/dislikes, log archives, comments by blog readers, blog rating etc.

The architecture of blog crawler consists of the following four functional components (see Fig. 4.1):

(i) extract URLs

(ii) check for blog

(iii) extract blog

(iv) extract post

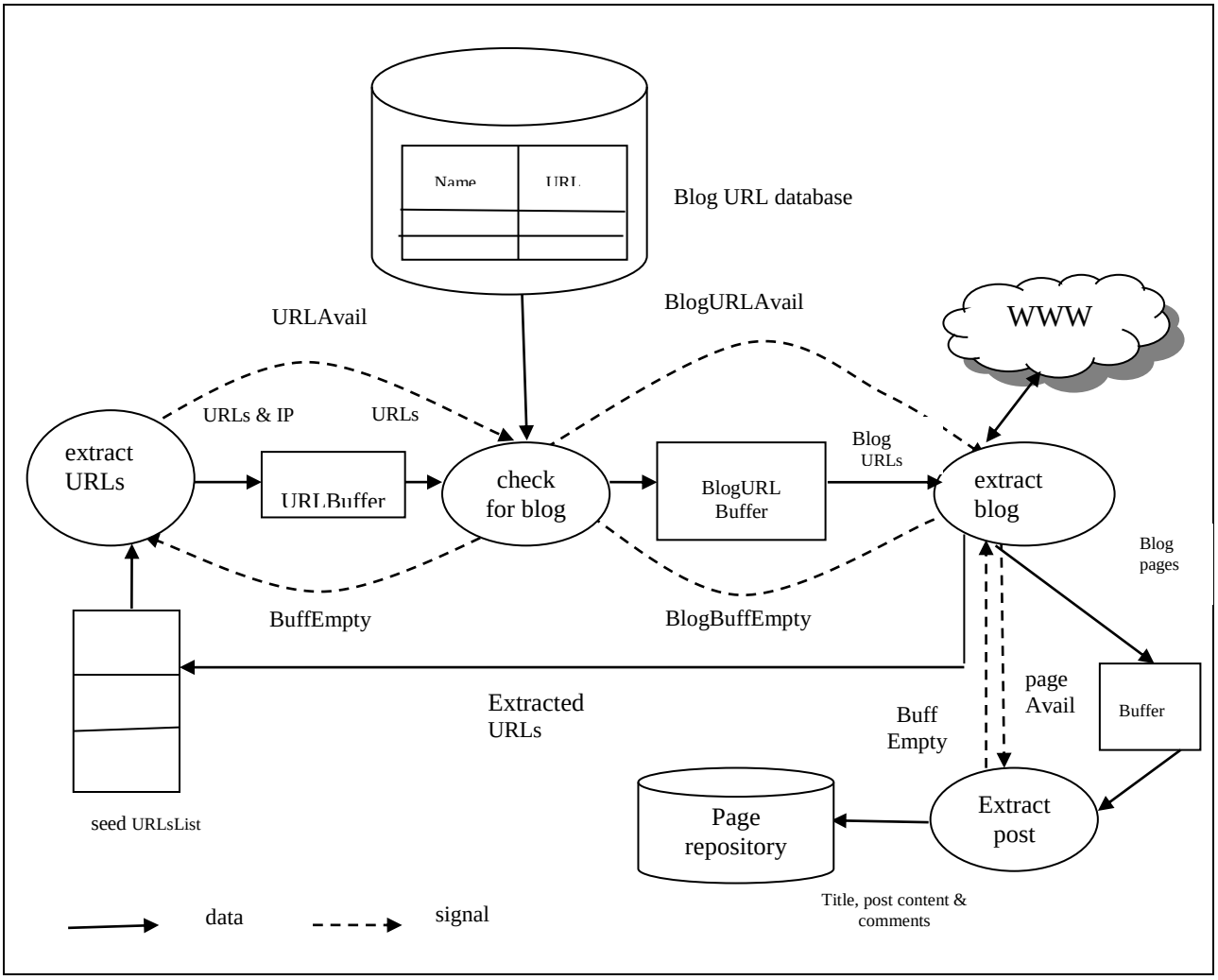


Fig. 4.1 Proposed design of blog crawler

The functionality of each component of blog crawler is given below:

4.2.1 extract URLs

Blog crawler uses a data structure that is a list of seed URLs termed as *SeedURLsList. extract URLs* module extracts URLs from *SeedURLsList* , resolves their domain name into their IP address and then stores these URLs along with their IP in a buffer termed as *URLBuffer.* Then it sends a signal *URLAvail* to the next module i.e. *check for blog* for URL availability which then fetches URLs from *URLBuffer* and check each URL that whether it is a URL of blog page or not. Also, if the buffer becomes empty, the module sends the signal *BuffEmpty* to the extract*URLs,* so that more URLs can be added further in the *URLBuffer*. The algorithm for the module is given in Fig. 4.2.

```
Algorithm extractURLs()
{
    wait (BuffEmpty)
    {
    Step 1. extract URLs from SeedURLsList
         2. resolves their domain name into their IP address
         3. store them in URLBuffer
    }
    signal(URLAvail)
}
```

Fig. 4.2 Algorithm: *extract URLs*

The explanation and algorithm of the next module i.e. *check for blog* is given in the next section.

4.2.2 check for blog

On receiving *URLAvail* signal from *extract URLs,* the c*heck for blog* fetches URLs and their IP addresses from the *URLBuffer* and checks each of these URLs that whether these are URLs of blog pages or not. For checking that whether URLs are of blog pages, it

looks for the “blog” mark in the URL. If a “blog” mark is found in the URL, it means that the URL is that of a blog page. If no “blog” mark is found then c*heck for blog* looks for RSS. If none of these features exist, the page is considered as a general page and not as a blog page, thus the URL is simple rejected. The blog URLs found are stored in *BlogURLBuffer* temporarily and the c*heck for blog* then sends a signal *BlogURLAvail* to the *extract blog* for downloading the blog pages from the WWW and extracting the links existing in them. Also, the ex*tract blog* sends *BuffEmpty* signal to the c*heck for blog* on buffer empty. The algorithm for the module is given in Fig. 4.3.

```
Algorithm check for blog ( ){
    wait(URLAvail){
    do{
    Step 1.  fetch URLs and their IP from URLBuffer
          for each URL{
          2. if(“blog” mark in the URL)
          3. store the URL in BlogURLBuffer
          4. else if(no “blog” mark){
          5. check for RSS
          6. if(RSS is there)
          7. store the URL in BlogURLBuffer }}
       }while(buffer is not empty);
    }signal(BlogURLAvail)
    signal(BuffEmpty)}
```

Fig. 4.3 Algorithm: *check for blog*

It is worth noting that to check for RSS, the *check for blog* module uses a database termed as *Blog URL database*. This database contains names of blog sites (see Appendix-2) that have RSS feed, which allows the user to subscribe to the blog and receive notification on their updates. The subscriber receives the updated blog post and can also go through the other content in the blog site. Along with the names of these sites, the

database also contains their URLs. The module *check for blog* uses this database and compares the incoming URLs with those stored in the database. If a match is found with any of the URLs existing in the database, then the URL is considered as the URL of a blog page. If no match is found, then it indicates that the URL is of a non-blog page and is simply discarded. The explanation and algorithm of the next module i.e. *extract blog* is given in the next section.

4.2.3 extract blog

On receiving the *BlogURLAvail* signal from the *check for blog,* the e*xtract blog* module reads URLs and their IP from the *BlogURLBuffer* and then traverses the WWW for downloading the blog pages. The downloaded blog pages are stored in a buffer. If the buffer becomes empty, the module sends a *BlogBuffEmpty* signal to the *check for blog* module. From the downloaded blog pages, the *extract blog* module extracts all links existing in them, converts the relative URLs to absolute URLs and adds them in *SeedURLsList* for further crawling. The *extract blog* module sends *pageavail* signal to the *extract post* module for extraction of blog post from the pages. The module *extract post* module sends a *buffempty* signal to the *extract blog* on buffer empty. The algorithm for the module is given in Fig. 4.4.

```
Algorithm extract blog ( ){
    wait (BlogURLAvail);
    wait(BuffEmpty){
    Step 1. extract BlogURLs and their IP from BlogURLBuffer
        2. download blog pages corresponding to BlogURLs
        3. store blog pages in buffer
        4. extracts all links existing in them
        5. converts the relative URLs to absolute URLs}
    signal (BlogBuffEmpty);
    signal(pageAvail)}
```

Fig. 4.4 Algorithm: *extract blog*

It is worth noting that some authoritative blog sources such as blogspot.com, wordpress.com, blog.technet.com etc. are chosen for selecting the seed blog URLs (see Table 4.1).

Table 4.1. List of Blog sources

S.No.	Blog sources	URL
1	Wordpress	http://wordpress.com
2	Blogspot	http://www.blogger.com/home
3	Blogs.technet	http://blogs.technet.com
4	Technorati	http://technorati.com
5	Weblogs.com	http://weblogs.com
6	Blogpopular	http://www.blogpopular.net/

The next module *extract post* extracts the posts existing in the blog pages crawled.

4.2.4 extract post

The module *extract post* waits for the signal *pageAvail* from the *extract blog* module. On receiving the signal, it takes blog pages from buffer and extracts the title, blog post content and comments from each of them. The title, post content and comments that have been extracted are now stored in form of a text page in *page repository*. The algorithm for the extract post is given in Fig. 4.5.

```
Algorithm extract post ( ){
 wait(pageAvail){
do{      Step1.fetch blog pages from the buffer
              2.for each blog post that exists in the blog page
           do
              2.1 extract the title of the blog post
              2.2 extract the posts' content
              2.3 extract the comments on the blog post
              2.4 store them as a single text page in the page repository
}while(the buffer is not empty);}
signal(buffEmpty)  }
```

Fig. 4.5 Algorithm: *extract post*

The module passes *buffEmpty* signal to the *extract post* module, if the buffer becomes empty. On receiving the signal *buffempty*, the *extract post* module sends more pages to the buffer, so that the title, post content and comments can be extracted further from the blog pages.

4.3 EXPERIMENTAL EVALUATION OF THE PROPOSED SYSTEM

For the implementation of the proposed system, Java 1.8 is chosen. Several experiments are conducted over various sources on the Web to evaluate the performance of the proposed blog crawler and it is found that the proposed blog crawler has shown fairly consistent results. The metrics used for the evaluating the performance of the proposed system is given below.

4.3.1 PERFORMANCE METRICS

There are three performance metrics that are used namely Precision, Recall and F-measure [84,85,86,87]. To define these metrics, the following terms are used:

a. *Precision (P)* is defined as a fraction of blog pages crawled over all the pages crawled by the proposed blog crawler.

   Mathematically, *Precision* is given by:

   $$P = BP/(BP + WBP) \quad \text{.....eq 4.1}$$

   where BP is the number of blog pages i.e. relevant pages crawled and WBP is the number of non-blog pages i.e. irrelevant pages crawled.

b. *Recall (R)* is defined as a fraction of blog pages crawled over all the relevant blog pages. The relevant pages are the blog pages that are crawled and also those blog pages that are not crawled by the proposed crawler but are relevant.

   Mathematically, *Recall* is given by:

   $$R = BP/(BP + NBP) \quad \text{.....eq.4.2}$$

   where BP is the number of blog pages crawled and NBP is the number of blog pages not crawled but are relevant.

c. *F-measure (F)* combines both precision and recall.

Mathematically, *F-measure* is given by:

$$F = 2PR/(P + R) \quad \ldots\ldots\ldots\ldots \text{eq.4.3}$$

where an equal weight is assigned to both P and R.

4.3.2 DATA SETS

For experimental analysis of the proposed blog crawler, list of seed URLs are chosen from the following four well known blog sources:

- http://www.elegantthemes.com/blog/
- http://www.computersciencedegreehub.com/
- http://technorati.com/
- http://www.blogdirs.com/blog/

and the crawler starts the crawling process to download the pages from the list.

The runs of the proposed blog crawler and the process of experimental evaluation are given below.

a). Run 1: On the first run of the blog crawler, it collected about 20 response pages corresponding to each of the assigned URLs, thus collectively a sample of 82 blog pages is collected. Out of 82 pages that have been crawled, 71 pages are the blog pages and 11 are the non-blog pages. There are 15 pages that not crawled by our proposed crawler. So, using the terms defined above:

BP=71, WBP=11, NBP=15. Using eq. 4.1, 4.2 and 4.3,

Therefore, P=71/(71+11)=86.5%

R=71/(71+15)=82.5%

and F=2*86.5*82.5/(86.5+82.5)=84.4%.

b). Run 2: On the second run of the proposed crawler, it collected about 30 response pages corresponding to each of the assigned URLs, thus collectively a sample of 120 blog pages is collected. Out of 120 pages that are crawled, 100 pages are the blog pages and 20 are the non-blog pages. There are 25 pages that are not crawled by our proposed crawler. So, using the terms defined above:

BP=100, WBP=20, NBP=25. Using eq. 4.1, 4.2 and 4.3,

Therefore, P=100/(100+20)=83.3%

R=100/(100+25)=80%

and F=2*83.3*80/(83.3+80)=81.6%.

c). Run 3: On the third run of the proposed crawler, it collected about 40 response pages corresponding to each of the assigned URLs, thus collectively a sample of 158 blog pages is collected. Out of 158 pages crawled, 134 pages are the blog pages and 24 are the non-blog pages. There are 34 pages not crawled by our proposed crawler. So, using the terms defined above:

BP=134, WBP=24, NBP=34. Using eq. 4.1, 4.2 and 4.3,

Therefore, P=134/(134+24)=84.8%

R=134/(134+34)=80%

and F=2*84.8*80/(84.8+80)=82.3%.

d). Run 4: On the fourth run of the proposed crawler, it is found that it collected about 50 response pages corresponding to each of the assigned URLs, thus collectively a sample of 197 blog pages is collected. Out of 197 pages crawled, 169 pages are the blog pages and 28 are the non-blog pages. There are 34 pages that are not crawled by our proposed crawler. So, using the terms defined above:

BP=169, WBP=28, NBP=34. Using eq. 4.1, 4.2 and 4.3,

P=169/(169+28)=85.7% ,

R=169/(169+34)=83.2%

and F=2*85.7*83.2/(85.7+83.2)=84.4%.

Summarizing, the Precision P is found in the range of 83.3% to 86.5%, recall R is found in the range of 80% to 83.2% and the F-measure F is found in the range of 81.6% to 84.4% (see Table 4.2 for the values of P, R and F). On average, P, R and f are found to be 85.0%, 81.4% and 83.1%.

Table 4.2 Precision, Recall and F-measure values of blog crawler

Run #	P (in %)	R (in %)	F (in %)
1	86.5	82.5	84.4
2	83.3	80	81.6
3	84.8	80	82.3
4	85.7	83.2	84.4
Average	85.0	81.4	83.1

The value of Precision, Recall and F-measure for each of the four runs of the proposed blog crawler is depicted graphically in Fig. 4.6.

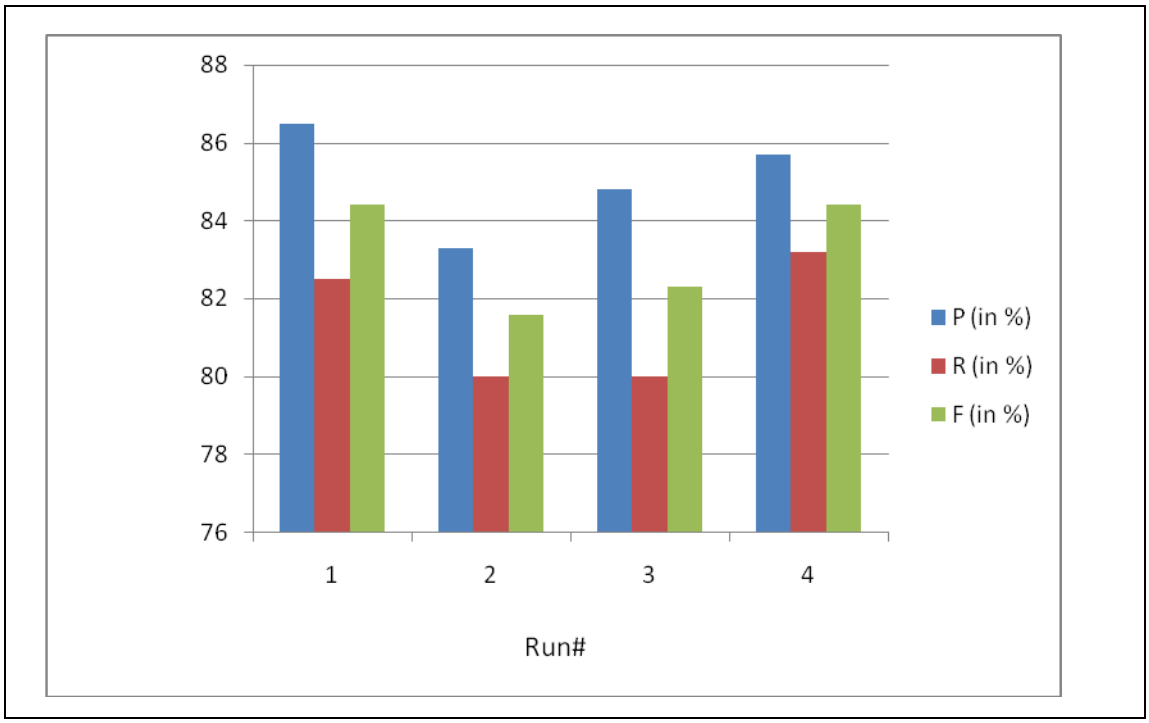


Fig. 4.6 P, R and F values for each of the four cases of blog crawler

It may be noted that average *Precision*, *Recall* and *F-measure* as shown in Table 4.2 is more than 80% suggesting a high performance for the proposed blog crawler.

The snapshots of the implementation of proposed blog crawler are shown in Appendix-3. It is worth noting that from the blog pages crawled, the *extract blog* module extracts the blog posts and stores each of them in the *page repository* in form of text pages where each text page comprises of title of the post, post content and comments, if existing. Each page in page repository is read by the *extract relevant content* module that extracts the relevant content contained in the pages. The techniques of *relevant content extraction* are discussed in Chapter-5.

Chapter V

RELEVANT CONTENT EXTRACTION FROM BLOG PAGES

5.1 GENERAL

In comparison to general web pages, the posts written in blog pages contain the content that is more likely to be related to the topic on which the blog post is written. So, the topic of each post contains the distilled information, very relevant to the process of relevant content extraction [105]. Moreover, the blog readers provide their feedback by giving comments on the post written by the bloggers. Thus the comment section of blog pages also plays a crucial role in relevant content extraction. It is worth noting that the post written on a topic consists of huge amount of text, all of which may not be relevant for the user. So, there is a need to extract the relevant content [53,54,57,72] from the blog pages. Two techniques for relevant content extraction from blog pages are discussed in this chapter.

5.2 PRESENCE FACTOR ORIENTED RELEVANT CONTENT EXTRACTION FROM BLOG PAGES

In general a blog page may consist of a number of blog posts comprising of the content that is very much related to the topic of the blog post. So, the title of a blog post plays an important role for identification and extraction of the relevant content existing in it. Thus the technique presented in the current section focuses on the title of the blog post. Also, an approach of Presence factor (PF) has been proposed [93] which indicates whether each term existing in the title of the blog post is present in a sentence of the blog post. This is a key feature of the proposed system because it considers those sentences of the blog post as more relevant in which all the title terms appear. The proposed PF based system for relevant content extraction from blog post is shown in Fig. 5.1.

There are three major modules of the proposed system explained as follows:

(i) extractor

(ii) linguistic module

(iii) Presence factor based sentence score generator (PFSS)

The detailed explanation of these modules is given in the sections below:

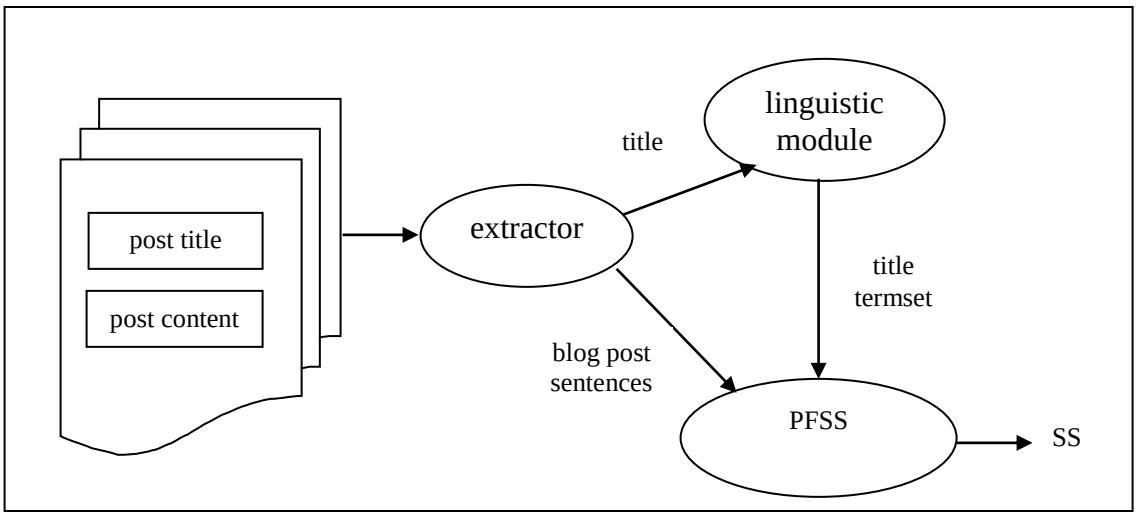


Fig. 5.1 PF based relevant content extraction

5.2.1 extractor

This module separates the title and post sentences from a blog post. The extracted title is then given an input to the *Linguistic module* and the blog post sentences are given to the PFSS module.

5.2.2 linguistic Module

This module carries out the following functions on the title extracted from the blog post for the purpose of generating the title termset from the title:

- Tokenization: It is the task of splitting sentences into pieces, called *tokens*, perhaps at the same time throwing away certain characters, such as punctuation.
- Stop word elimination: It is a process of eliminating the commonly occurring or rarely occurring words existing in a sentence like a, an, the, are etc.
- Stemming: It usually refers to a crude heuristic process that cuts off the ends of words, which may often includes the removal of derivational affixes.

- Lemmatization: It usually refers to doing things properly with the use of a vocabulary and morphological analysis of words, normally aiming to remove inflectional only and to return the dictionary form of a word, which is known as the *lemma*. If confronted with the token *saw*, stemming might return just *s*, whereas lemmatization would attempt to return either *see* or *saw* depending on whether the use of the token was as a verb or a noun.
- Normalization: It is a process of canonicalizing tokens so that matches occur despite superficial differences in the character sequences of the tokens. For instance, if the tokens *anti-discriminatory* and *antidiscriminatory* are both mapped onto the term *antidiscriminatory.*

The result of performing these functions is a termset T that contain all the terms in the title of the blog post. Formally, T = $\{t_1, t_2, t_3 \ldots, t_n\}$ where t_i is the i^{th} term in the title termset T.

5.2.3 Presence factor based sentence score generator (PFSS)

This module takes title termset and the sentences of the blog post as input and finds sentence score (SS) for each blog sentence. Considering the title termset and the post sentences, this module generates a matrix named Term-sentence matrix (TSM). Each term belonging to the title termset is represented corresponding to a row of the TSM and each sentence of the blog post is represented corresponding to a column of TSM. Each value in the TSM is represented by TSM[i, j] indicates the frequency of term T_i of the blog title in the sentence S_j of the blog post (see Fig. 5.2). The module also generates another matrix called Presence Factor Matrix (PFM) (see Fig. 5.3), with the title terms represented corresponding to the rows and the post sentence corresponding to the columns of the matrix. Each entry in PFM given by PFM [i, j] indicates the presence of term T_i of the blog title in a sentence given by S_j. If the term is present in the sentence, a value '1' is stored in the matrix otherwise a value '0' is stored.

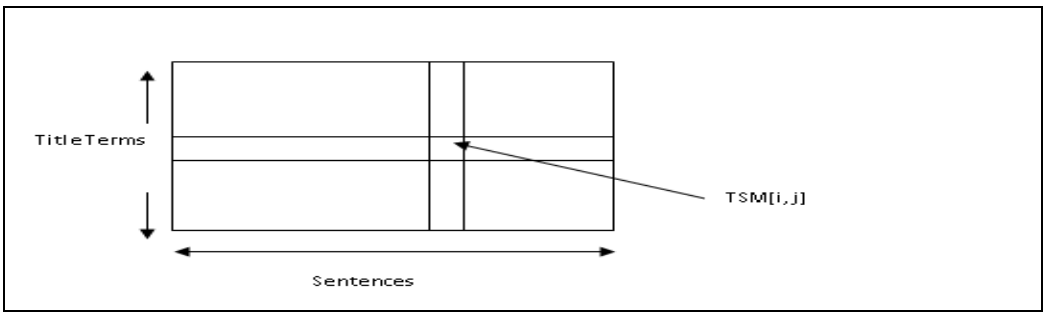


Fig. 5.2 Term-sentence matrix TSM

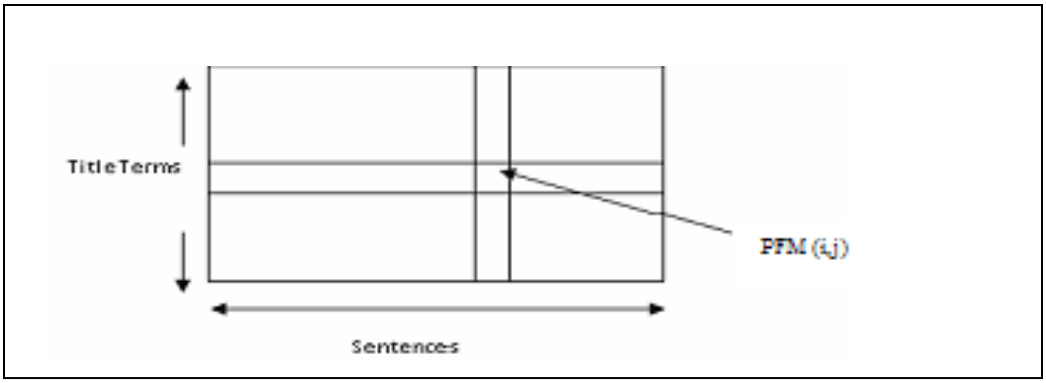


Fig. 5.3 Presence factor matrix PFM

Using both TSM and PFM, the module PFSS generator then computes a score for each sentence called *Sentence Score* given by SS, by using the following formula:

$$SS(S_j) = \Sigma_{i=1 \text{ to } n} TSM(i, j) * \Sigma_{i=1 \text{ to } n} PFM(i, j) \quad \text{.....eq. 5.1}$$

where *SS (Sj)* is Sentence Score for j^{th} sentence,

TSM(i, j) is the frequency of i^{th} term of blog title in j^{th} sentence,

and *PFM(i, j)* is the presence of i^{th} term of blog title in j^{th} sentence.

Presence factor is a key feature of the proposed system which ensures that the proposed system considers those sentences as more relevant that comprises of each term existing in the title termset and also a greater score is assigned to those sentences in the post that contains more number of title terms. Thus, it can be observed from eq. 5.1 that the *SS (Sj)* will increase with the increase of occurrences of title terms in that sentence. Using this approach, sentences are arranged in decreasing order of their *Sentence Score*. The system

then picks up the top k sentences as relevant. The next section gives the experimental analysis of the proposed approach.

Example: Consider the sample blog post:

> **Post Title:** object oriented programming
>
> **Post content:** Object Oriented Programming (OOP) is a paradigm shift in programming which defines, creates, and manipulates objects to develop reusable software. Then why not write programs using objects which would be very natural way of creating useful software comprising of interacting objects. The structured programming techniques did enforce some discipline on the programmers by way of shunning the use of 'goto' statements and encouraging the programmers to use 'easy to read and difficult to write' style of code statements i.e. choosing long and meaningful names for the variables, functions, procedures, modules etc. A program about a University would involve objects like students, professors, clerks, class rooms, books, chalk, mark sheets etc.

For the post title "object oriented programming", the title termset is given as follows:
Title termset=object, orient, program
The post sentences separated are:

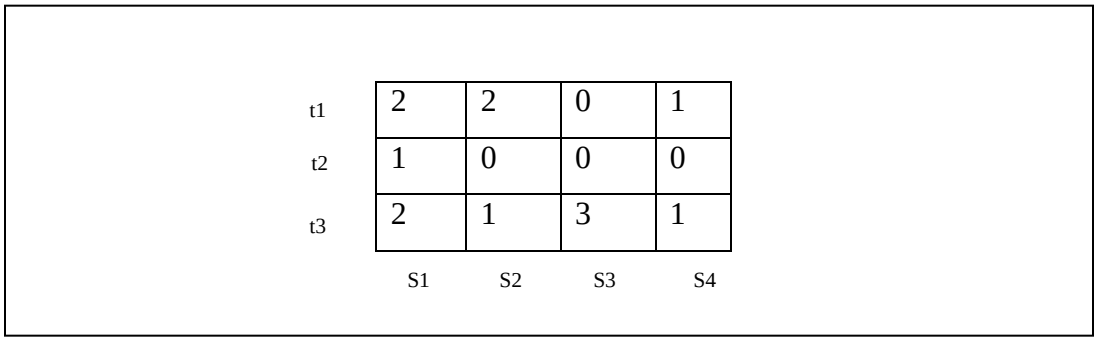
Sentence1: Object Oriented Programming (OOP) is a paradigm shift in programming which defines, creates, and manipulates objects to develop reusable software.

Sentence 2: Then why not write programs using objects which would be very natural way of creating useful software comprising of interacting objects.

Sentence 3: The structured programming techniques did enforce some discipline on the programmers by way of shunning the use of 'goto' statements and encouraging the programmers to use 'easy to read and difficult to write' style of code statements i.e. choosing long and meaningful names for the variables, functions, procedures, modules etc.

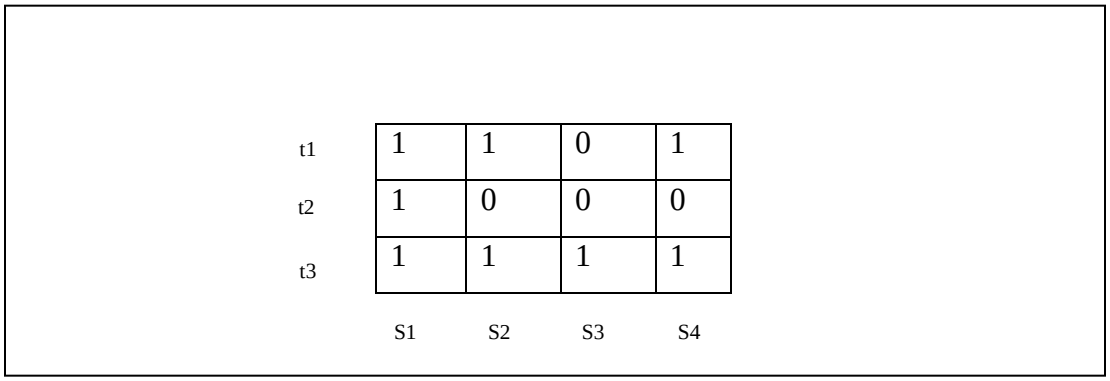
Sentence 4: A program about a University would involve objects like students, professors, clerks, class rooms, books, chalk, mark sheets etc.

The TSM and PFM have been constructed as shown in Fig. 5.4 and Fig. 5.5 respectively.



	S1	S2	S3	S4
t1	2	2	0	1
t2	1	0	0	0
t3	2	1	3	1

Fig. 5.4 Example TSM



	S1	S2	S3	S4
t1	1	1	0	1
t2	1	0	0	0
t3	1	1	1	1

Fig. 5.5 Example PFM

For s1: PFSS=5*3=15

For s2: PFSS=3*2=6

For s3: PFSS=3*1=3

For s4: PFSS=2*2=4

So, among the scores calculated above, the highest score is of sentence 1 i.e. SS=15. This is because it contains each title term. Sentence 2 has higher score than sentence 4 because it has more term frequencies than those in sentence 4. Sentence 3 has only one term present in it, so it has lowest score among the four.

5.2.4 EXPERIMENTAL EVALUATION OF THE PROPOSED SYSTEM

The proposed system has been implemented on .Net framework using C# as front end and MS-SQL as back end. The system collected about twenty five blog pages from

various blog sources on which the proposed approach of relevant content extraction was applied. The relevant content generated by applying the proposed approach on four sample blog pages is given in Appendix-4. To show the performance of proposed system, a sample blog page has been taken from the blog source *http://uanditalk.blogspot.in* shown in Fig. 5.6.

Fig. 5.6 Snapshot the sample blog on Object-oriented programming

For analysis of the content generated by the proposed system, a model text has been used generated by an expert of the same field. Relevant content generated by the proposed system has also been compared with that generated by the following online tools:

a. www.freesummarizer.com

b. www.smmry.com

c. www.textcompactor.com

The content generated by the online tools available for the blog post shown above is given in Table 5.1.

Table 5.1 Content generated by the online tools

Using online tool (www.freesummarizer.com	The journey started with programmers who would write programs which somehow worked without giving any importance to readability of the program. The structured programming techniques did enforce some discipline on the programmers by way of shunning the use of 'goto' statements and encouraging the programmers to use 'easy to read and difficult to write' style of code statements choosing long and meaningful names for the variables, functions, procedures, modules etc. This technique worked good for hard core programmers who were able to write large and complex programs using structured programming techniques. Then why not write programs using objects which would be very natural way of creating useful software Comprising of interacting objects. Object Oriented Programming (OOP) is a paradigm shift in programming which defines, creates, and manipulates objects to develop reusable software.
Using online tool (www.smmry.com**)**	The journey started with programmers who would write programs whichsomehow worked with out giving any importance to readability of the program.The languages like FORTRAN and BA SIC neither enforced any discipline norwere the programmers trained to write user centric prog rams.The structured programming techniques did enforce some discipline on theprogrammers b y way of shunning the use of 'goto' statements andencouraging the programmers to use 'easy to read and difficult to write' styleof code statements choosing long and meaningful names for the variables,functions, procedures, modules etc.In fact any real world program has to be a collecti on objects.A program about a University would involve objects like students, professors,clerks, class rooms, books, chalk, mark sheets etc..
Using online tool (www.textcompactor.com**)**	The programming process has evolved through many phases. The journey started with programmers who would write programs which somehow worked without giving any importance to readability of the program. The major problem was that the programs were not maintainable. The structured programming techniques did enforce some discipline on the programmers by way of shunning the use of 'goto' statements and encouraging the programmers to use 'easy to read and difficult to write' style of code statements choosing long and meaningful names for the variables, functions, procedures, modules etc. This technique worked good for hard core programmers who were able to write large and complex programs using structured programming techniques.

The content generated by human expert and that by the proposed approach is shown in Table 5.2.

Table 5.2 Content by expert and proposed approach

By Expert	The structured programming techniques did enforce some discipline on the programmers by way of shunning the use of 'goto' statements and encouraging the programmers to use 'easy to read and difficult to write' style of code statements choosing long and meaningful names for the variables, functions, procedures, modules etc. When we look around then we find that we are surrounded by nothing but objects only. Then why not write programs using objects which would be very natural way of creating useful software comprising of interacting objects. For example, in our day to day life we compose bigger objects from smaller objects. Object Oriented Programming (OOP) is a paradigm shift in programming which defines, creates, and manipulates objects to develop reusable software**.**
Using proposed approach	Object Oriented Programming (OOP) is a paradigm shift in programming which defines, creates, and manipulates objects to develop a reusable software. Then why not write programs using objects which would be very natural way of creating useful software comprising of interacting objects. In fact any real world program has to be a collection objects. A program about a University would involve objects like students, professors, clerks, class rooms, books, chalk, mark sheets etc. The structured programming techniques did enforce some discipline on the programmers by way of shunning the use of 'goto' statements and encouraging the programmers to use 'easy to read and difficult to write' style of code statements i.e. choosing long and meaningful names for the variables, functions, procedures, modules etc.

The snapshot of implementation of the proposed system is shown in Fig. 5.7.

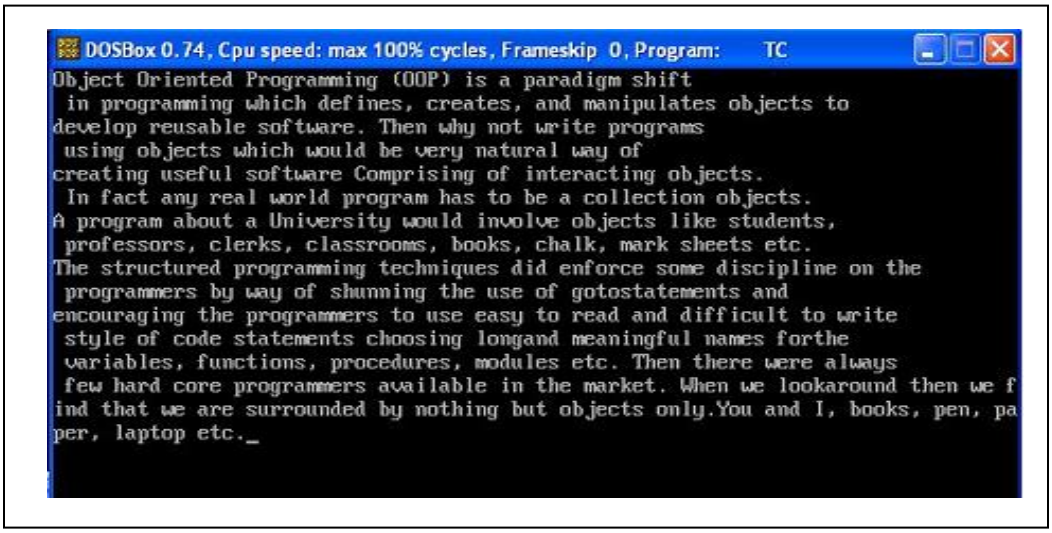


Fig. 5.7 Snapshot of relevant content generated

For the performance evaluation of the content generated by the proposed approach, two performance metrics namely *precision* and *recall* that have been used as defined below:

1. *Precision* is defined as a fraction of number of common sentences that appear in both the content to be evaluated and the content given by expert over the total number of sentences appearing in the content to be evaluated.

Precision is given by:

$$P = N_c / N_s, \ldots\ldots\ldots\ldots\text{eq. 5.2}$$

where N_c is the number of common sentences and N_s is the number of sentences in the content to be evaluated.

2. Recall is defined as a fraction of number of common sentences that appear in both the content to be evaluated and the content given by expert over the total number of sentences that appear in the content given by expert.

Recall is given by:

$$R = N_c / N_m, \ldots\ldots\ldots\ldots\text{eq. 5.3}$$

where N_c is the number of common sentences and N_m is the number of sentences in the content given by expert.

The value of Precision and Recall computed using eq. 5.2 and eq. 5.3 for the proposed technique is given in Table 5.3. The same performance metrics has been applied on the contents given by the online tools.

Table 5.3 Precision and recall values of PF oriented approach

Approach used	Precison (%)	Recall (%)
Proposed Approach	60	60
www.freesummarizer.com	60	60
www.smmry.com	40	40
www.textcompactor.com	20	20

The graph showing these resulting values is plotted in Fig. 5.8.

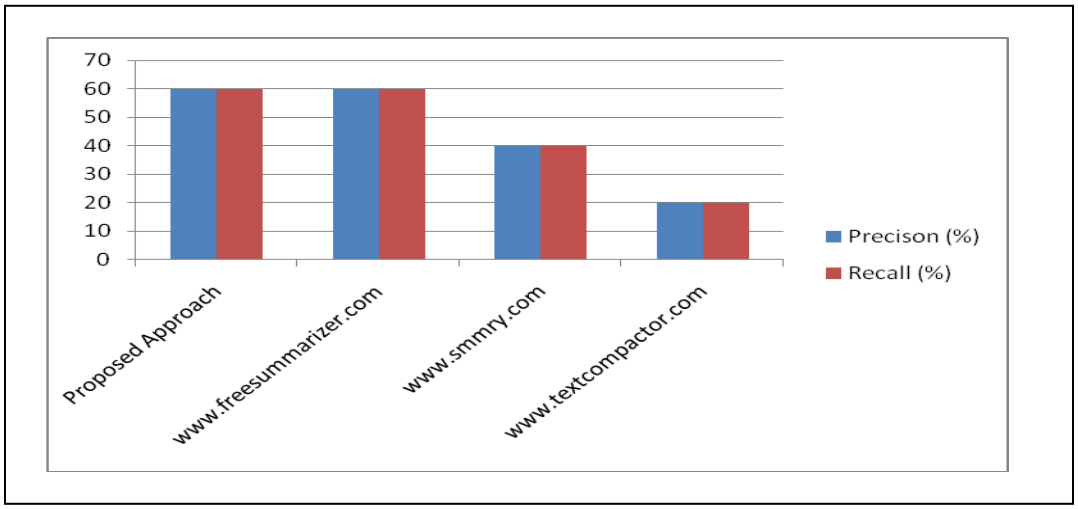


Fig. 5.8 Graph for precision and recall values of PF oriented approach

On applying the proposed technique on the sample consisting of four blog pages (see Appendix-4 for the contents generated), it has been found that the precision and recall w.r.t. the relevant content generated by the expert ranges between 80% to 85.5% and 78.9% to 83.7% respectively (see Table 5.4 for the values).

On analysis of the content generated by the proposed approach and the content given by the online tools show that the proposed approach works well and generates the content of good quality. The values of *precision* and *recall* are found to be higher for the content generated by the proposed system. Thus, the proposed approach works very well in extraction of relevant content from blog pages. Thus, it is found that the Presence factor is a major criterion for selection of sentences that best represent a blog.

Table 5.4 Precision and recall values for sample blog pages

URL of the blog page	Precison (%)	Recall (%)
https://raygun.io/blog/2014/10/5-interesting-data-structures-algorithms/	**82.5**	**80**
http://www.ashishsharma.me/2011/08/java-garbage-collection-notes.html	**80**	**78.9**
http://blog.betafamily.com/2014/07/08/testing-techniques-black-white/	**84**	**82.5**
http://javahungry.blogspot.com/2013/04/scheduling-algorithm-first-come-first.html	**85.5**	**83.7**

Another approach for relevant content extraction is given in the next section.

5.3 RELEVANT CONTENT EXTRACTION BASED ON FEATURES OF A BLOG PAGE

A blog page consists of a number of blog posts comprising of the information related to the topic of the blog post. A blog post in general consists of three main parts:

- Title of blog post
- Post content
- Readers' comments on blog post

The research carried out in the past [54,55,56] did not give much importance to the blog title and the comments of the readers of the blogs for extracting relevant content. But after analysis, it is observed that along with the title, the comments also play a very important role for determining and extracting the relevant content. The snapshot of a sample blog page taken from "blogger.com" is shown in Fig. 5.9.

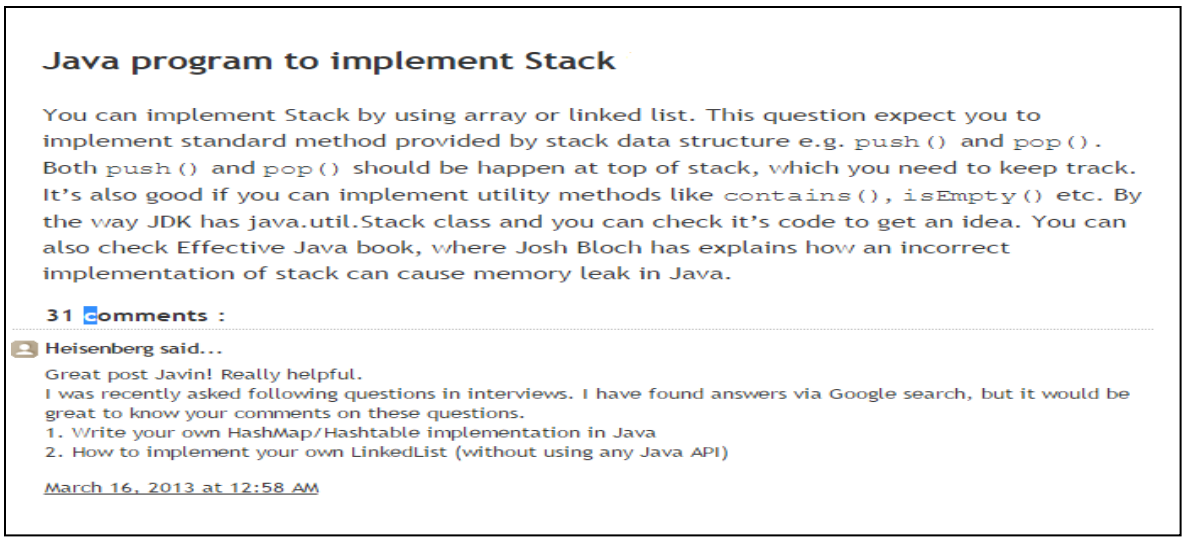


Fig. 5.9 Snapshot of a blog post with comments

See what actions the blog readers perform when reading a blog. He/she at first reads the title of the blog post and then reads the post content if the topic is of choice of interest. After reading the post content, the reader then gives his feedback by writing comments on it. Thus, these two parts i.e. title and comments play a major role in determining and extracting the relevant content of the blog page. The design of the proposed system for relevant content extraction [94] from a blog post is given in Fig. 5.10.

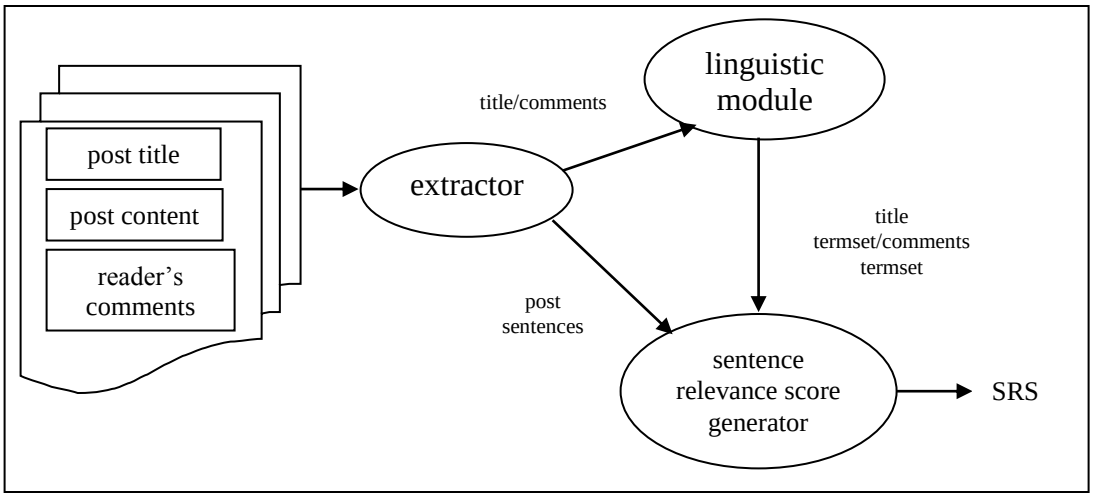


Fig. 5.10 Relevant content extraction based on features of a blog post

The proposed system comprises of the three major components:

(i) extractor module

(ii) linguistic module

(iii) sentence relevance score generator

5.3.1 extractor module

This module takes a blog post as input and separates the three parts of a blog post namely title, post sentences and the comments of the blog readers. The extracted title and comments are then given an input to the *linguistic module* and the post sentences are given to the *sentence relevance score generator* module.

5.3.2 linguistic module

This module receives the input from the *extractor* module and then performs the following functions on the title and comments:

- Tokenization
- Stemming
- Stop word elimination
- Lemmatization
- Normalization

After performing these functions, it generates the title termset and comments termset as follows:

(a). *Title termset:* This set represented by T that contains all the terms appearing in the title. It is represented by $T = \{t_1, t_2, t_3 \ldots, t_n\}$ where t_i is the i^{th} term in the title termset T.

(b) *Comments termset*: This set contains all the terms that appear in each comment present in the comment section C, where $C= \{C_1, C_2 \ldots C_n\}$ where $C_1, C_2 \ldots C_n$ represent the first comment, second comment and so on. Since, each comment contains one or more terms, so the *comments termset* for any comment C_i can be written as $C_iT= (c_it_1, c_it_2, c_it_3 \ldots c_it_n)$ where $c_it_1, c_it_2, c_it_3 \ldots c_it_n$ are first term, second term, so on of the comment C_i.

5.3.3 Sentence relevance score generator

This module takes title termset and comments termset for finding the corresponding Sentence relevance score of each sentence present in the blog post. This module follows the following steps given below to compute relevance score for each sentence in the blog post.

Step 1. This module considers the title termset and sentences of the blog post as input and then generates a matrix called Term-Sentence matrix (TSM). Each term of the title is represented corresponding to a row and each sentence of the post is represented corresponding to a column in the matrix. Each entry in the matrix is represented by TSM [i, j] where TSM is the *Term-sentence matrix* containing entries for the frequency of term T_i of title appearing in the sentence S_j of the blog post i.e. each entry of the matrix indicates the number of times the term of the title appears in corresponding sentence of the blog post (see Fig. 5.11).

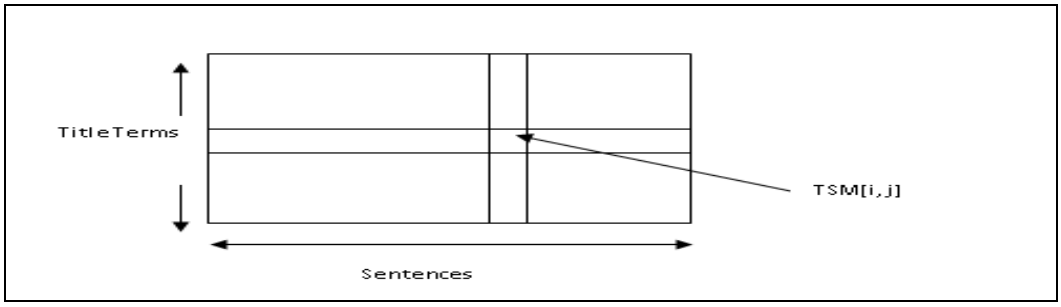


Fig. 5.11 Term-sentence matrix

Step 2. This module takes the termset of each comment given on the blog post and all the sentences $\{S_1, S_2 \ldots S_n\}$of the blog post as input. The frequency of each term in the termset of each comment in the blog post is computed using *Frequency calculator* [95]. Then the top N terms with higher frequency than the other terms are considered for generating matrix CSM[C(k,i), j] such that the N is the number of terms equal to the number of terms in the title of the post.

It is worth noting that if top N terms can be extracted from a comment, it indicates that the comment contains some term(s) as existing in the content of the blog post. If a comment does not contain any of these, it is considered irrelevant for the above task and

is simply ignored. Each term of the top N terms of a valid comment is represented by a row munber and each sentence of the post is represented by a column number of the matrix. Each entry in the matrix is represented by CSM[C(k,i), j] which gives the frequency of the is i^{th} term of k^{th} comment in j^{th} sentence of the blog post (see Fig. 5.12).

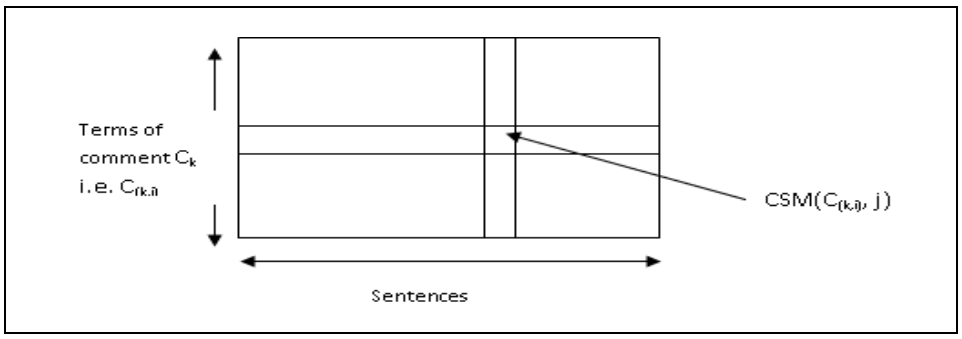


Fig.5.12 Comment-sentence matrix CSM

In this case, for each comment given on the blog post, one CSM matrix is generated. So, number of CSM matrices is equal to number of comments given by the readers of the blog in the comment section. For each comment C_k, the system generates a separate CSM matrix which contains the frequency of each comment term in each blog post sentence. For example, for two comments in the comment section, the system generates two CSMs.

As the result of these two steps, for each blog page a single TSM and multiple CSMs are generated. The SRS generator computes relevance score for each sentence that is called *Sentence relevance score (SRS)* using formula in relation to eq. 5.3.

$$SRS(Sj) = \alpha.(\sum_{i=1}^{n} TSM(i, j)) + \beta.(\sum_{i=1}^{n} CSM(C(k,i), j)) \quad \text{.....eq. 5.4}$$

where SRS (S_j) is Sentence Relevance Score for j^{th} post sentence;
TSM(i, j) is the frequency of i^{th} term of blog title in j^{th} post sentence;
CSM(C(k,i), j) contains the frequency of i^{th} term of k^{th} comment in j^{th} post sentence;
α, β are the weights given to TSM and CSM respectively and their value should be such that $\alpha+\beta=1$.

When the proposed blog crawler visits a blog page incrementally, it may be the case that one or more comments have been added to the post. So accordingly, the weights of α and β need to be adjusted dynamically, the explanation which is given below:

On a crawl: Let as the result of first crawl of a blog page, no comment is found for a post, then β=0 and α=1 in the eq. 5.4. This is because β is the weight associated with the comments.
When a comment is added to the post, the values are adjusted by using the formula:
Next value of α and β=current value of α and β + change in value of α and β,
where *change in value* may be positive or negative.

On recrawl : Let on a recrawl of the same blog page, one comment is found for the same post, then the valuet of α and β are updated as β= β+0.05 and α= α -0.05 in the eq. 5.4.

So, β=0.05 and α=0.95

On recrawl : Let on a recrawl of the same blog page, two comments are found for the same post, then β= β+0.05 and α= α -0.05 in the eq. 5.4.

So, β=0.1 and α=0.9

On recrawl : Let on a recrawl of the same blog page, four comments are found for the same post, then β= β+0.05 and α= α -0.05 in the eq. 5.4.

So, β=0.15 and α=0.85

On recrawl : Let on a recrawl of the same blog page, eight comments are found for the same post, then the weight of β= β+0.05 and α= α -0.05 in the eq. 5.4.

So, β=0.2 and α=0.8

Likewise, the values of α and β are adjusted for further recrawls of the same blog page when more and more comments are added to a post. It is worth noting that the weights are adjusted with increase in the number of comments equal to 2^i where the value of i is

increased by 1 on each recrawl. It is found that on further recrawls, when 256 comments are added, β=0.5 and α=0.5 i.e. their values become same. On further recrawls, their weights are kept unchanged. Using the formula, SRS is computed for each post sentence of the blog and then each sentence is ranked according to these scores assigned to them. Based on this evaluation, the top k sentences are selected thus forming relevant content for the blog post.

The next section gives discusses the experimental evaluation of the proposed approach to justify the proposed mechanism.

5.3.4 EXPERIMENTAL EVALUATION OF THE PROPOSED SYSTEM

The proposed system has been implemented on .Net framework using C# as the front end and MS-SQL as the back end. The system collected about twenty five blog pages from various blog sources on which the proposed approach of relevant content extraction was applied. The relevant content generated by applying the proposed approach on five sample blog pages is given in Appendix-4. To show the performance of proposed system, a sample blog page has been taken from the blog source *http://uanditalk.blogspot.in* shown in Fig. 5.13.

Fig. 5.13 Snapshot of the sample blog post

For analysis of the content generated by the proposed system, a model text has been used which has been generated by an expert in the same field. Relevant content generated by the proposed system has also been compared with the relevant content generated by following online tools.

a. www.freesummarizer.com

b. www.smmry.com

c. www.textcompactor.com

The content generated by the online tools available for the blog post shown in Fig. 5.13 is shown in Table 5.5.

Table 5.5 Content generated by the online tools

Using online tool (www.freesummarizer.com	The journey started with programmers who would write programs which somehow worked without giving any importance to readability of the program. The structured programming techniques did enforce some discipline on the programmers by way of shunning the use of 'goto' statements and encouraging the programmers to use 'easy to read and difficult to write' style of code statements choosing long and meaningful names for the variables, functions, procedures, modules etc. This technique worked good for hard core programmers who were able to write large and complex programs using structured programming techniques. Then why not write programs using objects which would be very natural way of creating useful software Comprising of interacting objects. Object Oriented Programming (OOP) is a paradigm shift in programming which defines, creates, and manipulates objects to develop reusable software.
Using online tool (www.smmry.com)	The journey started with programmers who would write programs which somehow worked without giving any importance to readability of the program.The languages li ke FORTRAN and BASIC neither enforced any discipline norwere the programmers trained to write user centric programs.The structured programming techniques did enforce some disciplin e on theprogrammers by way of shunning the use of 'goto' statements andencouraging the progr ammers to use 'easy to read and difficult to write' styleof code statements choosing long and me aningful names for the variables,functions, procedures, modules etc.In fact any real world progr am has to be a collection objects.A program about a University would involve objects like stud ents, professors,clerks, class rooms, books, chalk, mark sheets etc..
Using online tool (www.textcompactor.com)	The programming process has evolved through many phases. The journey started with programmers who would write programs which somehow worked without giving any importance to readability of the program. The major problem was that the programs were not maintainable. The structured programming techniques did enforce some discipline on the programmers by way of shunning the use of ‘goto’ statements and encouraging the programmers to use ‘easy to read and difficult to write’ style of code statements choosing long and meaningful names for the variables, functions, procedures, modules etc. This technique worked good for hard core programmers who were able to write large and complex programs using structured programming techniques.

The content generated by the human expert and the proposed approach is given in Table 5.6. The snapshot of implementation of the proposed system is shown in Fig. 5.14.

Table 5.6 Content generated by the human expert and the proposed approach

By Expert	The structured programming techniques did enforce some discipline on the programmers by way of shunning the use of 'goto' statements and encouraging the programmers to use 'easy to read and difficult to write' style of code statements choosing long and meaningful names for the variables, functions, procedures, modules etc. When we look around then we find that we are surrounded by nothing but objects only. Then why not write programs using objects which would be very natural way of creating useful software comprising of interacting objects. For example, in our day to day life we compose bigger objects from smaller objects. Object Oriented Programming (OOP) is a paradigm shift in programming which defines, creates, and manipulates objects to develop reusable software.
Using proposed approach	The structured programming techniques did enforce some discipline on the programmers by way of shunning the use of 'goto' statements and encouraging the programmers to use 'easy to read and difficult to write' style of code statements choosing long and meaningful names for the variables, functions, procedures, modules etc. Object Oriented Programming (OOP) is a paradigm shift in programming which defines, creates, and manipulates objects to develop reusable software. This technique worked good for hard core programmers who were able to write large and complex programs using structured programming techniques. Then why not write programs using objects which would be very natural way of creating useful software comprising of interacting objects. For example, in our day to day life we compose bigger objects from smaller objects.

The snapshot shows the relevant content generated as the result of the implementation of the proposed system.

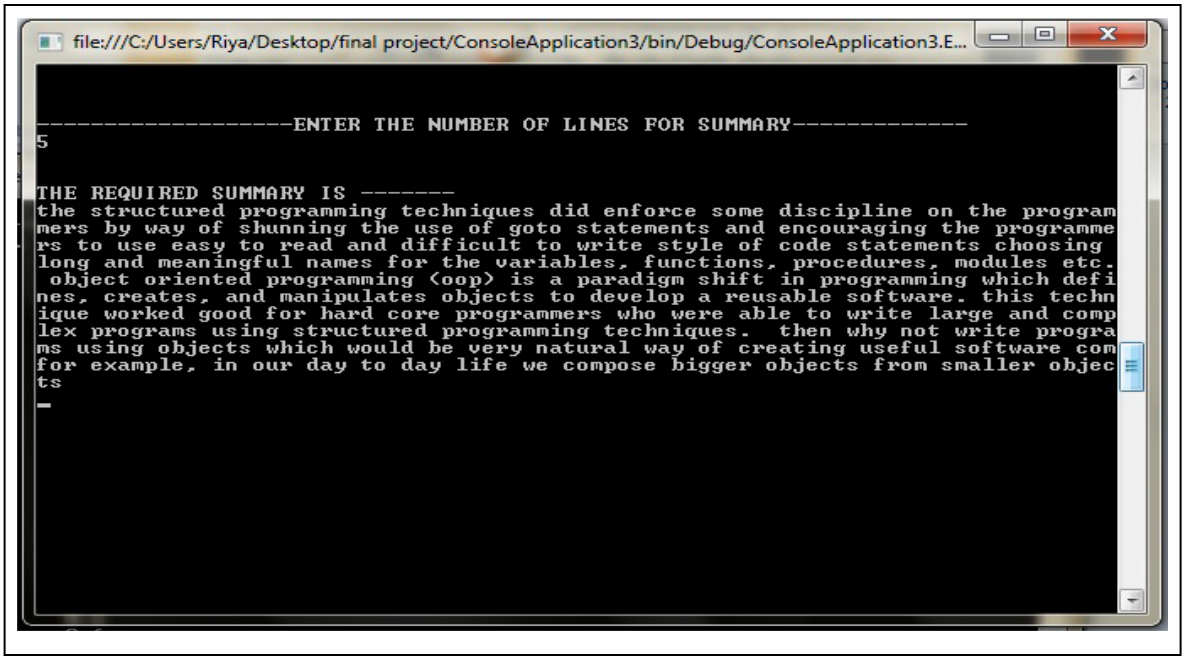


Fig. 5.14 Snapshot of the proposed system

For the performance evaluation of the content generated by the proposed approach, two performance metrics namely *precision* and *recall* that have been use as defined below:

1. *Precision* is defined as a fraction of number of common sentences that appear in both the content to be evaluated and the content given by expert over the total number of sentences appearing in the content to be evaluated.

Precision is given by:

$$P = N_c / N_s, \quad \text{.....eq. 5.5}$$

where N_c is the number of common sentences and N_s is the number of sentences in the content to be evaluated.

2. Recall is defined as a fraction of number of common sentences that appear in both the content to be evaluated and the content given by expert over the total number of sentences that appear in the content given by expert.

Recall is given by:

$$R = N_c / N_m, \quad \text{.....eq. 5.6}$$

where N_c is the number of common sentences and N_m is the number of sentences in the content given by expert.

The value of Precision and Recall computed for the proposed technique and the online tools using eq. 5.5 and eq. 5.6 is given in Table 5.7.

Table 5.7 Precision and Recall values of second approach

Approach/tools used for relevant content extraction	**Precision(in %)**	**Recall(in %)**
Proposed Approach	80	80
www.freesummarizer.com	60	60
www.smmry.com	40	40
www.textcompactor.com	20	20

A plot of precision and recall values for each of the techniques is shown in Fig. 5.15.

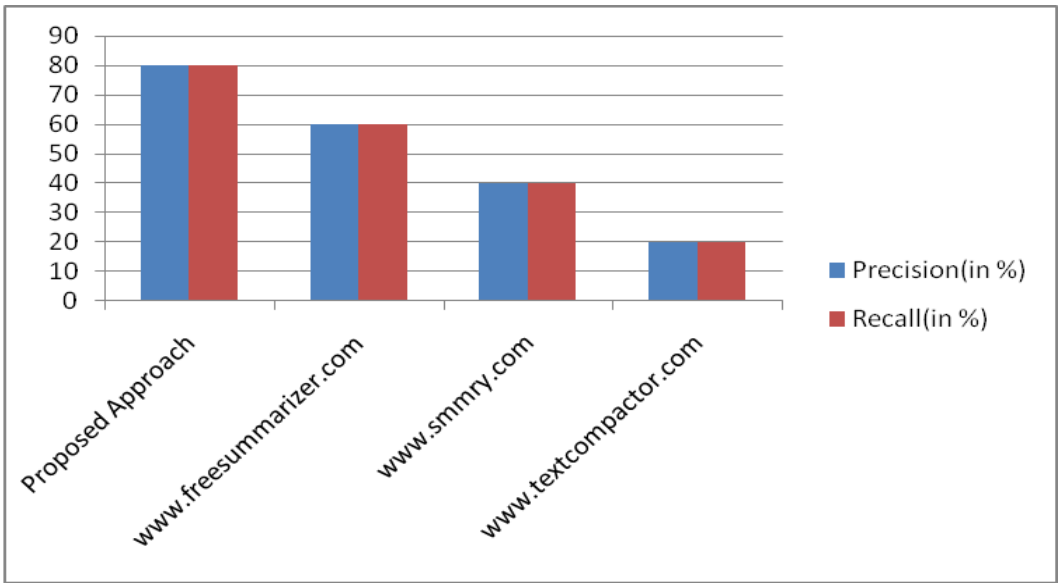


Fig. 5.15 Plot of P and R for second approach

On applying the proposed technique on the sample four blog pages (see Appendix-4 for the contents generated), it has been found that the precision and recall w.r.t. the relevant content generated by the expert lies between 80.5% to 85% and 79.5% to 83% respectively (see Table 5.8 for the values).

On analysis of the content generated by the proposed approach and the content given by the online tools show that the proposed approach works well and generates the content of good quality. The values of *precision* and *recall* are found to be higher for the content generated by the proposed system. Thus, the proposed approach works very well in extraction of relevant content from blog pages.

Table 5.8 Precision and recall values for blog pages

URL of the blog page	Precison (%)	Recall (%)
https://raygun.io/blog/2014/10/5-interesting-data-structures-algorithms/	**82**	**80**
http://www.ashishsharma.me/2011/08/java-garbage-collection-notes.html	**80.5**	**79.5**
http://blog.betafamily.com/2014/07/08/testing-techniques-black-white/	**83.8**	**82.5**
http://javahungry.blogspot.com/2013/04/scheduling-algorithm-first-come-first.html	**85**	**83**

The extracted relevant content of the blog is now indexed, thus producing an index in which searching is performed to respond to user's question. The question classification based indexing scheme is discussed in Chapter-6.

Chapter VI

A QUESTION CLASSIFICATION BASED INDEXING SCHEME FOR EFFICIENT QUESTION ANSWERING

6.1 GENERAL

For answering the questions posed by the user, there is a need to build an index of the relevant content. For efficient indexing of the relevant content of the blogs, the correct classification of the question [74,77,80] is required, which is done with respect to the type of answer expected by the user in response to his/her question. So, the index that is constructed consists of the relevant content on the basis of type of answer expected by the user. The searching operation is performed in this index based on Question classification, which results in efficient question answering.

6.2 PROPOSED SYSTEM FOR QUESTION CLASSIFICATION BASED INDEXING SCHEME

The proposed system [96] consists of the two main components:

i. Constructing question classified index
ii. Searching in question classified index

The detailed discussion on each of these is given in the following sections:

6.2.1 CONSTRUCTING QUESTION CLASSIFIED INDEX

The relevant blog content stored in *relevant content repository* is indexed by the *blog indexer* module which constructs an index. For indexing, at first correct classification of the question is needed and then the user's expectation in terms of expected answer type is recognized [80]. Index is then constructed on the basis of the expected answer types. The process of indexing is presented using the flowchart shown in Fig. 6.1.

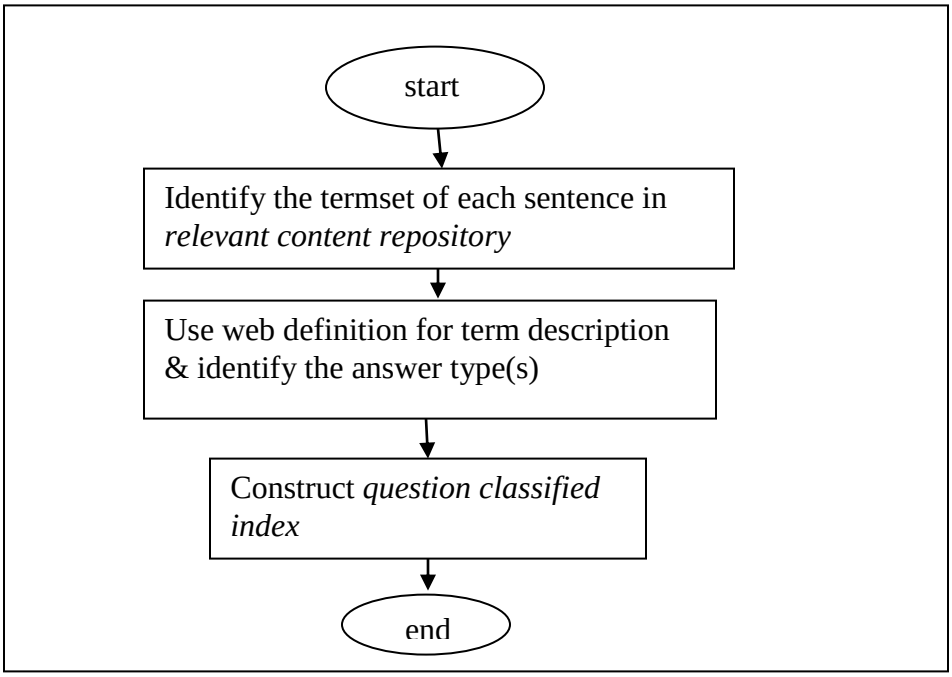


Fig. 6.1 Process of indexing

The details on indexing of relevant content and searching in the index for finding answer(s) to the questions is discussed in the steps given below:

i. The *blog indexer* module reads each sentence of the content stored in the *relevant content repository* and identifies its *termset* i.e. set of terms contained in the sentence. For termset identification, the following actions in the preprocessing are performed:

- Tokenization
- Stop word elimination
- Stemming
- Lemmatization
- Normalization

ii. The *Blog Indexer* module takes each term in the sentence termset and then the system uses the *Web definition* [90]. *Web definition* can be used by including "define" in front of a term, the search engine provides a description of the term by using online information sources like dictionaries [91] and/or WordNet [78,92]. Analyzing the Web definition, the appropriate answer type

under which the term can be categorized is identified. See Table 6.1 for appropriate answer type(s) corresponding to some terms. The first column of the table gives the term description obtained using Web definitions and the second column gives the corresponding answer types.

iii. To form a *question classified index*, the terms in the termset are indexed under the appropriate answer types identified in Step2 along with the Sentence Id (Sid) and Page Id (Pid) in which the term exists. The structure of the index formed is shown in Table 6.2. Thus the *Question classified index* is formed that contains mapping of the terms to their answer types, the sentence id (Sid) & the page id (Pid) of the page containing relevant content in which they appear. The search for the user's question is performed by the *searcher* module in this index.

Table 6.1 Mapping term description to Answer type(s)

Term description for the tems obtained using Web definition	**Answer type**
Government-Agency, Agency, Company, Airline, University, Institute, Sports-Team	Organization
Leader, Father, Mother , Sister, Brother, King, Queen, Emperor, Name	Person
City, Country, State, Territory, Mountain, Island, Street, Land, planet, river, ocean	Location
Full form/short form	Abbreviation
Quantity, Distance, weight, size, temperature, speed, percent, code, count, period, money, currency	Number
Procedure, Method, process	Process
Day, Days of the week	Day
AM, PM	Time
Year	Year
Months of the year like January etc.	Month
Date	Date
Concept, object, protocol, definition	Description
Animal, body, color, disease, medicine, event, food, instrument, language, letter, plant, product, religion, sport, substance, symbol, technique, vehicle	Entity
Explanation	Reason

The question classified index is shown in Table 6.2.

Table 6.2 Question Classified Index

Answer type	Terms	Sid and Pid
Person	Batsman	s5,p7
Person	President	s1,p9
Location	USA	s4,p7
Process	Scheduling	s3,p4
Day	Sunday	s1,p7
Month	December	(s4,p7), (s1,p9)
Abbreviation	ADT	(s5,p1), (s1,p2),(s4,p2)
Abbreviation	WHO	(s5,p4), (s1,p3)
Organization	Corporate	s5,p5
Description	concept	(s2,p9), (s4,p9)

The algorithm of the *blog indexer* module is given in Fig. 6.2.

```
Algorithm blog indexer ()
  {
        While (!Empty TermSet)
        {
         For each Term in TermSet
        {
        Step 1. Use web definition to obtain term description;
              2. Analyze the term description for answer type(s)
             3.  Index the Term
        }
       }
}
```

Fig. 6.2 Algorithm: *blog indexer*

Snapshot of the Web definition of the term “Continent” is shown in Fig. 6.3.

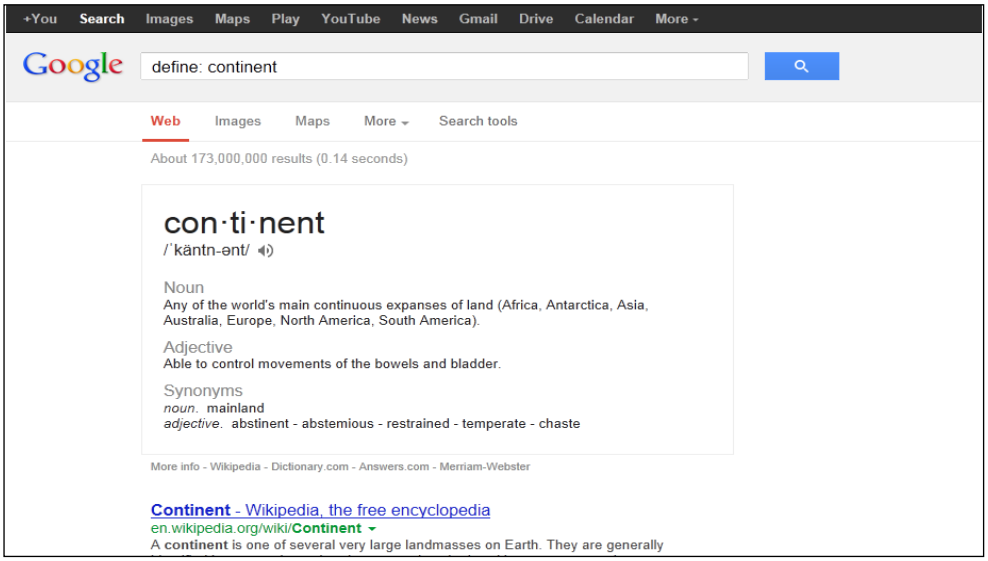


Fig. 6.3 Snapshot of web definition of "Continent"

Web definition for the term "continent" is "*Any of the world's main continuous expanses of land (Africa, Antarctica, Asia, Australia, Europe, North America, and South America)*". Since, the *Web definition* of "continent" contains description about *land*, its identified answer type is *Location*.

Hence, the term *"Continent"* is indexed under the answer type *"Location"*. The index contains the Sid of the sentence and the Pid of the particular page in which *continent* appears. Another term *Batsman*, is indexed under *person* answer type along with the Sid and Pid in which it appears.

6.2.2 SEARCHING IN QUESTION CLASSIFIED INDEX

The question asked by the user on PQAS interface is supplied to the Question Classifier module that identifies the class of the question. The algorithm of *Question classifier* module is given in Fig. 6.4.

The proposed system of Question answering restricts the question to start with *Who, What, Where, When, Which, Why and How*, for identifying the question class. The question class indicates what the user expects from the system in response to his/her

question. After identifying the question class, the rest of the question is converted into query in form of set of terms using *preprocessing*.

```
Algorithm Question classifier ( )
{
Step1. Separate the first term of the question and identify the question class
    2. Convert the remaining question into a query in form of set of terms using preprocessing
}
```

Fig. 6.4 Algorithm: *Question classifier*

Some examples of question classification are shown in Table 6.3**.**

Table 6.3 Examples of question classification

Question	**Question class**	**Query terms**
Who discovered stem cell	Who	Discover, Stem, Cell
Which is the coldest place in the world	Which	Cold, Place, World
When did titanic sink	When	Titanic, Sink
How is the president of USA elected	How	President, USA, Elect
Why did Hitler kill himself	Why	Hitler, Kill

As shown in Table 6.3, for the question, *"When did titanic sink"*, "*When" is* identified as the question class and *Titanic, Sink* forms a query. The question class and the query terms are taken as input by the searcher module of the proposed system. The module maps the question class into appropriate answer type(s) by using Table 6.4. After that the Question Classified index given in Table 6.2, is looked upon for the terms in the query corresponding to the identified answer type. The Sid and the Pid corresponding to the terms are extracted. The sentences are given as answers to the user's question.

For example, if the user's question is *"Who is the President of USA*", the Question Class is "who" i.e. the user expects name of the *Person* and/or *Organization* in the answer for the query terms *President* and *USA.*

Table 6.4 Identifying Answer types

Question class	Answer type
Who	Person, Organization
Where	Location
What	Number, Definition, Procedure, Abbreviation, Organization, Person, Year, Month, Day, Time, Location, entity, date
When	Time, Year, Day, Month, date
Which	Person, Location, Month, Time, Year, Day, organization, entity, date
Why	Reason
How	Process

The answers identified by the searcher module are given to the user in response to his/her question.

6.3 PERFORMANCE EVALUATION OF THE PROPOSED SYSTEM

The proposed approach has been implemented on .Net framework with MS-SQL as the backend. The snapshot of the system is shown in Fig. 6.5.

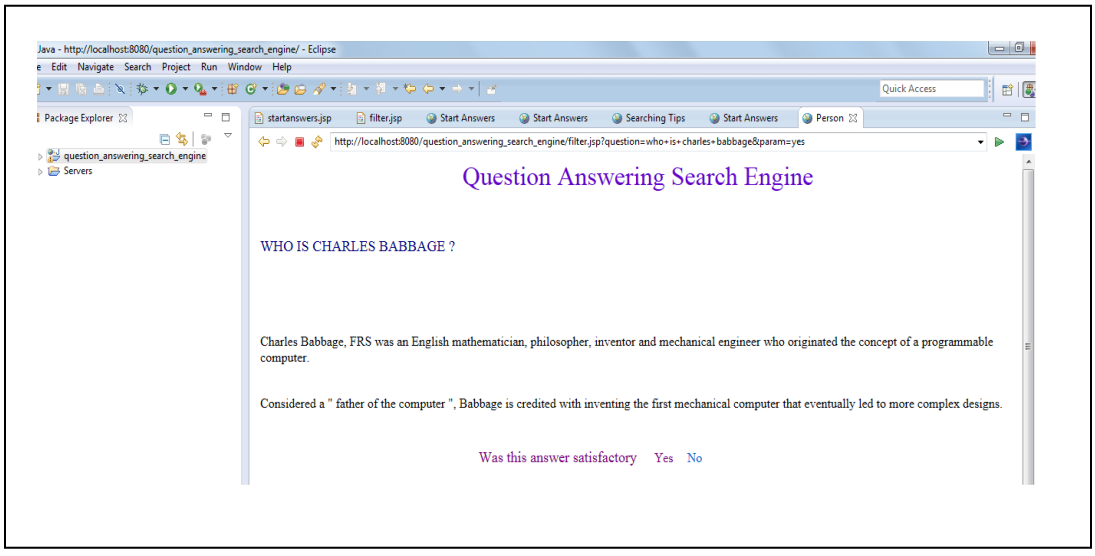


Fig. 6.5 Snapshot of the proposed system

A survey has been conducted in which a questionnaire was circulated that consisted of eleven questions in two engineering colleges namely Echelon Institute of Technology, Faridabad and YMCAUST, Faridabad among 60 participants. For each question class, an Answer Type has been given along with a number of factors related to that answer type. The participants were then asked to choose the factors relevant to each answer type. The survey and its result are given in Appendix-5.

Table 6.5 Most relevant factors for the answer types

Answer type	**Most relevant factors**
Location	i. Where it is located ii. Whether it is a city, state, country iii. It is nearby to iv. Its historical significance
Person	i. Name ii. Education iii. Birth place/native place iv. His/her contribution to the country as a leader, employer, scientist etc. v. When he/she was born/died
Organization	i. Name ii. Owner iii. Location iv. Year in which it is established v. Its product
Day	i. It is 1^{st} , 2^{nd} ,….or 7^{th} day of the week ii. It comes after iii. It may be known for something in India/other countries iv. Its significance-religious, historical v. Rahukaal of that day
Month	i. It is 1^{st} , 2^{nd} ,….or 12^{th} month of the year

	ii. Number of days in that month iii. It may be known for something in India/other countries iv. Its significance-religious, historical
Number	i. It is a number/ amount/ count etc. ii. How it is formed iii. What it actually signifies
Year	i. What event took place in the year ii. Its significance in the history iii. Total number of days
Abbreviation	i. Its full form is ii. It is short form of iii. Its basic concept iv. If an organization-its location, owner, year and for what it works for
Description	i. Its definition ii. Its concept iii. Its use as a noun, verb, adjective, its plural, its singular or some other form iv. Its synonyms/ hypernyms/ antonyms/ meronymns etc.
How	i. Process/procedure
Why	i. Reason

It has been observed that some of the factors were found to be relevant by majority of the participants and so taken into consideration and others were found to be less relevant and were ignored. The threshold value of 85% is considered in the present study. Hence, the most relevant factors for the answer types were identified as shown in Table 6.5.

6.3.1 CALCULATING ANSWER RELEVANCE SCORE

To analyze the proposed system, 45 questions in total were prepared, with questions belonging to each answer type. Then the proposed system was analyzed for its responses. The factors present in the responses were compared with the identified relevant factors. A

relevance score is calculated for the answers given by the system as a whole, using formula below:

Answer relevance score (ARS) in % =RF/TF*100……..eq. 6.1

where,

RF is the number of relevant factors existing in the answers and

TF is the total no. of relevant factors identified for that answer type.

Table 6.6 show some sample questions, belonging to "who" Question class along with the answer type(s), total number of relevant factors identified as the result of the survey (TF) and the number of relevant factors returned by the system under analysis (RF). The last column shows the calculated score (ARS). Also, average ARS is calculated.

Table 6.6 ARS for questions starting with "who" Question class

S.No.	Questions	Answer type	No. of Relevant Factors Returned (RF)	Total No. of Relevant Factors (TF)	ARS = RF/TF * 100 (in %)
Q1	Who is Mother Teresa?	Person	5	5	100.0
Q2	Who is the inventor of Telephone?	Person	3	5	60.0
Q3	Who is Charles Babbage?	Person	3	5	60.0
Q4	Who is Mahatma Gandhi?	Person	3	5	60.0
Average ARS					70.0

Similarly, for the questions belonging to other question classes i.e. where, what, when and which, the Answer relevance score (ARS) and the average ARS is computed as shown in Table 6.7 to 6.10.

Table 6.7 ARS for questions starting with "where" Question class

S.No.	Questions	Answer type	No. of Relevant Factors Returned (RF)	Total No. of Relevant Factors (TF)	ARS = RF/TF * 100 (in %)
Q1	Where is Delhi?	Location	4	4	100.0
Q2	Where is Faridabad?	Location	4	4	100.0

Q3	Where are Himalayas?	Location	3	4	75.0
Q4	Where is pacific ocean?	Location	3	4	75.0
Average ARS					87.5

Table 6.8 ARS for questions starting with "what" Question class

S.No.	Questions	Answer type	No. of Relevant Factors Returned (RF)	Total No. of Relevant Factors (TF)	ARS = RF/TF * 100 (in %)
Q1	What is the language?	Description	4	4	100.0
Q2	What is a variable?	Description	3	4	75.0
Q3	What is the abbreviation of phd?	Abbreviation	2	4	50.0
Q4	What is the network interface card?	Description	3	4	75.0
Average ARS					75.0

Table 6.9 ARS for questions starting with "when" Question class

S.No.	Questions	Answer type	No. of Relevant Factors Returned (RF)	Total No. of Relevant Factors (TF)	ARS = RF/TF * 100 (in %)
Q1	When is Diwali?	Time	1	1	100.0
Q2	When is Mahatma Gandhi born?	Date	1	1	100.0
Q3	When is Mahatma Gandhi died?	Date	1	1	100.0
Average ARS					100.0

Table 6.10 ARS for questions starting with "which" Question class

S.No.	Questions	Answer type	No. of Relevant Factors Returned (RF)	Total No. of Relevant Factors (TF)	ARS = RF/TF * 100 (in %)
Q1	Which organization is responsible for allocating IP addresses?	Organization	4	5	80.0
Q2	Which is the last day of the week?	Day	3	5	60.0

Q3	Which is the highest mountain range?	Entity	1	1	100
Q4	Which planet is closest to Earth?	Entity	1	1	100
Average ARS					85.0

The screenshots of the answers returned by the Question answering system are given in Appendix-6. Average ARS of the question classes is shown in Table 6.11.

Table 6.11 Average ARS for each Answer type

Question class	Average ARS (%)
Who	70.0
Where	87.5
What	75.0
When	100.0
Which	85.0

Graphs have been plotted for showing average ARS for the above answer types in Fig. 6.6.

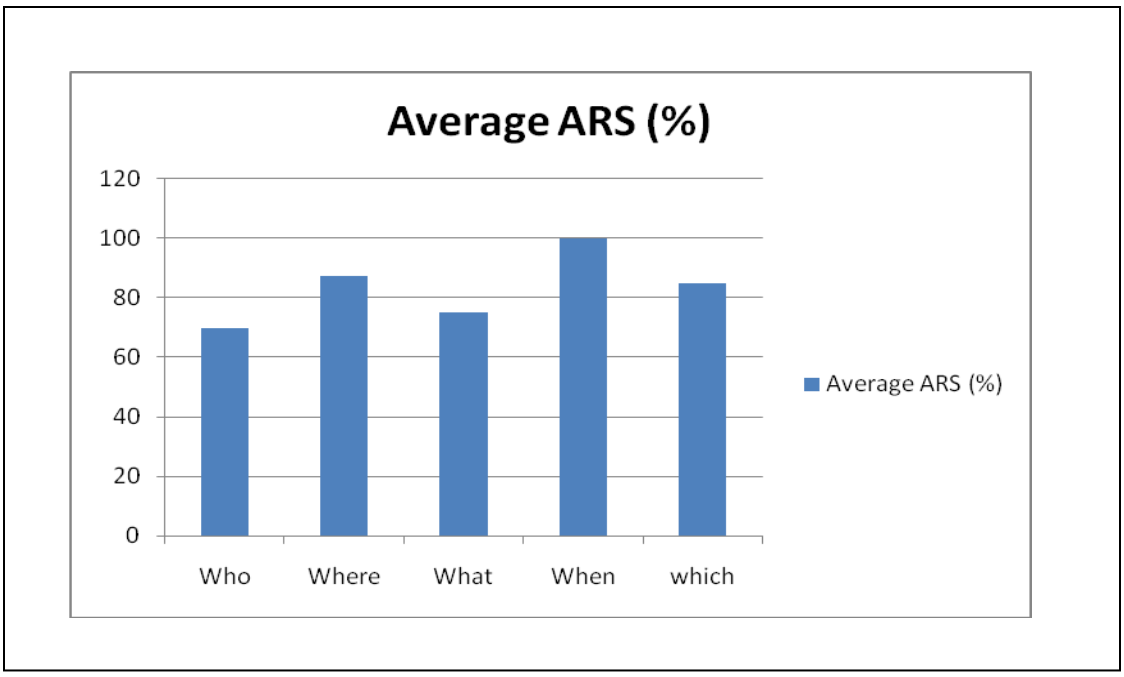


Fig. 6.6 Plot of average ARS

It is worth noting that the question starting with “why” is dealt with a slight difference i.e. by looking up for the answer(s) only in the *terms* column in the *Question classified index* and returning the corresponding sentences. Also, since the answer to such question needs a reason, so there are no possible *relevance factors* on the basis of which the answer can be evaluated. So, in this work, for such questions ARS cannot be computed.

On overall analysis, it has been observed that average Answer relevance score (ARS) of the answers obtained from the proposed system was found in the range from 70.0% to 100.0% which shows that the proposed system indexes the relevant content well. The system uses efficient techniques for both indexing and finding the relevant answers, and showing the high Answer relevance score. The set of sentences identified by the searcher module as the answers are given to the user.

A mechanism to improve the quality of the blog repository using popularity features of blog post is given in next chapter i.e. Chapter-7.

Chapter VII

A MECHANISM TO IMPROVE THE QUALITY OF THE BLOG REPOSITORY USING POPULARITY FEATURES OF BLOG POST

7.1 GENERAL

When the number of blog posts in the *page repository* increases to a huge extent, a need to filter some of them on some basis is felt. It is felt further that the quality of answers which is directly dependent upon the quality of blog repository can be significantly improved with the improvement in the quality of the blog repository.

In this work, the downloaded blog posts are assigned blog scores based on multiple criterions discussed in the following sections. The repository of the blog posts is built by including high scoring blog posts [104] and discarding the blog posts which have scored blog scores below threshold value. This activity ensures that the blog repository contains quality blog posts.

7.2 PROPOSED APPOACH FOR ASSIGNING BLOG SCORES

A blog reader reads a blog post and provides his/her feedback in many ways. The ways in which the blog reader provides his feedback, along with his/her interactions with the blogger and other visitors of the blog post plays a vital role. The given feedback which may be positive or negative is also important while computing rank of a blog post. After analyzing the blog posts, it is found that there are seven parameters [97] that are important for deciding rank of a blog page as given below:

i. Number of subscribers
ii. Number of related comments
iii. Number of votes
iv. Number of likes
v. Rating

vi. Sharing

vii. Presence of blogger information

A brief description of the above parameters is given below:

i. Number of subscribers: This parameter gives the total number of persons that have subscribed to the web blog through RSS feeds. When a reader subscribes to RSS Feed, the recent blog post/modifications in the blog post is automatically delivered to his mailbox. The updates appear just like the emails appearing in the mailbox, latest content on the top, with the headline and the first few lines of the post.

ii. Number of related comments: The blog readers go through the blogs and then can express their views or opinions on the blog posts by writing comments in its comment section. This is a way to provide feedback on the blog content and is also a way to interact with the blogger or other blog readers. The readers can also mention other blogger's posts' in his/her comments, which may be related or not related to the original blog posts. The related comments may play a crucial role in deciding the rank of blog post. The unrelated comments may be ignored. The comments that have some common terms with the blog text are considered as related and are used as a parameter for blog ranking.

iii. Likes, Voting, Rating, Sharing: These are some ways to review to a blog. To like a blog post, user can hit a like, rate it on the scale of 5, can vote or can share it on various social networking sites like facebook, twitter, google+ etc.

iv. Presence of Blogger information: A blog post may consists of the blogger information i.e. the name of the person who has written the blog, designation and the official address of the person. The presence of such information can be considered for authenticating the blog and differentiating it from other blogs in terms of ranking. To calculate score of a blog page the formula is given in equation 7.1.

$$Bscore=(F1+F2+Max(F3,F4,F5)+F6)/4+\Delta F7 \quad \text{eq. 7.1}$$

where F1is the number of subscribers of the blog post,

F2 is the number of related comments.

F3, F4, F5 are number of votes, number of likes and rates respectively,

F6 is the number of shares,

ΔF7 is a unit step function that indicates the presence/absence of blogger information.

Formally, ΔF7= 0, if blogger information is not present

=1, if blogger information is present

For incorporating the proposed method of blog ranking, the architecture of *blog crawler* has been modified as shown in Fig. 7.1, with the modification shown using dotted rectangle in blue.

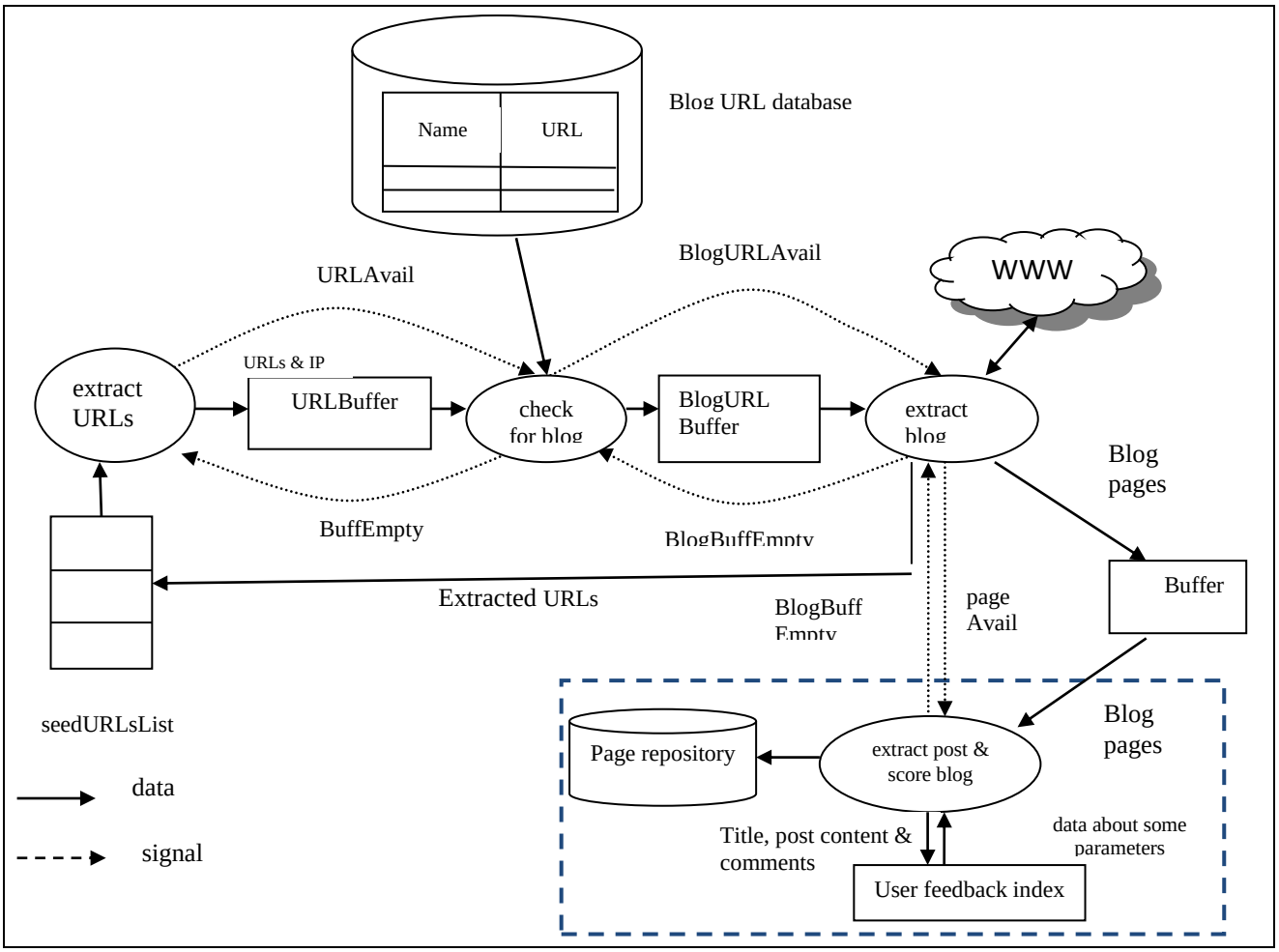


Fig. 7.1 Improved design of blog crawler

The *extract post* module of the *blog crawler* extracts the title of each post, its content and the comments written in its comment section and store them in form of a text pages. The module also extracts the data against the parameters of the blog post given below:

i. Number of subscribers
ii. Number of related comments
iii. Number of votes
iv. Number of likes
v. Rating
vi. Sharing
vii. Presence of blogger information

The algorithm for the modified working of *extract post & score blog* module is given in Fig. 7.2.

```
Algorithm extract post & score blog ( ){
do{
wait(pageAvail){
do{
Step1.fetch blog pages from the buffer
    2.for each blog post that exists in the blog page
      2.1 extract the title of the blog post
      2.2 extract the posts' content
      2.3 extract the comments on the blog post
      2.4 store them as a single text page in the page repository
      2.5 Retrieve the value for each parameter and store in variables F1 to F7
          F1= number of subscribers
          F2= number of related comments
          F3= number of votes
          F4= number of likes
          F5= rates
          F6= number of shares
          F7= 1 for presence of Blogger information or 0 for the absence of Blogger information
    2.6 Compute Bscore=(F1+F2+Max(F3,F4,F5)+F6)/4+ΔF7
    2.7 return Bscore. } while(the buffer is not empty);}
 signal(buffEmpty)  } forever  }
```

Fig. 7.2 Algorithm for blog scoring

By using this algorithm, the module computes the blog score of each blog post. The values of parameters fetched by *extract post & score blog* module is stored along with the *blog post id* for each blog post in *user feedback index*, the structure of which is shown in Table 7.1.

Table 7.1 User feedback index

	F1	F2	F3	F4	F5	F6	F7
Blog post Id	#subs	#related comments	#votes	#likes	#rating	#sharing	blogger info

The *user feedback index* comprises of Blog post id and the value of each of the parameters used for blog scoring given by F1 to F7.

7.2.1 EXAMPLE OF BLOG SCORING

Consider some blog posts shown in Appendix-7. They are assigned ids B1 to B5. The values of parameters for these blog posts with id B1 to B5 is given in Table 7.2 and the module utilizes this data to compute the blog score for each post, using the eq. 7.1.

Table 7.2 Value of parameters for blog posts

	F1	**F2**	**F3**	**F4**	**F5**	**F6**	**F7**
Blog post Id	**#subs**	**#related comments**	**#votes**	**#likes**	**#rating**	**#sharing**	**blogger info**
B1	50	14	0	161	0	2	1
B2	19	7	0	0	0	499	0
B3	2	0	0	1	0	1	0
B4	0	2	0	0	0	0	0
B5	1	4	0	0	0	0	1

Let us now stepwise calculate the blog score of the blog posts B1 to B5.

For B1, using the formula given in eq. 7.1, the blog score is

$$Bscore(B1)= (50+14+Max(0,161,0)+2)/4+1=226.5$$

Similarly, the Bscore is calculated for the blog posts with Blog Ids B2 to B5.

For B2, using the formula given in eq. 7.1, the blog score is

Bscore(B2)= (19+7+Max(0,0,0)+499)/4+0=131.25

For B3, using the formula given in eq. 7.1, the blog score is

Bscore(B3)= (2+0+Max(0,1,0)+1)/4+0=1

For B4, using the formula given in eq. 7.1, the blog score is

Bscore(B4)= (0+2+Max(0,0,0)+0)/4+0=0.5

For B5, using the formula given in eq. 7.1, the blog score is

Bscore(B5)= (1+4+Max(0,0,0)+0)/4+1=2.25

All the blog scores are shown in Table 7.3.

Table 7.3 Blog scores of blog posts

Blog post id	Bscore
B1	226.5
B2	131.25
B3	1
B4	0.5
B5	2.25

It may be observed that the blog page B1 gets the highest score 226.5 and the blog page B4 receives the lowest score i.e. 0.5. The *extract post & score blog* module may keep or ignore a blog post depending upon a threshold value, whose computation is given below.

7.2.2 THRESHOLD VALUE

When the number of crawled blog pages increase in huge amount, a need to filter the blog posts among all has been felt. For this, a threshold t_{bs} has been applied on the *blog score* i.e. blog posts with blog score less than t_{bs} are ignored and those with higher *blog score* are stored in *page repository* for further processing. For example, the initial value of threshold (t_{bs}) may be computed such that for any blog post, each parameter should have a non-zero value. Considering it, the initial threshold comes out to be 2.0.

$$t_{bs}= (1+1+1+1)/4+1= 2.0.$$

Further the *extract post* module analyzes each set of ten blog posts after scoring, and computes the *average blog score.* If for each next set, the *average blog score* is found to be increasing, then the *extract post* & *score blog* module increases the threshold t_{bs} to 2^i

where i increase by 1 for each next set which have shown the increasing bscore. The process is carried similarly and say a point in time reaches when i=5 and t_{bs} =32 (upper threshold), all the posts with Bscore less than 32.0 are ignored and only those with higher score are stored in *page repository*, thereby maintaining the quality of the blog repository. The reverse process will follow with decrease in Bscore for the next set taken, such that the value of i will start decreasing by one for each next set, up to the lower threshold value of t_{bs}. The graph for the same is shown in Fig. 7.3.

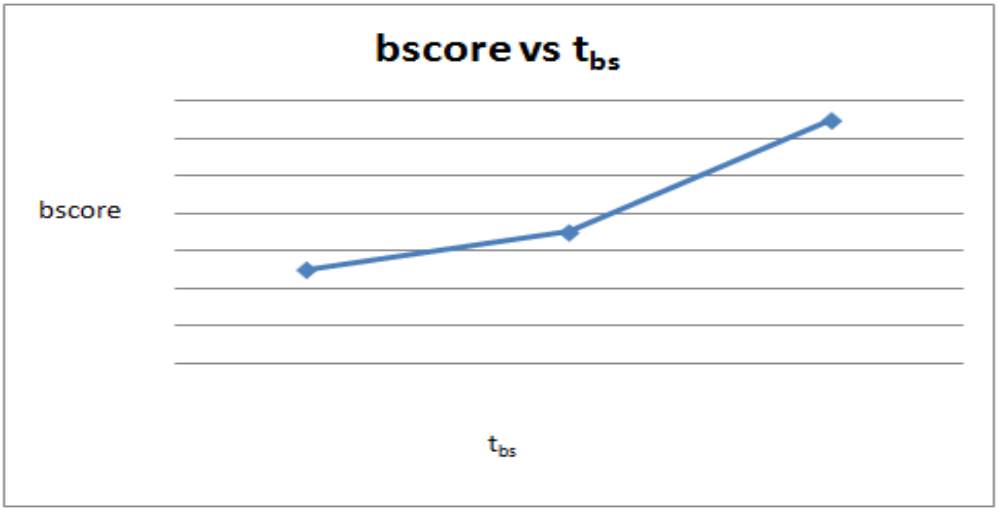


Fig. 7.3 Graph for Bscore vs t_{bs}

The increasing and decreasing behavior of threshold with increase and decrease in bscore is depicted in the graph.

7.3 IMPLEMENTATION OF THE PROPOSED SYSTEM

To analyze the proposed work, various experiments have been conducted. The proposed approach has been implemented in JDK 1.8. For the analysis of the proposed system, about 100 blog pages have been collected from different blog sites covering variety of topics and the proposed approach of ranking has been applied on them. A number of sample queries have been fired and the result pages obtained thereof are analyzed. The snapshot for sample query “musical instruments” has been shown in Fig. 7.4. It shows the result pages returned by the system after the approach of ranking has been applied and the pages are arranged according to their blog scores. It can be seen from the snapshot

that for each result returned by the system, the value of the parameter used for ranking is provided.

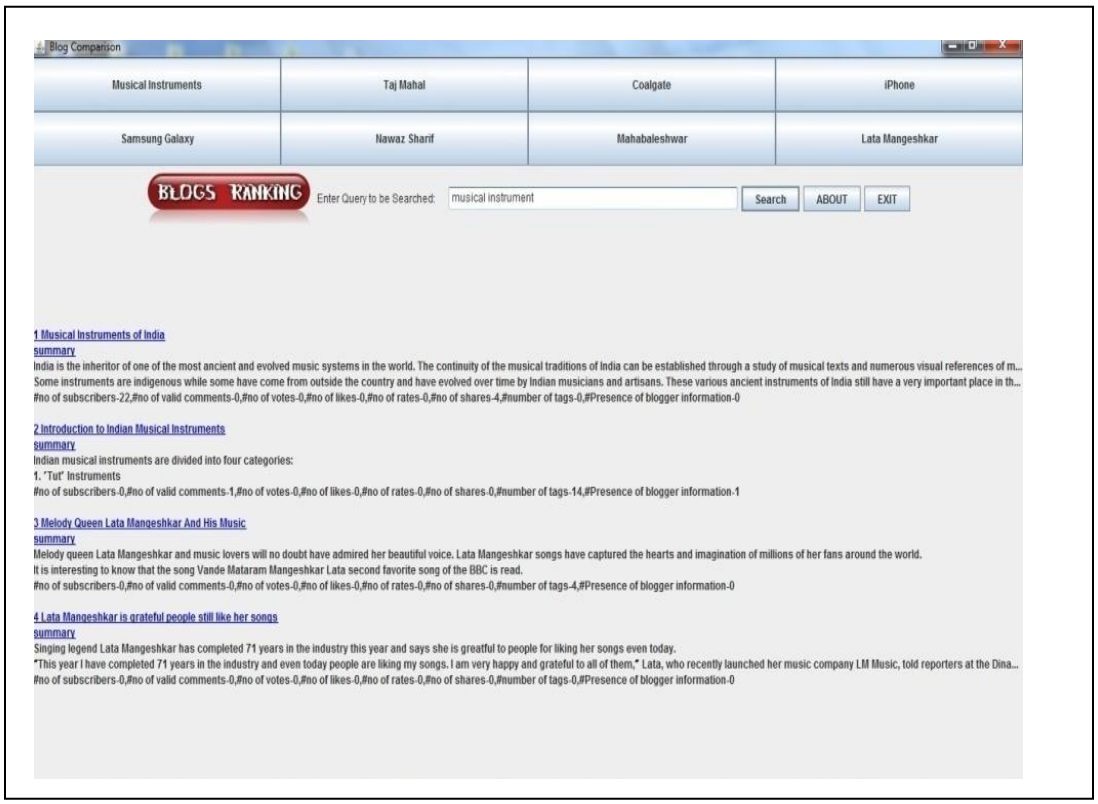


Fig. 7.4 Results for the query "musical instrument"

For improvement in the response time of PQAS, the system for the prediction of user's next question has been proposed in Chapter-8.

Chapter VIII

IMPROVEMENT IN RESPONSE TIME OF QUESTION ANSWERING SYSTEM BY PREDICTING USER'S NEXT QUESTION

8.1 GENERAL

In a Question Answering system, the user submits a question and waits for the answer in response. If the system is capable of predicting the user's future interest in terms of the next question [102]; its performance may improve greatly. The proposed system predicts the next prospective question, searches its answers and stores the question-answer pair in its database. When the user enters the next question and if the question matches with already predicted questions, the answer may be provided instantly instead of looking up in the index.

8.2 PROPOSED APPOACH TO DETERMINE USERS' NEXT PROSPECTIVE QUESTION

A novel approach to predict the next prospective question [98] expected from the user is presented in this chapter. If an initial question is given to the QA system, the system tries to predict, what the user may ask next. For this purpose, a system for the next question prediction [50,51,52] is being proposed that integrates with the existing QA system. It uses *Association rule mining* [49,82,83,112] which was initially used to find the association of one object/event with other object/event. For example, a query such as "does Tata sky airs Investigative discovery" is likely to be followed by "at what time" and "what is the channel number".

Association Rule is an implication expression of the form $X \rightarrow Y$, where X and Y are itemsets. It means that if X exists in a user's query, then Y also exists. *Support* is defined as the fraction of transactions that contain both X and Y. *Confidence* measures how often items in Y appear in transactions that contain X.

Formally, support=(X U Y).count/n…….……eq. 8.1

confidence=(X U Y).count/X.count……..eq. 8.2

where n is the total number of transactions, (X U Y).count is the number of transactions that contain both X and Y and X.count is the number of transactions that contain X. The problem of finding associations between the questions asked by the users of the system can be made similar to finding associations among itemsets in transaction databases.

As discussed in the previous chapters, the user enters his/her question on the interface of the QA system i.e. PQAS *Interface*. This question is classified by the *Question classifier* module. The module classifies the questions according to their question type and then converts the rest of the question into query. For the query, a search is conducted by the *searcher* module in *Question classified index*. It is proposed that the users' questions are stored in the *Questions log* along with the IP address from which the question has been asked, the Question ID (QID), date and time of questioning i.e. the interactions between the user and the PQAS interface are maintained. This information is maintained by PQAS interface in form of records in a file termed as *Questions log*. The system tries to predict users' future questions based on his/her current interaction with the system which indicates what they are interested in. From the *Questions log*, the user sessions are extracted. Based on the user sessions, the system tries to predict the next question. The proposed system of *prospective question answering* is shown in Fig. 8.1.

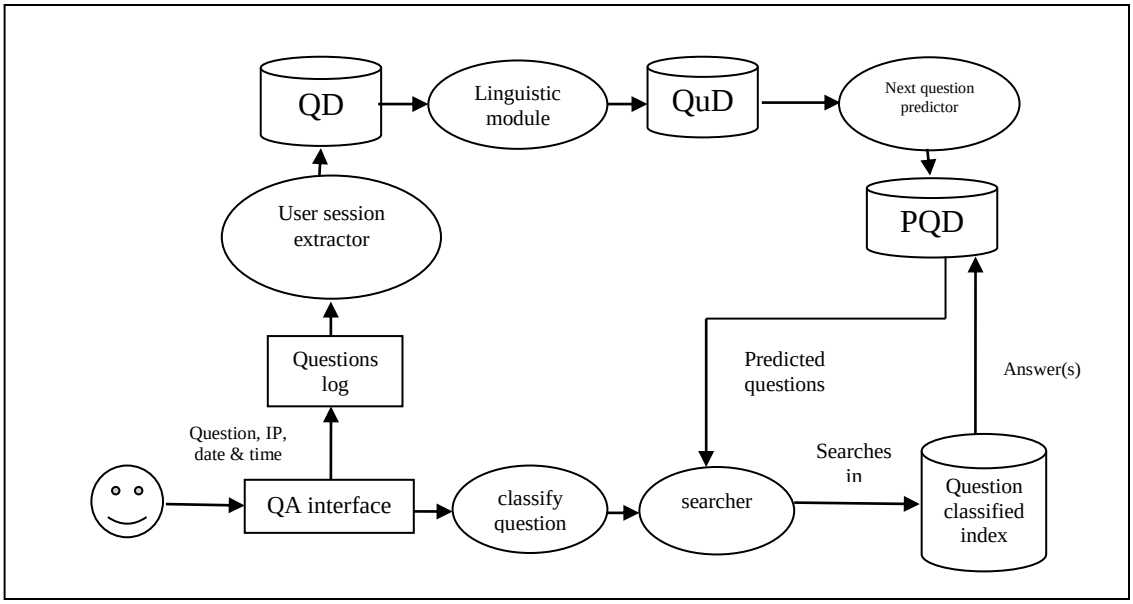


Fig. 8.1 Proposed system for *Next Question Prediction*

The major modules of the proposed system are discussed as follows:

8.2.1 User session extractor

In this work, a *questions log* is maintained that contains large number of questions asked by the users over a period of time say T units. The *session extractor* extracts *n* number of sessions from the *Questions log* comprising of *m* questions after every *t* units of time interval each represented by two sets S and Q respectively, as given below:

$S= \{S_1, S_2,.... S_n\}$ are the user sessions and

$Q=\{QID_1, QID_2,.... QID_m\}$ are the questions, each identified by a unique QID.

It may be that the user has provided a number of questions belonging to different question classes i.e. *who, when, where, what, which, why* and *how*, thus the module accordingly separates the questions and stores them into a database called *Question databse* QD. QD stores the questions corresponding to each question class given by QC. The structure of QD is shown in Fig. 8.2.

Question class(QC)	Questions($QID_1, QID_2,.... QID_m$)

Fig. 8.2 Structure of Question database QD

8.2.2 Linguistic module

This module preprocesses each question Q stored in QD and converts it into a query where the query is a set of terms contained in the Q. *Queries database* QuD is formed corresponding to QD. QuD contains QC and the queries $Qu =\{Qu_1, Qu_2,\ldots Qu_m\}$ corresponding to each question contained in the QD. The structure of QuD is given in Fig. 8.3.

Question class(QC)	Queries(Qu_1, Qu_2,….Qu_m)

Fig. 8.3 Structure of Queries database QuD

8.2.3 Next question predictor

In this work, each question entered by the user is considered as a transaction and the terms in the query are considered as the set of items associated with in the transaction. Then the *association rules* are mined from the available data using *Apriori* algorithm [83] which comprises of the following two steps:

i. Find the itemsets having minimum *support* (called *frequent itemsets).*
ii. Generate association rules using *frequent itemsets*.

Using the association rules, the questions are predicted and are stored in *predicted questions database* given by PQD. Each PQD contains QC and termset of predicted questions, the structure of which is shown in Fig. 8.4.

Question class(QC)	Termset of Predicted Questions(PQs)	Answer(s)

Fig. 8.4 Structure of Predicted questions database PQD

The *question class* and the *termset of predicted questions* are passed to the *searcher* that searches for the answer(s) to each predicted question in the *Question classified index.* The answers found thereof are stored in the same database corresponding to the *termset of predicted question* (see Fig. 8.4).

Now, when a user asks next question, it goes under the classification process and its terms are matched with the termset of the predicted questions stored in PQD. If the question is found, its answer(s) can be provided from there otherwise the answer is searched in the index. This improves the response time of the PQAS by providing a faster response to the user. The example of the above is given in the section below:

8.3 EXAMPLE OF NEXT QUESTION PREDICTION

Consider the questions extracted from a user session belonging to "what" question class shown in Table 8.1.

Table 8.1 Questions belonging to "what" question class

Questions
Q1:What is B.Tech.?
Q2:What is the procedure for admission in B.Tech.?
Q3.What is the eligibility for B.Tech. admission?
Q4:What is fee for B.Tech. admission?
Q5:What is scope of placement for B.Tech.?
Q6:What is M.Tech.?
Q7:What is the procedure for admission in M.Tech.?
Q8.What is the eligibility for M.Tech. admission?
Q9:What is fee for M.Tech. admission?
Q10. What is scope of placement for M.Tech.?

After applying the linguistic preprocessing, the queries formed are shown in Table 8.2.

Table 8.2 Queries formed for the above questions

Question type	Queries
What	B.Tech.
What	Procedure,admission,B.Tech.
What	Eligibility, admission, B.Tech.
What	Fee, admission, B.Tech.
What	Scope, placement, B.Tech.
What	M.Tech.
What	Procedure,admission,M.Tech.
What	Eligibility, admission, M.Tech.
What	Fee, admission, M.Tech.
What	Scope, placement, M.Tech.

Now, applying Step1 of *A-priori* algorithm, several itemsets are generated consisting of one item each. Some sample itemsets are listed below:

Itemset1:{B.Tech.}

2:{procedue}

3{admission}

4:{eligibility}

5:{fee}

6:{scope}

7:{placement}

8:{M.Tech}

After identifying the itemsets, *support (sup)* for each of the above is computed using eq. 8.1. Assuming the threshold value for *support* i.e. *minsup*=20% (0.2). Those itemsets that satisfy the criterion *sup>=minsup* are selected as *frequent itemsets.*

Sup(Itemset1):5/10=0.5

Sup(Itemset2):0.2

Sup(Itemset3):0.3

Sup(Itemset4):0.2

Sup(Itemset5):0.2

Sup(Itemset6): 0.2

Sup(Itemset7): 0.2

Sup(Itemset8): 0.5

It has been found that the Itemsets have *sup> minsup*, so are considered and two-item itemsets are generated using them as follows:

Itemset9:{B.Tech., procedure}

10:{ B.Tech., admission}

11:{ B.Tech., eligibility}

12:{ B.Tech., fee}

13:{ B.Tech., scope}

14:{ B.Tech., placement}

15:{ B.Tech., M.Tech.}

16:{procedure, admission}

17:{ procedure, eligibility}

18:{ procedure, fee}

19:{ procedure, scope}

20:{procedure, placement}

21:{procedure, M.Tech.}

22:{admission, eligiblity}

23:{admission,fee}

24:{admission,scope}

25:{admission,placement}

26:{admission,M.Tech.}

27:{eligibility, fee}

28:{eligibility,scope}

29:{eligibility,placement}

30:{eligibility,M.Tech}

31:{fee,scope}

32:{fee,placement}

33:{fee,M.Tech}

34:{scope, placement}

35:{scope, M.Tech}

36:{placement, M.Tech}

Now, for these above itemsets *sup* is computed and those itemsets that satisfy the criterion *sup>=minsup* are selected for generating 3-item itemsets.

It has been found that the *sup* of itemset 10, 16, 22, 23, 26 and 34 is greater than or equal to *minsup*. Similarly, the three item itemsets are generated but none of them satisfy the specified criterion. So, the frequent itemsets found are listed below:

{B.Tech, admission}

{procedure, admission}

{admission, eligibility}

{admission, fee}

{admission, M.Tech}

{scope, placement}

All the above six itemsets satisfy the criterion *sup>=minsup*, so these are selected as frequent itemsets and are used to generate the association rules, as follows:

Rule 1:B.Tech->admission

2: admission->B.Tech

3:procedure->admission

4:admission->procedure

5: admission->eligibility

6:eligibility->admission

7:admission->fee

8:fee->admission

9:admission->M.Tech

10:M.Tech.->admission

11: scope->placement

12: placement->scope

The *confidence* value of each of the above rules is computed and those rules that have *conf>=minconf* are selected. Threshold value of *minconf* is assumed to be 0.5. For the above rules, the *conf* value is as follows:

conf(Rule1)=3/5=0.6

conf(Rule2)=0.5

conf(Rule3)=1.0

conf(Rule4)=0.3

conf(Rule5)=0.3

conf(Rule6)=1.0

conf(Rule7)=0.3

conf(Rule8)=1.0

conf(Rule9)=0.5

conf(Rule10)=0.6

conf(Rule11)=1.0

conf(Rule12)=1.0

It may be observed that Rule1,2,3,6,8,9,10.11,12 satisfy the criterion *conf>=minconf.* So, these five rules are selected to predict next questions. By using the selected a*ssociation rules* given below, the module predicts next questions. The termset of predicted questions are given in Table 8.3:

Table 8.3 Predicted questions

Question class	**Termset of predicted questions**
What	{B.Tech., admission}
What	{M.Tech., admission}
What	{scope, placement}

Answers of the above predicted questions are looked up in the Question classified index and are stored in PQD in form of Question-Answer pair. If the next user's question matches with any of the question in PQD, its answer can be provided from there only without an need to look up in the Question classified index. This improves the response time of the PQAS by providing a faster response to the user.

8.4 IMPLEMENTATION AND RESULTS

The proposed approach is implemented on .Net framework using C# as front end and MS-SQL as back end. For the implementation of the proposed system, about 50 questions have been selected as sample and on them the proposed approach is applied. The snapshots of the implementation are given in Appendix-8.

Consider some sample questions given to PQAS,

Q1:What is B.Tech.?

Q2:What is the procedure for admission in B.Tech.?

Q3:What is fee for B.Tech. admission?

Q4:What is scope of placement for B.Tech.?

The response time of PQAS is shown in Table 8.4, in case when this approach of prediction is not followed.

Table 8.4 Questions given to PQAS

Question	Response time (in sec)
What is B.Tech.?	14
What is the procedure for admission in B.Tech.?	12
What is fee for B.Tech. admission?	12
What is scope of placement for B.Tech.?	16

After following the proposed approach of prediction, the termset of predicted questions along with the PQAS response time are shown in Table 8.5. See the first three predicted questions, for the improved response time.

Table 8.5 Improvement in PQAS response time

Termset of predicted Question	Response time (in sec)
{B.Tech, admission}	6
{M.Tech, admission}	7
{scope, placement}	4

From the above table, it can be analyzed that the PQAS has shown a significant improvement in the response time.

The implementation and results of Prospective Question Answering System is given in Chapter-9.

Chapter IX

IMPLEMENTATION & RESULT ANALYSIS

9.1 GENERAL

The *prospective question answering system (PQAS)*[99] designed in this thesis focuses on various components of the question answering process i.e. crawling, extracting relevant content and indexing. The quality of information is maintained by using the information from blogosphere as the source. In the direction of improvement, some approaches have been proposed for blog page ranking and next question prediction. As discussed, in the previous chapters, the architecture of PQAS comprises of six functional modules as listed below:

(i) crawl blogs
(ii) extract relevant content
(iii) index blogs
(iv) classify question
(v) searcher
(vi) look up for alternate data sources

The experimental analysis of PQAS is given in the following sections.

9.2 EXPERIMENTAL EVALUATION OF THE PROPOSED SYSTEM

In order to test the proposed *question answering system*, the architecture has been implemented using JDK 1.8 whose home page is shown in Fig. 9.1.

The home page comprises of a text box for writing the question, some searching tips and a search button. On clicking the button, the answer(s) are displayed on the PQAS interface. Also, a message “was the answer satisfactory” is displayed with two options “yes” or “no” for the user to click. For analysis, a set of 140 questions belonging to seven question classes of 20 questions each were taken as a sample data set.

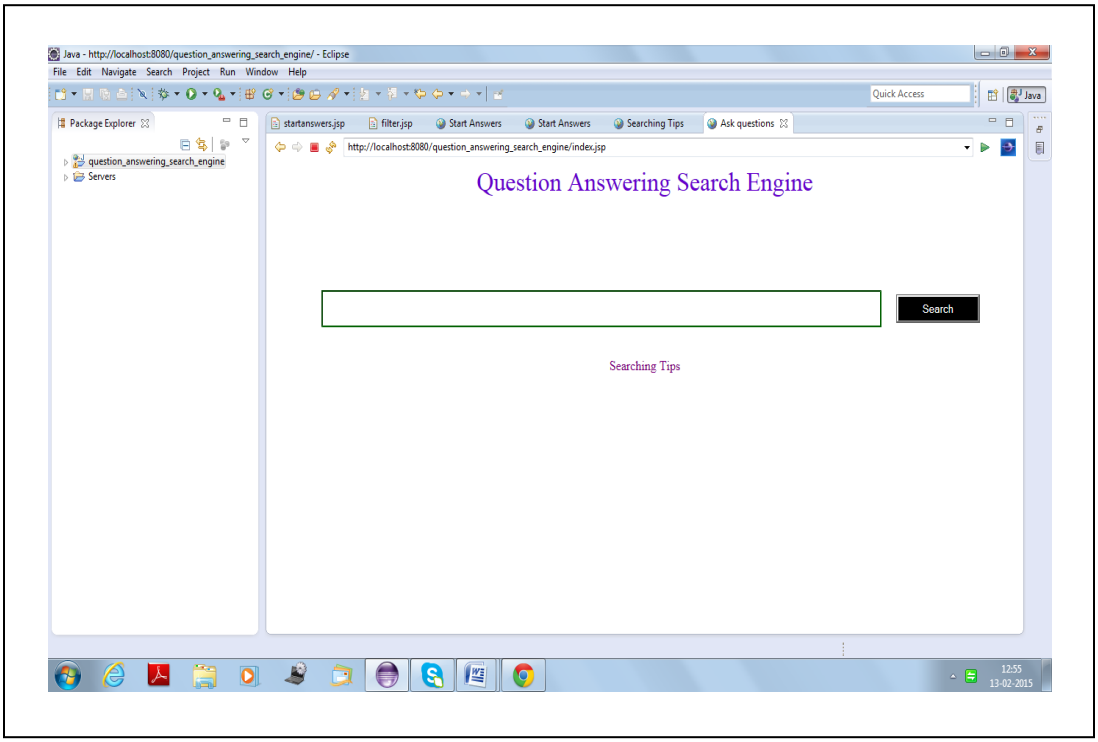


Fig. 9.1 Home Page of PQAS

The performance of PQAS has been measured via the metric *Answer accuracy* which may be defined as given below:

> *Answer accuracy* is defined as a fraction of answers found satisfactory by the user returned over all the answers returned by PQAS.

Mathematically, the *Answer accuracy* is given by:

$$Acc = C/(C + W) \ldots\ldots\ldots\ldots \text{eq. } 9.1$$

where C is numbers of answers found satisfactory by the user and

> W is the number of answers not found to be satisfactory.

At first, a set of twenty questions were prepared for "who" question class and it was handed over to five users. The first user enters each question one by one on the interface of PQAS. For each question, the response given by PQAS was collected and the feedback

of the user was taken in terms of "satisfaction" or "dissatisfaction" as already stated. Then the same set was given to the second user following the same process and so on. The same procedure was carried for the set of questions belonging to other question classes by choosing different five users. On the responses the performance metric namely *Answer accuracy* was applied for each user and its average was taken.

A detailed discussion on the performance analysis of answer(s) given by PQAS for each set of questions is given in the following sections.

9.3 SET-1 (QUESTIONS STARTING WITH "WHO")

The set consisting of 20 questions for "who" question class is shown in table 9.1 along with the answer(s).

Table 9.1 Questions-Answers for "who" class

Question	Answer(s) returned by PQAS
Who is Mahatma Gandhi?	Gandhi led India to independence and inspired movements for civil rights and freedom across the world. The honorific Mahatma applied to him first in 1914 in South Africa. He is also called Bapu in India.
Who is Alan Turing?	Alan Turing is often called the father of modern computing. Alan Turing was a brilliant mathematician and logician.
Who is Charles Babbage?	Charles Babbage, FRS was an English mathematician philosopher, inventor and mechanical engineer who originated the concept of a programmable computer. Considered a "father of the computer", Babbage is credited with inventing the first mechanical computer that eventually led to more complex designs.
Who is Andrew Barto?	Andrew Barto is a professor of computer science at University of Massachusetts Amherst,
Who is Tim Bernershee?	Tim Berners-Lee invented the World Wide Web in 1989.
Who is Ramesh Jain?	Ramesh joined University of California, Irvine as the first Bren Professor in Bren School of Information and Computer Sciences in 2005. Ramesh has been an active researcher in experiential computing, multimedia information systems, machine vision, and intelligent systems.
Who is John Hughes?	John Wilden Hughes, Jr. (February 18, 1950 – August 6, 2009) was an American film director, producer, and screenwriter.
Who is Ivar Jacobson?	Ivar Hjalmar Jacobson (born 1939) is a Swedish computer

	scientist and software engineer, known as major contributor to UML, Objectory, Rational Unified Process (RUP), aspect-oriented software development and Essence.
Who is Sonia Gandhi?	New delhi the nation on January 30 remembered mahatma Gandhi on his 65th death anniversary with president pranab mukherjee and prime minister manmohan singh leading the country in paying homage to the father of the nation. Leaders of various political parties and people from different walks of lfe also paid homage to Gandhi. Upa chairperson Sonia Gandhi senior bjp leaders lk advani and sushma swaraj and chiefs of three services paid tributes to the father of the nation at his memorial. Vice president hamid ansari and singh paid floral tributes at gandhi's memorial at rajghat at a function.
Who is Jonathan James?	Jonathan Joseph James was an American hacker who was the first juvenile incarcerated for cybercrime in the United States.
Who was the first female writer to win Nobel Prize?	The first woman to win a Nobel Prize was Marie Curie.
Who was the first to win the Nobel Peace Prize?	No answer
Who is Peter Wegner?	Peter Wegner (born 1963) is an American artist whose works consist of paintings, photographs, collages, prints, artist's books, and large-scale installations.
Who is Richard Karp?	Richard Manning Karp (born January 3, 1935) is an American computer scientist and computational theorist at the University of California, Berkeley.
Who is David Korn?	David Korn received his undergraduate degree in mathematics from RPI in 1965 and his Ph.D. in applied mathematics.
Who is Donald Knuth?	Donald Knuth was born January 10, 1938, in Milwaukee, Wisconsin.
Who is Bill Joy?	Bill Joy left Sun Microsystems, the computer company he cofounded, with no definite plans.
Who is Cliff Jones?	Cliff Jones (born 1968, London) is a British musician, songwriter, record producer and journalist who came to prominence as the singer with the Britpop band Gay Dad.
Who is David Johnson?	David Johnson (born 1957) was born in Jacksonville, Florida, and grew up in Daytona Beach, Florida.
Who is Dennis Wisnosky?	No answer

The answer(s) were analyzed using the performance metrics *Answer accuracy* by each of the five users and the response obtained thereof is given in Table 9.2 in terms of Answer Accuracy (in %) and its Average.

Based on the feedback of the first user, it was found that out of 20 questions, the system provided answers for 18 questions, out of which answers for 17 questions were found to be satisfactory and for 1 question, the system has not given satisfactory response. So, using the terms defined above, C=17, W=1.

Using eq. 9.1, Acc=17/(17+1)=94%.

Table 9.2 Average answer accuracy for "who" class

User	Answer accuracy (in %)
1	94
2	100
3	88
4	83
5	94
Average	91.8

Similarly, the Answer accuracy is found for the other users. The snapshots of some sample questions and their answer(s) given by PQAS have been shown in Appendix-9. The average answer accuracy for the questions belonging to "who" question class has been found to be 91.8%.

9.4 SET-2 (QUESTIONS STARTING WITH "WHERE")

The set consisting of 20 questions for "where" question class is shown in table 9.3 along with the answer(s).

Table 9.3 Questions-Answers for "where" class

Question	Answer(s) returned by PQAS
Where are Himalyas?	The Himalayas or Himalaya Sanskrit, hima (snow) literally "abode of snow" is a mountain range in Asia.

Where is Delhi?	Delhi is located at 28.61°N 77.23°E, and lies in northern India.
Where is Mumbai?	Mumbai (previously known as Bombay) is the biggest metropolis of India.
Where is Faridabad?	Faridabad is the largest city of Haryana in northern India in Faridabad district. The railway station of Old Faridabad and new industrial township are the major ones.
Where is Burma?	Burma Country Information: Burma is located in southeastern Asia. Burma is bordered by the Bay of Bengal and the Andaman Sea, Bangladesh and India to the north, China, Thailand, and Laos to the east.
Where is Brazil?	South America is the continent on which Brazil is located.
Where is Argentina?	Argentina is the second largest country in South America, constituted as a federation of 23 provinces and an autonomous city, Buenos Aires.
Where is Cuba?	Cuba is located at the entrance to the gulf of Mexico.
Where is Chile?	Chile is located in the Western South America and lies between latitudes 30° 0' S, and longitudes 71° 00' W.
Where is Canada?	Canada is a country in North America consisting of ten provinces and three territories. Located in the northern part of the continent, it extends from the Atlantic to the Pacific and northward into the Arctic Ocean.
Where is North Korea?	North Korea shares land borders with China and Russia to the north, and borders South Korea along the Korean Demilitarized Zone.
Where is India?	The country of India is located in southeast Asia. India is a country in South Asia.
Where is Maldives?	The Maldives is an independent republic state with a population of some 300,000. Maldives is situated in the Indian Ocean near Sri Lanka and India.
Where is Libya?	No answer
Where is Turkey?	The country Turkey is located on the continent of Asia.
Where is Thailand?	The country Thailand is located on the continent of Asia.
Where is United Arab Emirates?	The United Arab Emirates, in the eastern part of the Arabian Peninsula, extends along part of the Gulf of Oman and the southern coast of the Persian Gulf.
Where is Tunisia?	Tunisia is situated on the Mediterranean coast of North Africa, midway between the Atlantic Ocean and the Nile Delta.
Where is Sweden?	Sweden has the largest population among the Nordic countries and is the third-largest country in the European Union by surface area.
Where is Singapore?	Singapore is an island nation, both a city and a country, located just off the southern tip of Malaysia in Southeast Asia.

The answer(s) were analyzed using the performance metrics *Answer accuracy* by each of the five users and the response obtained thereof is given in Table 9.4 in terms of Answer accuracy (in %) and its average.

Table 9.4 Average Answer accuracy for "where "question class

User	Answer accuracy (in %)
1	95
2	89
3	84
4	100
5	100
Average	93.6

Based on the feedback of the first user, it is found that out of 20 questions, the system provided answers for 19 questions, out of which answers for 18 questions were found to be satisfactory and for 1 question, the system has not given satisfactory response. So, using the terms defined above, C=18, W=1.

Using eq. 9.1, Acc =18/(18+1)=95%,

Similarly, the Answer accuracy is found for the other users. The snapshots of some sample questions and their answer(s) given by PQAS have been shown in Appendix-9. The average answer accuracy for the questions belonging to "who" question class has been found to be 93.6%.

9.5 SET-3 (QUESTIONS STARTING WITH "WHEN")

The set consisting of 20 questions for "when" question class is shown in Table 9.5 along with the answer(s).

Table 9.5 Questions-Answers for "when" class

Question	Answer(s) returned by PQAS
When was Manmohan Singh born?	Manmohan Singh was born on September 26, 1932 (age 82 years).
When was Sonia Gandhi born?	Sonia Gandhi, original name Edvige Antonia Albina Maino (born December 9, 1946,

	Lusiana, Veneto region, Italy).
When was Jonathan Alder born?	Jonathan Alder, was born in Maryland in 1773 and a few years later he and his family moved to western Virginia in Wythe County.
When was William Harvey born?	William Harvey (1 April 1578) was an English physician
When was Pope Julius died?	Pope Julius died on February 21, 1513, Rome, Italy.
When did Henry died?	No answer
When did Marie Alfred Cornu, a French physicist died?	No answer
When was Humphry Ditton, an English Mathematician born?	Ditton was born at Salisbury on May 29, 1675, and died in London in 1715 at Christ's Hospital
When was James Fitz Gerald born?	FitzGerald was born on 4 March 1818 in Bath, England.
When did Feodor Chaliapin, Russian opera singer died?	No answer
When did Mahatma Gandhi died?	February 22,1944
When was Mahatma Gandhi born?	April 11 1869
When is Diwali?	Diwali also called the festivals of lights is an ancient Hindu festival celebrated in autumn every year. The festivals preparations and rituals typically extend over a five day period, but the main festival night of Diwali coincides with the darkest new moon night of Hindu Lunisolar month Kartik in the Gregorian calendar.
When did Georg Mohr born?	No answer
When did Mother Teresa born?	Blessed Teresa of Calcutta, MC, commonly known as Mother Teresa (26 August 1910– 5 September 1997), was a Roman Catholic religious sister and missionary who lived most of her life in India.
When did Mother Teresa died?	Mother Teresa died on 5 September 1997.
When was Joseph de Maistre born?	Joseph de Maistre was born on April 1, 1753.
When was Wallace Beery born?	Wallace Fitzgerald Beery born on April 1, 1885 was an American actor.
When was Abraham born?	Abraham was born under the name Abram in the city of Ur in Babylonia in the year 1948.
When was King John of France died?	King John died on April 8, 1364.

The answer(s) were analyzed using the performance metrics *Answer accuracy* by each of the five users and the response obtained thereof is given in Table 9.6 in terms of Answer Accuracy (in %) and its Average.

Table 9.6 Average Answer accuracy for "when "question class

User	Answer accuracy (in %)
1	87
2	93
3	81
4	81
5	87
Average	85.8

Based on the feedback of the first user, it is found that out of 20 questions, the system provided answers for 16 questions, out of which answers for 14 questions were found to be satisfactory and for 2 questions, the system has not given satisfactory response. So, using the terms defined above, C=14, W=2.

Using eq. 9.1, Acc =14/(14+2)=87%,

Similarly, the Answer accuracy is found for the other users. The snapshots of some sample questions and their answer(s) given by PQAS have been shown in Appendix-9. The average answer accuracy for the questions belonging to "when" question class has been found to be 85.8%.

9.6 SET-4 (QUESTIONS STARTING WITH "WHAT")

The set consisting of 20 questions for "what" question class is shown in table 9.7 along with the answer(s).

Table 9.7 Questions-Answers for "what" class

Question	Answer(s) returned by PQAS
What do you mean by grammar?	Grammar is the set of structural rules governing the composition of clauses, phrases, and words in any given natural language.
What is a Language?	Language is the human capacity for acquiring and using complex systems of communication and a language is any specific example of such systems. The scientific study of language is called linguistics.
What is Network Interface card?	Often abbreviated as NIC, an expansion board inserted into computer as the computer can be connected to a network. NICs are designed for

	particular types of network protocol and media although some can serve multiple networks.
What is a variable?	In programming, a variable is a value that can change depending on condition or on information passed to the program. Variable is a logical set of attributes.
What is RGB model?	The RGB color model is an additive color modelin which red, green, and blue light are added together in various ways to reproduce a broad array of colors.
What is Multimedia?	Definition of multimedia: Integration of animation, audio, graphics, text, and full-motion video through computer hardware and software for education.
What is the concept of queues?	In general, a queue is a line of people or things waiting to be handled, usually in sequential order starting at the beginning or top of the line or sequence. In computer technology, a queue is a sequence of work objects that are waiting to be processed.
What is a Data structure?	In computer science, a data structure is a particular way of organizing data in a computer so that it can be used efficiently.
What is the difference between Linux and Unix?	Linux is an open source operating system widely used for computer hardware and software, game development, tablet PCS, mainframes etc. Unix is an operating system commonly used in workstations by IBM and PCs by Solaris, Intel, HP etc.
What is IP address?	Internet Protocol address (IP address) is a numerical label assigned to each device (e.g., computer, printer) participating in a computer network that uses the Internet Protocol for communication.
What is cloud computing?	In cloud computing, the word cloud (also phrased as "the cloud") is used as a metaphor for "the Internet," so the phrase cloud computing means "a type of Internet-based computing".
What is the currency of USA?	currency of USA is United States Dollar
What is the currency of Dubai?	The official currency in Dubai is named Dirham.
What is hyperlink?	hyperlink is a reference to data that the reader can directly follow either by clicking or by hovering.
What is hypertension?	Hypertension occurs when the pressure inside the blood vessels is too high.
What is the motto of Asian Games?	No answer
What is the most basic level of storage?	No answer
What is the meaning of hypothesis?	A hypothesis (plural hypotheses) is a proposed explanation for a phenomenon.

What is the meaning of hypocrite?	Think of a hypocrite as a person who pretends to be a certain way, but really acts and believes the total opposite.
What is the definition of computer?	A computer generally means a programmable machine.

The answer(s) were analyzed using the performance metrics *Answer accuracy* by each of the five users and the response obtained thereof is given in Table 9.8 in terms of Answer Accuracy (in %) and its Average.

Table 9.8 Average Answer accuracy for "what" question class

User	Answer accuracy (in %)
1	89
2	94
3	88
4	88
5	94
Average	90.6

Based on the feedback of the first user, Out of 20 questions, the system provided answers for 18 questions, out of which answers for 16 questions were found to be satisfactory and for 2 questions, the system has not given satisfactory response. So, using the terms defined above, C=16, W=2.

Using eq. 9.1, Acc =16/(16+2)=89%.

Similarly, the Answer accuracy is found for the other users. The snapshots of some sample questions and their answer(s) given by PQAS have been shown in Appendix-9. The average Answer accuracy for the questions belonging to "what" question class has been found to be 90.6%.

9.7 SET-5 (QUESTIONS STARTING WITH "WHICH")

The set consisting of 20 questions for "which" question class is shown in table 9.9 along with the answer(s).

Table 9.9 Questions-Answers for ”which” class

Question	Answer(s) returned by PQAS
Which is the highest mountain range?	Mount Everest located on the border of Tibet and Nepal, in the central Himalayas, is the tallest mountain in the world.
Which holiday falls on April 4?	Qingming Festival is on April 4 or 5.
Which holiday falls on August 15?	Independence day is on August 15.
Which is the highest mountain in the world?	Mount Everest is called the world's highest mountain because it has the highest elevation above sea level.
Which is the longest river in the world?	Nile is the longest river in the world.
Which is the largest bridge in the world?	Jiaozhou Bay Bridge is the longest bridge.
Which planet is closest to the Earth?	It depends because planets are moving all the times in their orbits around the sun, the distance from each planet to earth is constantly changing.
Which is the smallest planet?	The smallest and largest planets in our solar system are Mercury and Jupiter, respectively.
Which is the smallest state in India?	smallest state in India Lakshadweep
Which is the city of skyscrapers?	New York is the city of skyscrapers. Hong kong is the city of sky scrapers.
Which city is known as Manchester of India?	Kanpur and Ahmedabad are called Manchester of India.
Which city is known as pink city?	Jaipur has been popularized with the name of Pink City because of the color of the stone exclusively used for the construction.
Which city is known as city of palaces?	City of Palaces is a traditional nickname for Mexico City, Mexico, because of its many palaces.
Which is the hottest planet?	The greenhouse effect is what scientists believe causes Venus to be the hottest planet
Which is the highest populated city in India?	No answer
Which country is zero waste country?	No answer
Which country is known as sugar bowl of the world?	Sugar Bowl of the World, Cuba.
Which Island is best in Maldives?	No answer
Which color is the best radiator of electromagnetic energy?	No answer
Which holiday falls on April 13?	Vaisakhi is a Sikh holy day as well as a celebration of the New Year in the Nanakshahi calendar.

The answer(s) were analyzed using the performance metrics *Answer accuracy* by each of the five users and the response obtained thereof is given in Table 9.10 in terms of Answer Accuracy (in %) and its Average.

Table 9.10 Average Answer accuracy for "which" question class

User	Answer accuracy (in %)
1	81
2	80
3	80
4	86
5	80
Average	81.4

Based on the feedback of the first user, Out of 20 questions, the system provided answers for 16 questions, out of which answers for 13 questions were found to be satisfactory and for 3 questions, the system has not given satisfactory response. So, using the terms defined above, C=13, W=3.

Using eq. 9.1, Acc =13/(13+3)=81%,

Similarly, the Answer accuracy is found for the other users. The snapshots of some sample questions and their answer(s) given by PQAS have been shown in Appendix-9. The average answer accuracy for the questions belonging to "which" question class has been found to be 81.4%.

9.8 SET-6 (QUESTIONS STARTING WITH "WHY")

The set consisting of 20 questions for "why" question class is shown in table 9.11 along with the answer(s).

Table 9.11 Questions-Answers for "why" class

Question	Answer(s) returned by PQAS
Why is sky blue?	The color of sky results from scattering of sunlight by the gas molecules and dust particles in the atmosphere. When the sky is clear, the waves of blue light are scattered much more than those of any other color. As the result, the sky appears blue.
Why is Diwali celebrated?	Diwali or Deepawali, the festival of lights is one of the biggest and brightest festivals in India

	andcelebrated across the country as a symbol of victory of good over evil.
Why is Holi celebrated?	Holi or 'Phagwah' is the most colourful festival celebrated by followers of the Vedic Religion.
Why is sea water salty?	After years and years of river inflow and evaporation, the salt content of the lake waterbuilt up to the present levels. The same process made the seas salty. Rivers carry dissolved salts to the ocean.
Why is Hockey the National sport of India?	Hockey is a family of sports in which two teams play against each other by trying to maneuver a ball or a puck into the opponent's goal using a hockey stick
Why is mkv better than mp4?	No answer
Why is android better than ios?	No answer
Why is angiography done?	No answer
Why is republic day celebrated?	66th Republic Day of India - Know more about India's most important national events.
Why is email important?	Wsj.com is available in the following editions and languages we sent an email to. please click on the link to complete your registration please register a new survey ranks the tata group as the most successful company in India com is available in the following editions and languages: we sent an email to please click on the link to complete your registration please register to gain free access to wsj tools.
Why is road safety important?	No answer
Why is teamwork important?	Well it's one thing to create a team, but quite another to create teamwork. Just as it's one thing to join a team, but quite another to perform as a team member. To put it simply, teams don't work without teamwork.
Why is lotus known as the national flower of India?	The Lotus, the national flower of India, is a symbol of supreme reality.
Why is linux better than unix?	Linux is an open source operating system widely used for computer hardware and software, game development, tablet PCS, mainframes etc. Unix is an operating system commonly used in workstations by IBM and PCs by Solaris, Intel, HP etc.
Why is Jodhpur blue?	Jodhpur is second largest city in the Indian state of Rajasthan and has long been a popular destination among international tourists.
Why is Jodhpur called sun city?	Jodhpur is a popular tourist destination, featuring many palaces, forts and temples, set in the stark landscape of the Thar Desert. The city is known as the "Sun City" for the bright, sunny weather it enjoys all the year round.
Why is pole star stationery?	The Polar Star or the North Star just happens to be aligned with the Axis of the revolving Earth. It has no connection what-so-ever.
Why is rti important?	RTI becomes very important for educators to use as a tool for addressing disproportionality because of its

	focus on data-based decision making.
Why is ruble falling?	No answer
Why is texture important in Interior designing?	Texture is an element which is generally overlooked when designing a home. Texture refers to how the surface of an object feels; therefore, you are no longer confined to visual elements such as line and color, now you can actually determine the way the space will feel too by using texture.

The answer(s) were analyzed using the performance metrics *Answer accuracy* by each of the five users and the response obtained thereof is given in Table 9.12 in terms of Answer accuracy (in %) and its average.

Based on the feedback of the first user, Out of 20 questions, the system provided answers for 15 questions, out of which answers for 10 questions were found to be satisfactory and for 5 questions, the system has not given satisfactory response. So, using the terms defined above, C=10, W=5.

Using eq. 9.1, Acc =10/(10+5)=67%.

Table 9.12 Average Answer accuracy for "why" question class

User	Answer accuracy (in %)
1	67
2	73
3	67
4	60
5	73
Average	68

Similarly, the Answer accuracy is found for the other users. The snapshots of some sample questions and their answer(s) given by PQAS have been shown in Appendix-9. The average answer accuracy for the questions belonging to "who" question class has been found to be 68.0%.

9.9 SET-7 (QUESTIONS STARTING WITH "HOW")

The set consisting of 20 questions for "how" question class is shown in table 9.13 along with the answer(s).

The answer(s) were analyzed using the performance metrics *Answer accuracy* by each of the five users and the response obtained thereof is given in Table 9.14 in terms of Answer Accuracy (in %) and its Average.

Table 9.13 Questions-Answers for ”how” class

Question	Answer(s) returned by PQAS
How is gratuity calculated?	Gratuity Calculation India is calculated by the employee's basic salary and the years of working experience.
How is paper made?	To make paper from trees, the raw wood must first be turned into “pulp.” Wood pulp is a watery “soup” of cellulose wood fibers, lignin, water and the chemicals used during the pulping process. Wood can be turned to pulp in a couple of different ways.
How is cotton made?	No answer
How is tax calculated?	Calculate your income tax as per new Budget 2015 - 16. Quick Tax Calculator, Free Income Tax Calculator Tool.
How to prepare tea?	For tea; Boil a mixture of water, milk and sugar (as much as needed for the entire serving). Once that happens, add the tea leaves and spices and continue to heat, turn off the burner and have tea.
How to implement vector class in Java?	The Vector class implements a growable array of objects.
How is vector implemented in C++?	C++ Source Code for a Vector - index based array of objects - implementation is given on this page with example. Vector is usually implemented as a contiguous block of memory.
How is cotton processed?	The cotton gin is where cotton fiber is separated from the cotton seed. The first step in the ginning process is when the cotton is vacuumed into tubes that carry it to a dryer to reduce moisture and improve the fiber quality.
How is vector graphics used?	No answer
How to use internet?	The internet is a global stream of interconnected computer networks that can use the standard Internet protocol suite (TCP/IP) to link several billions devices worldwide.
How to convert dollar to rupee?	As on Saturday, January 17, 2015; 2:25:20 am, there are 61.62 Indian rupees in united states dollar.
How to lose weight?	No answer
How is Christmas celebrated?	How to Celebrate Christmas. Christmas is one of those holidays that just seems to be filled with cheer and wonder.
How is beer made?	Beer is made from four basic ingredients: Barley, water, hops and yeast.

How is Celsius converted to Kelvin?	You can convert between Celsius and Kelvin like this: Kelvin =Celsius + 273.15.
How is gm converted to kg?	There are 1000 gm in one kg.
How is Precision computed?	Precision is the measure of accuracy.
How is Recall computed?	Recall is the measure of coverage.
How is bird flu transmitted?	No answer
How is Buddha Purnima celebrated?	No answer

Based on the feedback of the first user, Out of 20 questions, the system provided answers for 14 questions, out of which answers for 10 questions were found to be satisfactory and for 4 questions, the system has not given satisfactory response. So, using the terms defined above, C=10, W=4.

Using eq. 9.1, Acc =10/(10+4)=71%.

Table 9.14 Average Answer accuracy for "how" question class

User	Answer accuracy (in %)
1	71
2	78
3	71
4	78
5	64
Average	72.4

Similarly, the Answer accuracy is found for the other users. The snapshots of some sample questions and their answer(s) given by PQAS have been shown in Appendix-9. The average answer accuracy for the questions belonging to "who" question class has been found to be 72.4%. The graph shown in Fig. 9.2 shows the values of average Answer Accuracy for each question class. High values of *Answer accuracy Average (i.e. Acc=83.3%)* for various tests conducted on the system indicate that it accurately answers the questions that the user asks.

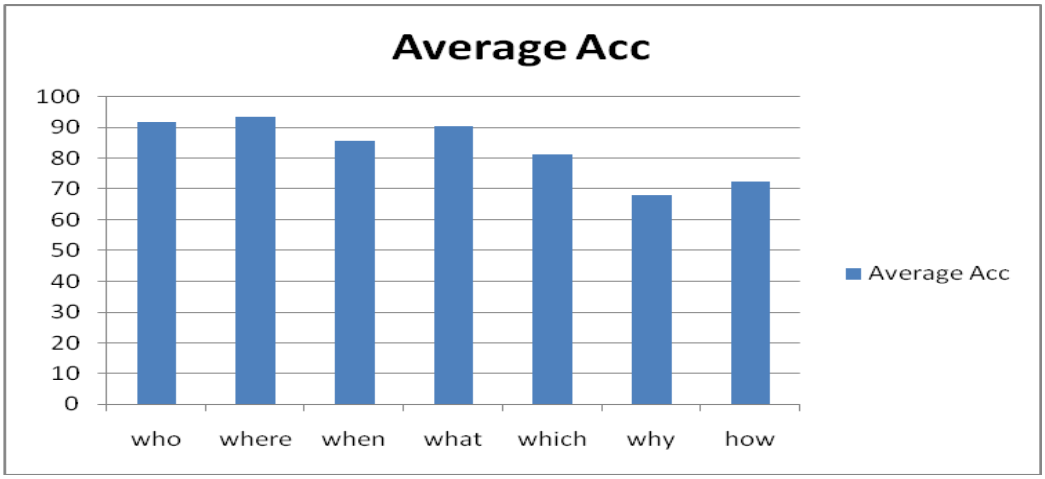


Fig. 9.2 Plotted values of *Average Answer accuracy*

It can be observed from the graph that the plotted values of *Answer accuracy* are higher for "who", "where", "what" and somewhat lower for "when" and "which". For "why" and "how", the plotted values are comparatively low.

9.10 COMPARISON OF PQAS WITH EXISTING QA SYSTEMS

In this work, architecture for *a novel search engine for prospective Question Answering* has been proposed. The accuracy of the proposed system, measured in term of *Precision* is found to be higher as compared to the existing systems for Question Answering as shown in Table 9.15. Thus, the proposed system is able to respond to user's question with high accuracy.

Table 9.15 Comparison of PQAS with Existing Question Answering systems

Characteristics	PQAS	Ask.com	Answers.com	START
History	The proposed system PQAS comprises of seven functional components and it is able to respond to the user's questions posed in a natural language with accurate answers. It is able to answer the questions that start with who, where, what, when, which, how and why.	Ask.com [100] (originally known as Ask Jeeves) was founded in 1996 by Garrett Gruener and David Warthen in Berkeley, Calif. It allows online searchers to get answers to questions posed in everyday, natural language.	Answers.com [101] is an Internet-based knowledge exchange, which includes WikiAnswers, ReferenceAnswers, VideoAnswers, and five international language Q&A communities. The Answers.com domain name was purchased by entrepreneurs Bill Gross and Henrik Jones at Idealab in 1996.	START [102], the world's first Web-based Natural language question answering system, has been on-line and continuously operating since December, 1993. It has been developed by Boris Katz and his associates of the InfoLab Group at the MIT Computer Science and Artificial Intelligence Laboratory.
Crawling	The system uses its own Blog crawler to download the blog pages.	Doesn't use crawler	Uses a general crawler	Uses a general crawler
Quality data	The system maintains the quality data by ranking the blog posts and ignoring those that have rank lower than the decided threshold value.	*	*	*
Answer generation	The proposed downloads the blog pages and then extracts the relevant content from these pages. The pages are then indexed for searching corresponding to user's questions. The system constructs and uses Question classified index for indexing and responding to user's questions. The system also	Uses Google to answer to the user's question with a list of web pages as result. It also provides an option that puts the question on community and asks its members to respond. The answers given by the members are then sent to the user who asked the question by email.	Uses wikianswers, ReferenceAnswers, VideoAnswers, and five international language Q&A communities to respond to user's questions. Also, puts the question asked under a suitable category and ask the members of the corresponding community	Uses some web sources like Wikipedia, some books, dictionaries, projects and some web sources to answer the user's question. START parses incoming questions, matches the queries created from the parse trees against its knowledge base and

	enriches its repository with the data obtained from other alternate sources containing user satisfactory answers.		to respond.	presents the appropriate information segments to the user.
Answer rating	Asks, if the user is satisfied?-yes or no	No	Asks, if the answer is useful?-yes, no or somewhat	No
Source of answers shown	No	Yes	Yes	Yes
User oriented results	If the user says that the answer to his question is not satisfactory, then the system looks for an alternate data source to answer to user's question.	No	Sometimes	Sometimes
Precision	High	*	*	*
Extensible	Yes	Can't say	Can't say	Can't say
Scalable	Yes	Can't say	Can't say	Can't say
Limitatons	1. Shows somewhat low precision in case of the questions that start with why and how.	1. User has to search for the answer in the result page given by the Search engine, thus aim the objective of Question answering not fulfilled. 2.Also, the user has to wait for the answer until it is provided by some member of the QA community of Ask.com	1. Question has to be categorized into an appropriate category and then is then forwarded to the members of the QA community for answering. 2. The user waits until it is responded by someone else.	1. The grammatical structure of the question asked needs to be accurate otherwise the system fails to answer the question. 2. Also, the question that mean the same as an already answered question but has few more terms may not be responded by the system.

* *not claimed.*

It may be noted that the *precision* of the proposed system in this thesis is higher as compared to other works [106,107,108]. The conclusion and future scope is presented in chapter-10.

Chapter IX

IMPLEMENTATION & RESULT ANALYSIS

9.1 GENERAL

The *prospective question answering system (PQAS)*[99] designed in this thesis focuses on various components of the question answering process i.e. crawling, extracting relevant content and indexing. The quality of information is maintained by using the information from blogosphere as the source. In the direction of improvement, some approaches have been proposed for blog page ranking and next question prediction. As discussed, in the previous chapters, the architecture of PQAS comprises of six functional modules as listed below:

(i) crawl blogs
(ii) extract relevant content
(iii) index blogs
(iv) classify question
(v) searcher
(vi) look up for alternate data sources

The experimental analysis of PQAS is given in the following sections.

9.2 EXPERIMENTAL EVALUATION OF THE PROPOSED SYSTEM

In order to test the proposed *question answering system*, the architecture has been implemented using JDK 1.8 whose home page is shown in Fig. 9.1.

The home page comprises of a text box for writing the question, some searching tips and a search button. On clicking the button, the answer(s) are displayed on the PQAS interface. Also, a message “was the answer satisfactory” is displayed with two options “yes” or “no” for the user to click. For analysis, a set of 140 questions belonging to seven question classes of 20 questions each were taken as a sample data set.

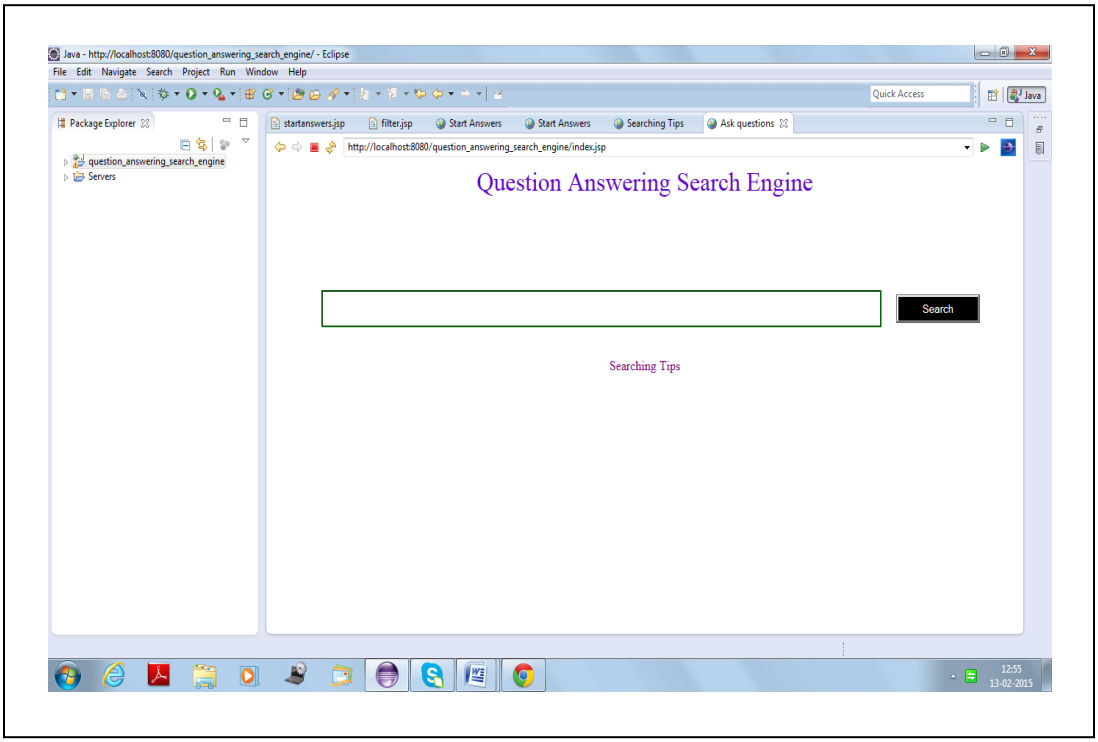


Fig. 9.1 Home Page of PQAS

The performance of PQAS has been measured via the metric *Answer accuracy* which may be defined as given below:

> *Answer accuracy* is defined as a fraction of answers found satisfactory by the user returned over all the answers returned by PQAS.

Mathematically, the *Answer accuracy* is given by:

$$Acc = C/(C + W) \ldots\ldots\ldots\ldots \text{eq. } 9.1$$

where C is numbers of answers found satisfactory by the user and

> W is the number of answers not found to be satisfactory.

At first, a set of twenty questions were prepared for "who" question class and it was handed over to five users. The first user enters each question one by one on the interface of PQAS. For each question, the response given by PQAS was collected and the feedback

of the user was taken in terms of "satisfaction" or "dissatisfaction" as already stated. Then the same set was given to the second user following the same process and so on. The same procedure was carried for the set of questions belonging to other question classes by choosing different five users. On the responses the performance metric namely *Answer accuracy* was applied for each user and its average was taken.

A detailed discussion on the performance analysis of answer(s) given by PQAS for each set of questions is given in the following sections.

9.3 SET-1 (QUESTIONS STARTING WITH "WHO")

The set consisting of 20 questions for "who" question class is shown in table 9.1 along with the answer(s).

Table 9.1 Questions-Answers for "who" class

Question	Answer(s) returned by PQAS
Who is Mahatma Gandhi?	Gandhi led India to independence and inspired movements for civil rights and freedom across the world. The honorific Mahatma applied to him first in 1914 in South Africa. He is also called Bapu in India.
Who is Alan Turing?	Alan Turing is often called the father of modern computing. Alan Turing was a brilliant mathematician and logician.
Who is Charles Babbage?	Charles Babbage, FRS was an English mathematician philosopher, inventor and mechanical engineer who originated the concept of a programmable computer. Considered a "father of the computer", Babbage is credited with inventing the first mechanical computer that eventually led to more complex designs.
Who is Andrew Barto?	Andrew Barto is a professor of computer science at University of Massachusetts Amherst,
Who is Tim Bernershee?	Tim Berners-Lee invented the World Wide Web in 1989.
Who is Ramesh Jain?	Ramesh joined University of California, Irvine as the first Bren Professor in Bren School of Information and Computer Sciences in 2005. Ramesh has been an active researcher in experiential computing, multimedia information systems, machine vision, and intelligent systems.
Who is John Hughes?	John Wilden Hughes, Jr. (February 18, 1950 – August 6, 2009) was an American film director, producer, and screenwriter.
Who is Ivar Jacobson?	Ivar Hjalmar Jacobson (born 1939) is a Swedish computer

	scientist and software engineer, known as major contributor to UML, Objectory, Rational Unified Process (RUP), aspect-oriented software development and Essence.
Who is Sonia Gandhi?	New delhi the nation on January 30 remembered mahatma Gandhi on his 65th death anniversary with president pranab mukherjee and prime minister manmohan singh leading the country in paying homage to the father of the nation. Leaders of various political parties and people from different walks of lfe also paid homage to Gandhi. Upa chairperson Sonia Gandhi senior bjp leaders lk advani and sushma swaraj and chiefs of three services paid tributes to the father of the nation at his memorial. Vice president hamid ansari and singh paid floral tributes at gandhi's memorial at rajghat at a function.
Who is Jonathan James?	Jonathan Joseph James was an American hacker who was the first juvenile incarcerated for cybercrime in the United States.
Who was the first female writer to win Nobel Prize?	The first woman to win a Nobel Prize was Marie Curie.
Who was the first to win the Nobel Peace Prize?	No answer
Who is Peter Wegner?	Peter Wegner (born 1963) is an American artist whose works consist of paintings, photographs, collages, prints, artist's books, and large-scale installations.
Who is Richard Karp?	Richard Manning Karp (born January 3, 1935) is an American computer scientist and computational theorist at the University of California, Berkeley.
Who is David Korn?	David Korn received his undergraduate degree in mathematics from RPI in 1965 and his Ph.D. in applied mathematics.
Who is Donald Knuth?	Donald Knuth was born January 10, 1938, in Milwaukee, Wisconsin.
Who is Bill Joy?	Bill Joy left Sun Microsystems, the computer company he cofounded, with no definite plans.
Who is Cliff Jones?	Cliff Jones (born 1968, London) is a British musician, songwriter, record producer and journalist who came to prominence as the singer with the Britpop band Gay Dad.
Who is David Johnson?	David Johnson (born 1957) was born in Jacksonville, Florida, and grew up in Daytona Beach, Florida.
Who is Dennis Wisnosky?	No answer

The answer(s) were analyzed using the performance metrics *Answer accuracy* by each of the five users and the response obtained thereof is given in Table 9.2 in terms of Answer Accuracy (in %) and its Average.

Based on the feedback of the first user, it was found that out of 20 questions, the system provided answers for 18 questions, out of which answers for 17 questions were found to be satisfactory and for 1 question, the system has not given satisfactory response. So, using the terms defined above, C=17, W=1.

Using eq. 9.1, Acc=17/(17+1)=94%.

Table 9.2 Average answer accuracy for "who" class

User	Answer accuracy (in %)
1	94
2	100
3	88
4	83
5	94
Average	91.8

Similarly, the Answer accuracy is found for the other users. The snapshots of some sample questions and their answer(s) given by PQAS have been shown in Appendix-9. The average answer accuracy for the questions belonging to "who" question class has been found to be 91.8%.

9.4 SET-2 (QUESTIONS STARTING WITH "WHERE")

The set consisting of 20 questions for "where" question class is shown in table 9.3 along with the answer(s).

Table 9.3 Questions-Answers for "where" class

Question	Answer(s) returned by PQAS
Where are Himalyas?	The Himalayas or Himalaya Sanskrit, hima (snow) literally "abode of snow" is a mountain range in Asia.

Where is Delhi?	Delhi is located at 28.61°N 77.23°E, and lies in northern India.
Where is Mumbai?	Mumbai (previously known as Bombay) is the biggest metropolis of India.
Where is Faridabad?	Faridabad is the largest city of Haryana in northern India in Faridabad district. The railway station of Old Faridabad and new industrial township are the major ones.
Where is Burma?	Burma Country Information: Burma is located in southeastern Asia. Burma is bordered by the Bay of Bengal and the Andaman Sea, Bangladesh and India to the north, China, Thailand, and Laos to the east.
Where is Brazil?	South America is the continent on which Brazil is located.
Where is Argentina?	Argentina is the second largest country in South America, constituted as a federation of 23 provinces and an autonomous city, Buenos Aires.
Where is Cuba?	Cuba is located at the entrance to the gulf of Mexico.
Where is Chile?	Chile is located in the Western South America and lies between latitudes 30° 0' S, and longitudes 71° 00' W.
Where is Canada?	Canada is a country in North America consisting of ten provinces and three territories. Located in the northern part of the continent, it extends from the Atlantic to the Pacific and northward into the Arctic Ocean.
Where is North Korea?	North Korea shares land borders with China and Russia to the north, and borders South Korea along the Korean Demilitarized Zone.
Where is India?	The country of India is located in southeast Asia. India is a country in South Asia.
Where is Maldives?	The Maldives is an independent republic state with a population of some 300,000. Maldives is situated in the Indian Ocean near Sri Lanka and India.
Where is Libya?	No answer
Where is Turkey?	The country Turkey is located on the continent of Asia.
Where is Thailand?	The country Thailand is located on the continent of Asia.
Where is United Arab Emirates?	The United Arab Emirates, in the eastern part of the Arabian Peninsula, extends along part of the Gulf of Oman and the southern coast of the Persian Gulf.
Where is Tunisia?	Tunisia is situated on the Mediterranean coast of North Africa, midway between the Atlantic Ocean and the Nile Delta.
Where is Sweden?	Sweden has the largest population among the Nordic countries and is the third-largest country in the European Union by surface area.
Where is Singapore?	Singapore is an island nation, both a city and a country, located just off the southern tip of Malaysia in Southeast Asia.

The answer(s) were analyzed using the performance metrics *Answer accuracy* by each of the five users and the response obtained thereof is given in Table 9.4 in terms of Answer accuracy (in %) and its average.

Table 9.4 Average Answer accuracy for "where "question class

User	Answer accuracy (in %)
1	95
2	89
3	84
4	100
5	100
Average	93.6

Based on the feedback of the first user, it is found that out of 20 questions, the system provided answers for 19 questions, out of which answers for 18 questions were found to be satisfactory and for 1 question, the system has not given satisfactory response. So, using the terms defined above, C=18, W=1.

Using eq. 9.1, Acc =18/(18+1)=95%,

Similarly, the Answer accuracy is found for the other users. The snapshots of some sample questions and their answer(s) given by PQAS have been shown in Appendix-9. The average answer accuracy for the questions belonging to "who" question class has been found to be 93.6%.

9.5 SET-3 (QUESTIONS STARTING WITH "WHEN")

The set consisting of 20 questions for "when" question class is shown in Table 9.5 along with the answer(s).

Table 9.5 Questions-Answers for "when" class

Question	Answer(s) returned by PQAS
When was Manmohan Singh born?	Manmohan Singh was born on September 26, 1932 (age 82 years).
When was Sonia Gandhi born?	Sonia Gandhi, original name Edvige Antonia Albina Maino (born December 9, 1946,

	Lusiana, Veneto region, Italy).
When was Jonathan Alder born?	Jonathan Alder, was born in Maryland in 1773 and a few years later he and his family moved to western Virginia in Wythe County.
When was William Harvey born?	William Harvey (1 April 1578) was an English physician
When was Pope Julius died?	Pope Julius died on February 21, 1513, Rome, Italy.
When did Henry died?	No answer
When did Marie Alfred Cornu, a French physicist died?	No answer
When was Humphry Ditton, an English Mathematician born?	Ditton was born at Salisbury on May 29, 1675, and died in London in 1715 at Christ's Hospital
When was James Fitz Gerald born?	FitzGerald was born on 4 March 1818 in Bath, England.
When did Feodor Chaliapin, Russian opera singer died?	No answer
When did Mahatma Gandhi died?	February 22,1944
When was Mahatma Gandhi born?	April 11 1869
When is Diwali?	Diwali also called the festivals of lights is an ancient Hindu festival celebrated in autumn every year. The festivals preparations and rituals typically extend over a five day period, but the main festival night of Diwali coincides with the darkest new moon night of Hindu Lunisolar month Kartik in the Gregorian calendar.
When did Georg Mohr born?	No answer
When did Mother Teresa born?	Blessed Teresa of Calcutta, MC, commonly known as Mother Teresa (26 August 1910– 5 September 1997), was a Roman Catholic religious sister and missionary who lived most of her life in India.
When did Mother Teresa died?	Mother Teresa died on 5 September 1997.
When was Joseph de Maistre born?	Joseph de Maistre was born on April 1, 1753.
When was Wallace Beery born?	Wallace Fitzgerald Beery born on April 1, 1885 was an American actor.
When was Abraham born?	Abraham was born under the name Abram in the city of Ur in Babylonia in the year 1948.
When was King John of France died?	King John died on April 8, 1364.

The answer(s) were analyzed using the performance metrics *Answer accuracy* by each of the five users and the response obtained thereof is given in Table 9.6 in terms of Answer Accuracy (in %) and its Average.

Table 9.6 Average Answer accuracy for "when "question class

User	Answer accuracy (in %)
1	87
2	93
3	81
4	81
5	87
Average	85.8

Based on the feedback of the first user, it is found that out of 20 questions, the system provided answers for 16 questions, out of which answers for 14 questions were found to be satisfactory and for 2 questions, the system has not given satisfactory response. So, using the terms defined above, C=14, W=2.

Using eq. 9.1, Acc =14/(14+2)=87%,

Similarly, the Answer accuracy is found for the other users. The snapshots of some sample questions and their answer(s) given by PQAS have been shown in Appendix-9. The average answer accuracy for the questions belonging to "when" question class has been found to be 85.8%.

9.6 SET-4 (QUESTIONS STARTING WITH "WHAT")

The set consisting of 20 questions for "what" question class is shown in table 9.7 along with the answer(s).

Table 9.7 Questions-Answers for "what" class

Question	Answer(s) returned by PQAS
What do you mean by grammar?	Grammar is the set of structural rules governing the composition of clauses, phrases, and words in any given natural language.
What is a Language?	Language is the human capacity for acquiring and using complex systems of communication and a language is any specific example of such systems. The scientific study of language is called linguistics.
What is Network Interface card?	Often abbreviated as NIC, an expansion board inserted into computer as the computer can be connected to a network. NICs are designed for

	particular types of network protocol and media although some can serve multiple networks.
What is a variable?	In programming, a variable is a value that can change depending on condition or on information passed to the program. Variable is a logical set of attributes.
What is RGB model?	The RGB color model is an additive color modelin which red, green, and blue light are added together in various ways to reproduce a broad array of colors.
What is Multimedia?	Definition of multimedia: Integration of animation, audio, graphics, text, and full-motion video through computer hardware and software for education.
What is the concept of queues?	In general, a queue is a line of people or things waiting to be handled, usually in sequential order starting at the beginning or top of the line or sequence. In computer technology, a queue is a sequence of work objects that are waiting to be processed.
What is a Data structure?	In computer science, a data structure is a particular way of organizing data in a computer so that it can be used efficiently.
What is the difference between Linux and Unix?	Linux is an open source operating system widely used for computer hardware and software, game development, tablet PCS, mainframes etc. Unix is an operating system commonly used in workstations by IBM and PCs by Solaris, Intel, HP etc.
What is IP address?	Internet Protocol address (IP address) is a numerical label assigned to each device (e.g., computer, printer) participating in a computer network that uses the Internet Protocol for communication.
What is cloud computing?	In cloud computing, the word cloud (also phrased as "the cloud") is used as a metaphor for "the Internet," so the phrase cloud computing means "a type of Internet-based computing".
What is the currency of USA?	currency of USA is United States Dollar
What is the currency of Dubai?	The official currency in Dubai is named Dirham.
What is hyperlink?	hyperlink is a reference to data that the reader can directly follow either by clicking or by hovering.
What is hypertension?	Hypertension occurs when the pressure inside the blood vessels is too high.
What is the motto of Asian Games?	No answer
What is the most basic level of storage?	No answer
What is the meaning of hypothesis?	A hypothesis (plural hypotheses) is a proposed explanation for a phenomenon.

What is the meaning of hypocrite?	Think of a hypocrite as a person who pretends to be a certain way, but really acts and believes the total opposite.
What is the definition of computer?	A computer generally means a programmable machine.

The answer(s) were analyzed using the performance metrics *Answer accuracy* by each of the five users and the response obtained thereof is given in Table 9.8 in terms of Answer Accuracy (in %) and its Average.

Table 9.8 Average Answer accuracy for "what" question class

User	Answer accuracy (in %)
1	89
2	94
3	88
4	88
5	94
Average	90.6

Based on the feedback of the first user, Out of 20 questions, the system provided answers for 18 questions, out of which answers for 16 questions were found to be satisfactory and for 2 questions, the system has not given satisfactory response. So, using the terms defined above, C=16, W=2.

Using eq. 9.1, Acc =16/(16+2)=89%.

Similarly, the Answer accuracy is found for the other users. The snapshots of some sample questions and their answer(s) given by PQAS have been shown in Appendix-9. The average Answer accuracy for the questions belonging to "what" question class has been found to be 90.6%.

9.7 SET-5 (QUESTIONS STARTING WITH "WHICH")

The set consisting of 20 questions for "which" question class is shown in table 9.9 along with the answer(s).

Table 9.9 Questions-Answers for ”which” class

Question	Answer(s) returned by PQAS
Which is the highest mountain range?	Mount Everest located on the border of Tibet and Nepal, in the central Himalayas, is the tallest mountain in the world.
Which holiday falls on April 4?	Qingming Festival is on April 4 or 5.
Which holiday falls on August 15?	Independence day is on August 15.
Which is the highest mountain in the world?	Mount Everest is called the world's highest mountain because it has the highest elevation above sea level.
Which is the longest river in the world?	Nile is the longest river in the world.
Which is the largest bridge in the world?	Jiaozhou Bay Bridge is the longest bridge.
Which planet is closest to the Earth?	It depends because planets are moving all the times in their orbits around the sun, the distance from each planet to earth is constantly changing.
Which is the smallest planet?	The smallest and largest planets in our solar system are Mercury and Jupiter, respectively.
Which is the smallest state in India?	smallest state in India Lakshadweep
Which is the city of skyscrapers?	New York is the city of skyscrapers. Hong kong is the city of sky scrapers.
Which city is known as Manchester of India?	Kanpur and Ahmedabad are called Manchester of India.
Which city is known as pink city?	Jaipur has been popularized with the name of Pink City because of the color of the stone exclusively used for the construction.
Which city is known as city of palaces?	City of Palaces is a traditional nickname for Mexico City, Mexico, because of its many palaces.
Which is the hottest planet?	The greenhouse effect is what scientists believe causes Venus to be the hottest planet
Which is the highest populated city in India?	No answer
Which country is zero waste country?	No answer
Which country is known as sugar bowl of the world?	Sugar Bowl of the World, Cuba.
Which Island is best in Maldives?	No answer
Which color is the best radiator of electromagnetic energy?	No answer
Which holiday falls on April 13?	Vaisakhi is a Sikh holy day as well as a celebration of the New Year in the Nanakshahi calendar.

The answer(s) were analyzed using the performance metrics *Answer accuracy* by each of the five users and the response obtained thereof is given in Table 9.10 in terms of Answer Accuracy (in %) and its Average.

Table 9.10 Average Answer accuracy for "which" question class

User	Answer accuracy (in %)
1	81
2	80
3	80
4	86
5	80
Average	81.4

Based on the feedback of the first user, Out of 20 questions, the system provided answers for 16 questions, out of which answers for 13 questions were found to be satisfactory and for 3 questions, the system has not given satisfactory response. So, using the terms defined above, C=13, W=3.

Using eq. 9.1, Acc =13/(13+3)=81%,

Similarly, the Answer accuracy is found for the other users. The snapshots of some sample questions and their answer(s) given by PQAS have been shown in Appendix-9. The average answer accuracy for the questions belonging to "which" question class has been found to be 81.4%.

9.8 SET-6 (QUESTIONS STARTING WITH "WHY")

The set consisting of 20 questions for "why" question class is shown in table 9.11 along with the answer(s).

Table 9.11 Questions-Answers for "why" class

Question	Answer(s) returned by PQAS
Why is sky blue?	The color of sky results from scattering of sunlight by the gas molecules and dust particles in the atmosphere. When the sky is clear, the waves of blue light are scattered much more than those of any other color. As the result, the sky appears blue.
Why is Diwali celebrated?	Diwali or Deepawali, the festival of lights is one of the biggest and brightest festivals in India

	andcelebrated across the country as a symbol of victory of good over evil.
Why is Holi celebrated?	Holi or 'Phagwah' is the most colourful festival celebrated by followers of the Vedic Religion.
Why is sea water salty?	After years and years of river inflow and evaporation, the salt content of the lake waterbuilt up to the present levels. The same process made the seas salty. Rivers carry dissolved salts to the ocean.
Why is Hockey the National sport of India?	Hockey is a family of sports in which two teams play against each other by trying to maneuver a ball or a puck into the opponent's goal using a hockey stick
Why is mkv better than mp4?	No answer
Why is android better than ios?	No answer
Why is angiography done?	No answer
Why is republic day celebrated?	66th Republic Day of India - Know more about India's most important national events.
Why is email important?	Wsj.com is available in the following editions and languages we sent an email to. please click on the link to complete your registration please register a new survey ranks the tata group as the most successful company in India com is available in the following editions and languages: we sent an email to please click on the link to complete your registration please register to gain free access to wsj tools.
Why is road safety important?	No answer
Why is teamwork important?	Well it's one thing to create a team, but quite another to create teamwork. Just as it's one thing to join a team, but quite another to perform as a team member. To put it simply, teams don't work without teamwork.
Why is lotus known as the national flower of India?	The Lotus, the national flower of India, is a symbol of supreme reality.
Why is linux better than unix?	Linux is an open source operating system widely used for computer hardware and software, game development, tablet PCS, mainframes etc. Unix is an operating system commonly used in workstations by IBM and PCs by Solaris, Intel, HP etc.
Why is Jodhpur blue?	Jodhpur is second largest city in the Indian state of Rajasthan and has long been a popular destination among international tourists.
Why is Jodhpur called sun city?	Jodhpur is a popular tourist destination, featuring many palaces, forts and temples, set in the stark landscape of the Thar Desert. The city is known as the "Sun City" for the bright, sunny weather it enjoys all the year round.
Why is pole star stationery?	The Polar Star or the North Star just happens to be aligned with the Axis of the revolving Earth. It has no connection what-so-ever.
Why is rti important?	RTI becomes very important for educators to use as a tool for addressing disproportionality because of its

	focus on data-based decision making.
Why is ruble falling?	No answer
Why is texture important in Interior designing?	Texture is an element which is generally overlooked when designing a home. Texture refers to how the surface of an object feels; therefore, you are no longer confined to visual elements such as line and color, now you can actually determine the way the space will feel too by using texture.

The answer(s) were analyzed using the performance metrics *Answer accuracy* by each of the five users and the response obtained thereof is given in Table 9.12 in terms of Answer accuracy (in %) and its average.

Based on the feedback of the first user, Out of 20 questions, the system provided answers for 15 questions, out of which answers for 10 questions were found to be satisfactory and for 5 questions, the system has not given satisfactory response. So, using the terms defined above, C=10, W=5.

Using eq. 9.1, Acc =10/(10+5)=67%.

Table 9.12 Average Answer accuracy for "why" question class

User	Answer accuracy (in %)
1	67
2	73
3	67
4	60
5	73
Average	68

Similarly, the Answer accuracy is found for the other users. The snapshots of some sample questions and their answer(s) given by PQAS have been shown in Appendix-9. The average answer accuracy for the questions belonging to "who" question class has been found to be 68.0%.

9.9 SET-7 (QUESTIONS STARTING WITH "HOW")

The set consisting of 20 questions for "how" question class is shown in table 9.13 along with the answer(s).

The answer(s) were analyzed using the performance metrics *Answer accuracy* by each of the five users and the response obtained thereof is given in Table 9.14 in terms of Answer Accuracy (in %) and its Average.

Table 9.13 Questions-Answers for ”how” class

Question	Answer(s) returned by PQAS
How is gratuity calculated?	Gratuity Calculation India is calculated by the employee's basic salary and the years of working experience.
How is paper made?	To make paper from trees, the raw wood must first be turned into “pulp.” Wood pulp is a watery “soup” of cellulose wood fibers, lignin, water and the chemicals used during the pulping process. Wood can be turned to pulp in a couple of different ways.
How is cotton made?	No answer
How is tax calculated?	Calculate your income tax as per new Budget 2015 - 16. Quick Tax Calculator, Free Income Tax Calculator Tool.
How to prepare tea?	For tea; Boil a mixture of water, milk and sugar (as much as needed for the entire serving). Once that happens, add the tea leaves and spices and continue to heat, turn off the burner and have tea.
How to implement vector class in Java?	The Vector class implements a growable array of objects.
How is vector implemented in C++?	C++ Source Code for a Vector - index based array of objects - implementation is given on this page with example. Vector is usually implemented as a contiguous block of memory.
How is cotton processed?	The cotton gin is where cotton fiber is separated from the cotton seed. The first step in the ginning process is when the cotton is vacuumed into tubes that carry it to a dryer to reduce moisture and improve the fiber quality.
How is vector graphics used?	No answer
How to use internet?	The internet is a global stream of interconnected computer networks that can use the standard Internet protocol suite (TCP/IP) to link several billions devices worldwide.
How to convert dollar to rupee?	As on Saturday, January 17, 2015; 2:25:20 am, there are 61.62 Indian rupees in united states dollar.
How to lose weight?	No answer
How is Christmas celebrated?	How to Celebrate Christmas. Christmas is one of those holidays that just seems to be filled with cheer and wonder.
How is beer made?	Beer is made from four basic ingredients: Barley, water, hops and yeast.

How is Celsius converted to Kelvin?	You can convert between Celsius and Kelvin like this: Kelvin =Celsius + 273.15.
How is gm converted to kg?	There are 1000 gm in one kg.
How is Precision computed?	Precision is the measure of accuracy.
How is Recall computed?	Recall is the measure of coverage.
How is bird flu transmitted?	No answer
How is Buddha Purnima celebrated?	No answer

Based on the feedback of the first user, Out of 20 questions, the system provided answers for 14 questions, out of which answers for 10 questions were found to be satisfactory and for 4 questions, the system has not given satisfactory response. So, using the terms defined above, C=10, W=4.

Using eq. 9.1, Acc =10/(10+4)=71%.

Table 9.14 Average Answer accuracy for "how" question class

User	Answer accuracy (in %)
1	71
2	78
3	71
4	78
5	64
Average	72.4

Similarly, the Answer accuracy is found for the other users. The snapshots of some sample questions and their answer(s) given by PQAS have been shown in Appendix-9. The average answer accuracy for the questions belonging to "who" question class has been found to be 72.4%. The graph shown in Fig. 9.2 shows the values of average Answer Accuracy for each question class. High values of *Answer accuracy Average (i.e. Acc=83.3%)* for various tests conducted on the system indicate that it accurately answers the questions that the user asks.

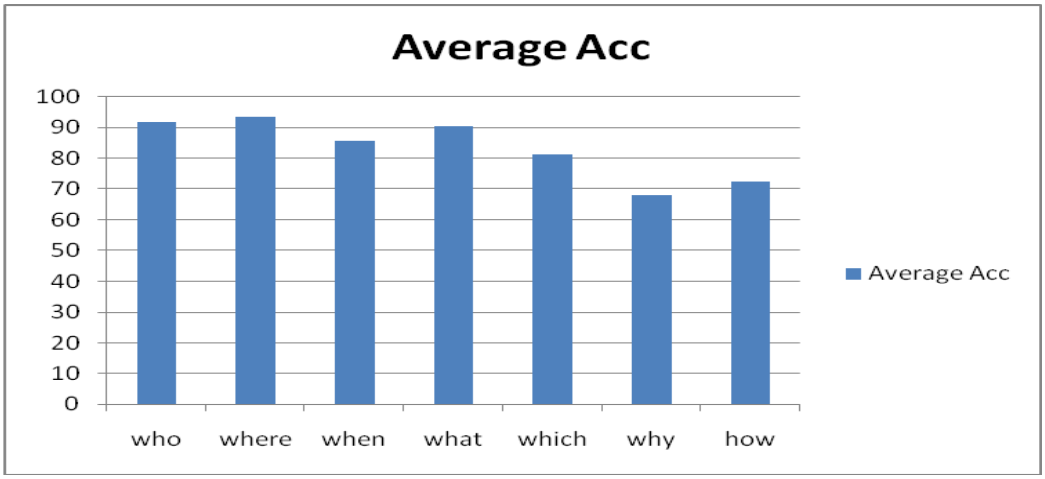


Fig. 9.2 Plotted values of *Average Answer accuracy*

It can be observed from the graph that the plotted values of *Answer accuracy* are higher for "who", "where", "what" and somewhat lower for "when" and "which". For "why" and "how", the plotted values are comparatively low.

9.10 COMPARISON OF PQAS WITH EXISTING QA SYSTEMS

In this work, architecture for *a novel search engine for prospective Question Answering* has been proposed. The accuracy of the proposed system, measured in term of *Precision* is found to be higher as compared to the existing systems for Question Answering as shown in Table 9.15. Thus, the proposed system is able to respond to user's question with high accuracy.

Chapter X

CONCLUSION & FUTURE SCOPE

10.1 CONCLUSION

In this dissertation, an effective technique to answer questions asked by the user has been developed. The main challenges involved in the task have been addressed and resolved.

During this work, many existing systems of question answering were studied with a view to understand the existing question answering mechanisms and some unresolved issues found thereof were addressed and resolved, discussed as follows.

i. **Focus on multiple tasks:** Previous work in this field has not focussed on all the aspects related to question answering like summarization, question analysis, answer extraction or ranking. So, it has being analyzed that there is a need to design and develop a system that is user interactive and combines all the modules together for efficient question answering. So, in the dissertation, novel architecture of a search engine for prospective question answering has been designed and implemented.

ii. **Quality of information:** The referred work in this section collected the pages from the Web. However, it has been found that the information contained in web pages is not of high quality and also, the major portion of the web pages contain information that is not of much importance to the user. So, despite that web is a huge repository of information; it is not likely to contain the topical information i.e. information related to the topic of interest, required for question answering. For this purpose, the system to be developed must be able to collect information from some other sources. Also, there is a need to extract only a relevant portion of the text from the pages collected from the quality sources. In this dissertation a crawler has been designed and implemented that collects pages from blog sources. Average precision, recall and F-measure are found to be equal to **85.0, 81.4 and 83.1** respectively. Also, to ignore the irrelevant portion of the text, two

techniques of extracting relevant content from blog posts have been proposed and implemented. Two performance metrics have been used for the performance evaluation of both the techniques. Precision and recall of the first technique w.r.t. the relevant content given by the expert ranges between **80% to 85.5% and 78.9% to 83.7%** respectively. For the second technique, precision and recall w.r.t. the relevant content generated by the expert lies between **80.5% to 85% and 79.5% to 83%** respectively.

iii. **Indexing of pages:** The existing systems have not much focussed on the indexing of the Web pages for answering the questions. So, there is a need to design an efficient indexing scheme for developing a fast QA system. So, in the dissertation, a novel indexing technique based on *question class* has been proposed and implemented for efficient questing answering. Average ARS has been used as performance metric for all the question classes. It has been observed that average Answer relevance score (ARS) of the answers obtained from the proposed system was found in the range from **70.0% to 100.0%** which shows that the proposed system indexes the relevant content well.

iv. **As an improvement in the proposed PQAS:** The techniques for ranking blog posts based on their popularity features and prediction of user's next questions have been proposed as an improvement to the proposed system of prospective question answering (PQAS).

Summarizing, a design of a novel search engine for prospective question answering [99] has been proposed that not only addresses the problems prevailing in the existing QA systems but also uses the blogosphere as the major source of the topical information.

The system for prospective question answering has been implemented using JDK 1.8. The answer(s) were analyzed using the performance metrics *Answer accuracy.* High values of *Average Answer accuracy (i.e.* **Acc=83.3%***)* for various tests conducted on the system indicate that it accurately answers the questions that the user asks. Also, the classification of the proposed work is done in such a way that a modular architecture is developed with the expectation that new functionalities can easily be added by third

parties according to their requirements.

10.2 FUTURE SCOPE

In this dissertation, the problems related to question answering has been explored extensively. Some of the possible issues that could be further explored or the areas that can be extended in the future are as follows:

- **Handling the questions with some specific question classes:** The work can be extended in the direction to answer the questions that start with "how many", "how much" and "is/am/are".
- **Focus on Semantic web:** The question answering system may be made compatible with the *semantic web*.
- **Keyword expansion:** It is suggested that for a given question, the keyword expansion may be used to extract terms similar to that in the question from the thesaurus and use them for searching through the index to obtain answers.

APPENDIX-1

Survey

1. *Do you like surfing on the Internet?*
   a. *Yes*
   b. *no*
2. *For what purpose, you use Internet?*
   a. *As a source of information*
   b. *For entertainment*
   c. *For fun*
3. *Do you use search engine, for your queries?*
   a. *Yes*
   b. *No*
4. *How often you use search engine?*
   a. *Many times a day*
   b. *Daily*
   c. *Weekly*
   d. *Monthly*
   e. *Never*
5. *Do you think GOOGLE is one of the best search engines?*
   a. *Yes*
   b. *No*
   c. *Cannot say*
6. *Does using a search engine for your queries yield useful results?*
   a. *Yes*
   b. *No*
   c. *sometimes*
7. *Do you use search engine, for asking a question that needs a precise answer?*
   a. *Yes*
   b. *no*
8. *Does the search engine returns precise answers.*
   a. *Yes*
   b. *No*
9. *Are you satisfied with the answer(s) returned by the search engine for your question?*
   a. *Yes*
   b. *no*
10. *Have you ever used a Question answering website (QA system) for answer to your question?*

a. *Yes*
b. *No*

11. *Does the QA system used returns precise answer(s).*
    a. *Yes*
    b. *No*
    c. *Sometimes*
    d. *Cannot say*

12. *Are you satisfied with the answer(s) returned by the QA system for your question?*
    a. *Yes*
    b. *No*
    c. *cannot say*
13. *Can you name some QA systems?*
    a……
    b……
    c……
    d……..
    e. *Don't know*
14. *Do you know blogs?*
    a. *Yes*
    b. *no*
15. *Have you seen a blog?*
    a. *Yes*
    b. *no*
16. *Have you written a blog?*
    a. *Yes*
    b. *no*
17. *What type of blogs have you seen? Which category?*
    a. *Educational*
    b. *technical*
    c. *Recreation*
    d. *Entertainment*
    e. *Health*
    f. *business*
18. *Does blogs contain information about almost everything?*
    a. *Yes*
    b. *no*
19. *Can a blog be helpful for answering your question?*
    a. *Yes*

b. No

c. Cannot say

20. When you search, do you see some links to blogs in the search results?

a. Yes

b. no

21. Do you navigate the blog sites, by clicking on the links, for your search?

a. Yes

b. No

c. Sometimes

After conducting the survey, all the responses were collected as shown in Table A1.1, where Qid is the Question in the survey and the options provided are given by a, b, c, d and e.

Table A1.1 Survey responses

Qid	a	b	C	d	e
1	57	3			
2	20	19	21		
3	45	15			
4	40	10	7	2	1
5	35	10	15		
6	40	10	10		
7	45	15			
8	10	43	7		
9	8	42	9		
10	40	20			
11	38	5	10	7	
12	38	10	12		
13	47	13			
14	45	15			
15	10	50			
16	48	12			
17	47	10	3		
18	45	15			
19	45	10	5		

APPENDIX-2

The following is the list of Blog sites having RSS feed:

1	Buzzerhut	24	Regator	47	Devasp	70	Keegy
2	Feedmap blog	25	Roask	48	Feedagg	71	Metafeeder
3	Feednuts blog	26	SmallBusiness.com	49	Feedage	72	Millionrss
4	Feedplex blog	27	Spicypage	50	Feedcat	73	Moneyhighstreet
5	Flookie	28	Technorati	51	Rssfeeddirectory	74	Newsnow
6	FyberSearch	29	Theseoking	52	Feedest	75	NGOID
7	Google	30	TruthLaidBear	53	Feedgy	76	Oobdoo
8	Gozoof	31	Ubdaily	54	FeedListing	77	Plazoo
9	Grokodile	32	Wilsdomain	55	Feedmailer	78	Rss001
10	Icerocket	33	Webloogle	56	Feedsee	79	Rssmountain
11	Info-listings	34	Webworldindex	57	Feeds4all	80	Rssmotron
12	Leighrss	35	Zimbio	58	Feedzie	81	Rsstop10
13	Loaded	36	4guysfromrolla	59	Finance-investing	82	Rubhub
14	Minnesota blog	37	5z5.com	60	Goldenfeed	83	Scribnia
15	Mozdex	38	9rules.com	61	Guzzle	84	Solarwarp
16	Ontoplist blog	39	Chordata Blog	62	Jordomedia	85	Swoogle
17	Syndic8	40	Ezilon	63	SmallBusiness.com	86	Word.ess
18	Urlfan	41	freshlinkmedia	64	Spicypage	87	Yahoo.com
19	Yopod	42	Google	65	submitlinkurl	88	Zimbo
20	Alltop	43	Loaded Web	66	Theseoking	89	DmozEZE
21	Ckalari	44	Ontoplist	67	Topsiteswebdirectory	90	Regator
22	Conseillemoi	45	PuppyURL	68	Websandiego	91	Wilsdomain
23	Crayon	46	Rateitall	69	Webworldindex		

APPENDIX-3

The following figures show the snapshots of implementation of the proposed blog crawler:

The list of seed URLs given to the Blog crawler as input for crawling is shown in the following A3.1.

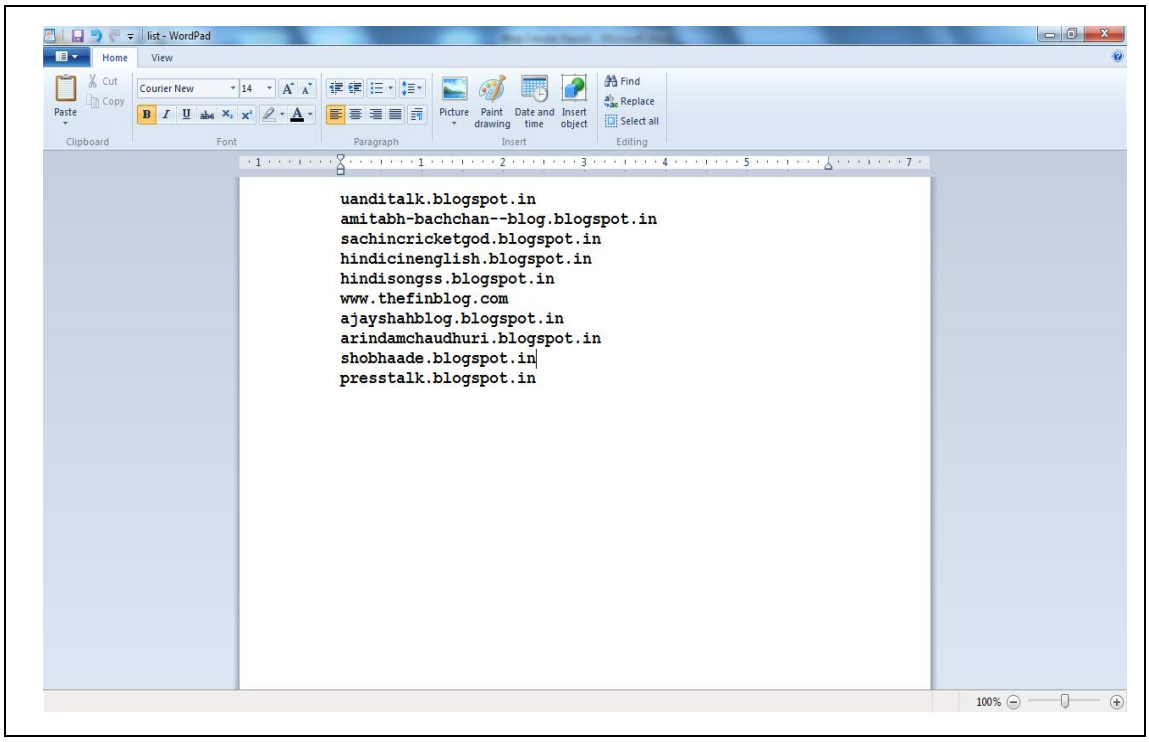


Fig. A3.1. Input seed URLs

This list of seed URLs is provided to the blog crawler to start the crawling process. Also, a new URL can be added in this list as shown in Fig. A3.1.

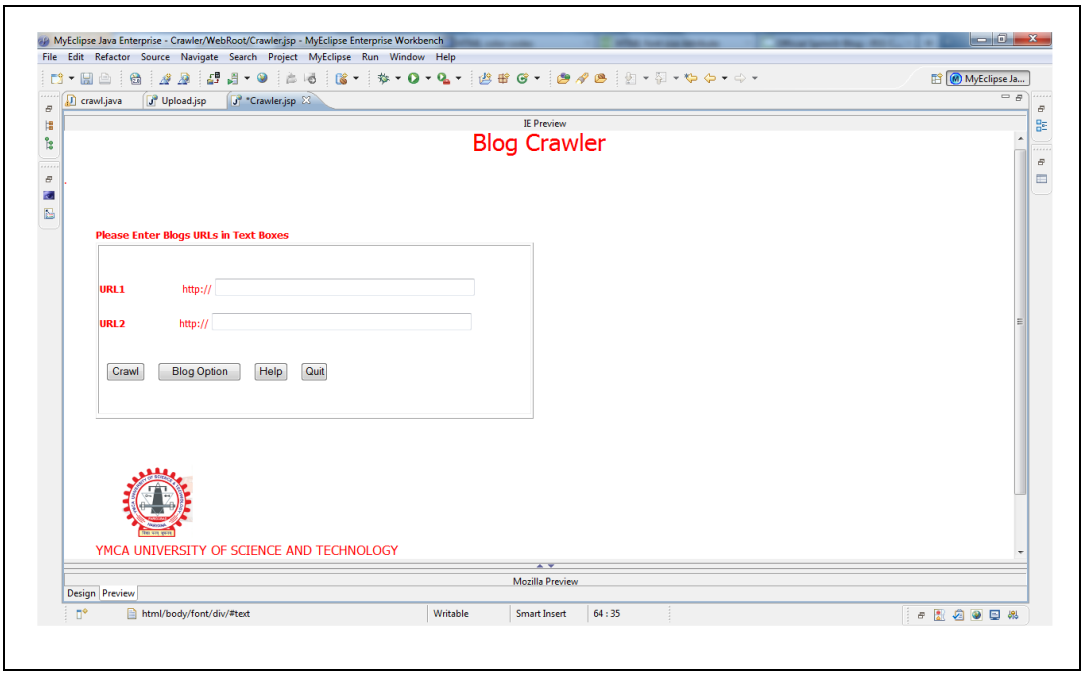


Fig. A3.2 Addition of new URL in the list

The snapshot shown in Fig.A3.2 shows the addition of a new URL to the existing list of URLs. To update the list of URLs to be assigned to the Blog crawler for crawling.

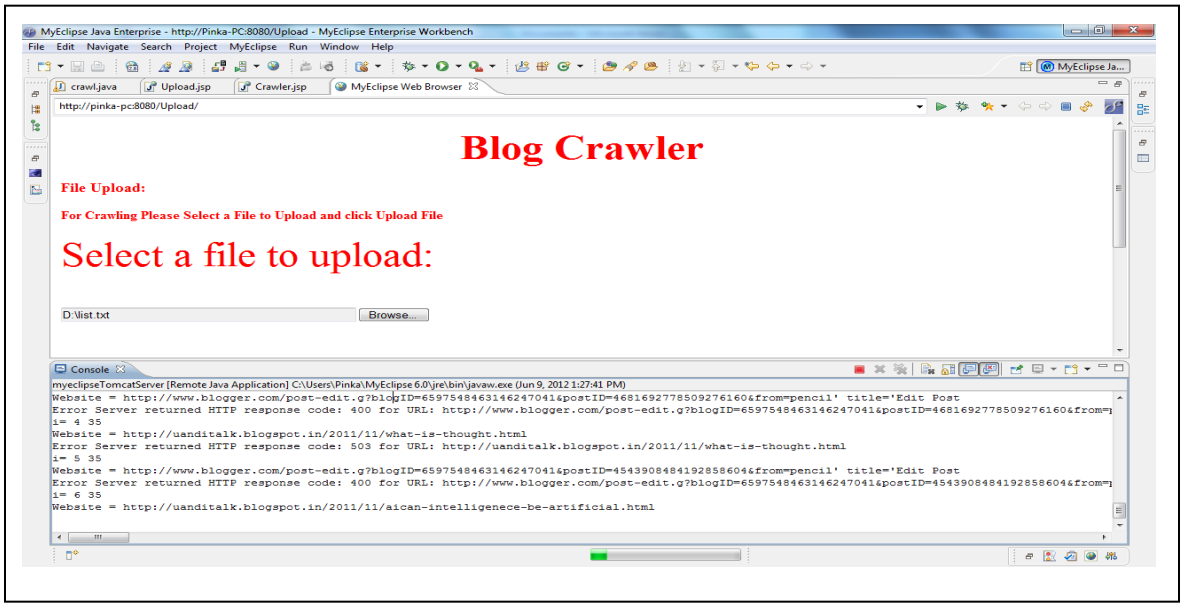


Fig. A3.3 File consisting of seed URLs to be uploaded

The snapshot shown in Fig. A3.3 shows the uploading of the file that consists of seed URLs to crawl. The Blog crawler takes each URL one by one and downloads the page corresponding to it.

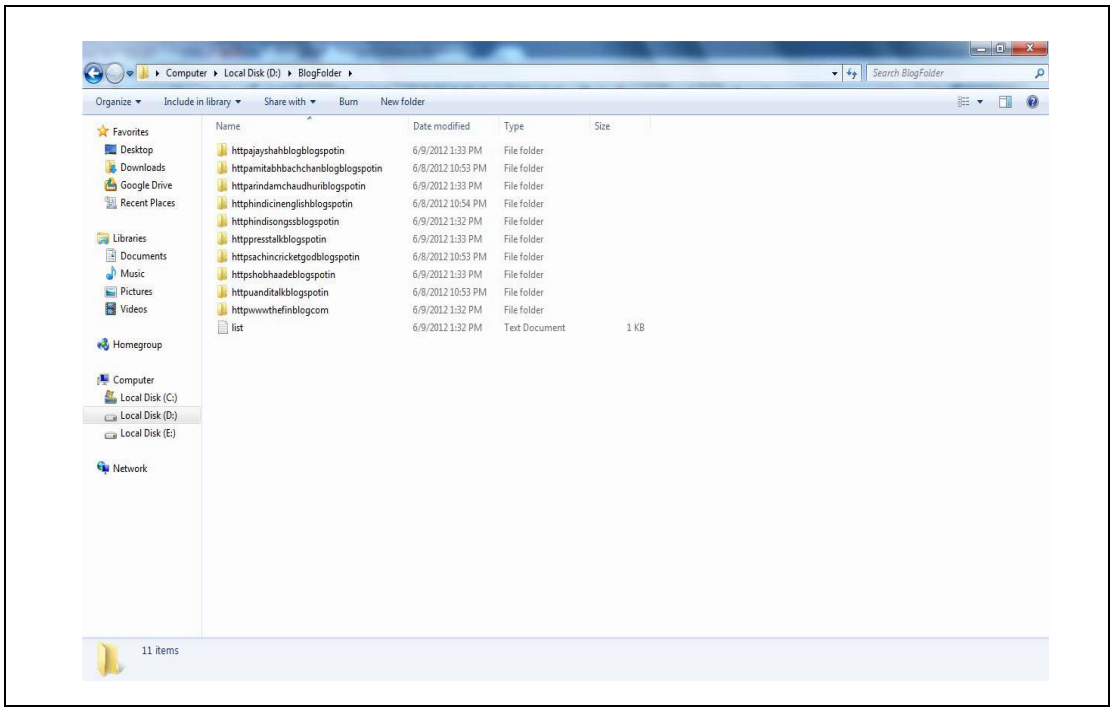


Fig. A3.4. List of blog pages that have been downloaded

The snapshot shown in Fig. A3.4 shows the list of blog pages that have been downloaded by the Blog crawler.

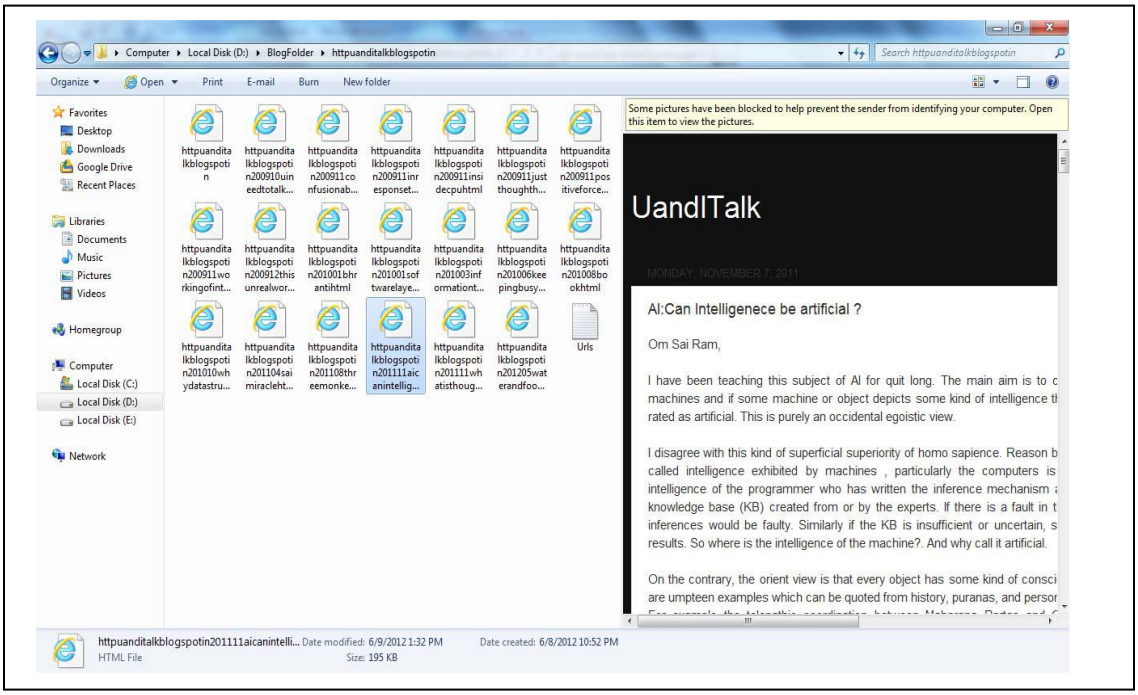


Fig. A3.5 View of directory containing all the blog pages

Fig. A3.5 shows the directory that consists of blog pages that have been downloaded by the Blog crawler. Also, the blog is shown on the right hand side of the directory view.

APPENDIX-4

The relevant content generated as the result of implementation of the proposed techniques of relevant content extraction is given below:

Example1. URL: http://www.ashishsharma.me/2011/08/java-garbage-collection-notes.html

Title: Understanding Java Garbage Collection and its algorithms.

Post content: Here is a deeper dive into - Understanding Java Garbage Collection and its algorithms. This article is in continuation to my article on "Understanding Java Heap Space and Memory Tuning". Make sure you understand the concepts behind Java heap usage before diving into garbage collection details.
We cannot force Garbage Collection, but could request it by calling System.gc() or Runtime.gc(). JVM does not guarantee that GC will be started immediately. Also, Garbage collection does not guarantee that a program will not run out of memory. JVM automatically removes the unused variables/objects from the memory... to create free space for the programs that are currently running.
Garbage Collection runs on Heap and NOT stack.
The purpose of finalization is to give an unreachable object the opportunity to perform any cleanup processing before the object is garbage collected. We usually nullify the object references of large objects like collections, maps etc in the finalize block.
An object's finalize() method cannot be invoked by the garbage collector while the object is still reachable. However, an object's finalize() method may be invoked by other objects.
An object's finalize() method may only be invoked once by the garbage collector.
A reachable object cannot be garbage collected. Only unreachable objects may be garbage collected. The system will not do it and you cannot force it either.
An unreachable object may become reachable again. This can happen when the object's

finalize() method is invoked and the object performs an operation which causes it to become accessible to reachable objects.

Relevant content generated: Here is a deeper dive into - Understanding Java Garbage Collection and its algorithms. Make sure you understand the concepts behind Java heap usage before diving into garbage collection details. This article is in continuation to my article on "Understanding Java Heap Space and Memory Tuning". We cannot force Garbage Collection, but could request it by calling System.gc() or Runtime.gc(). Also, Garbage collection does not guarantee that a program will not run out of memory.

Example 2: URL: Genetic Algorithmshttps://raygun.io/blog/2014/10/5-interesting-data-structures-algorithms/

Title: Genetic Algorithms

Content: Genetic algorithms are a solution to optimisation problems inspired by biological evolutionary adaption. Genetic algorithms usually involve a generational approach to evolving a set of genetic information to achieve the best result at a specific task. The task is usually posited as a fitness function, which returns a value determining how well the actor achieved. For example one may wish to evolve a solution for optimising the distribution of resources. The fitness function may return the cost of the distribution when given the genetic information of the actor. Genetic algorithms can range from being very complicated to very simple.

Some may have genomes with a fixed number of genes. For instance (in an absurdly simple example) an algorithm evolving the optimum angle to throw a ball may have only the angle of the throw as it’s genetic code. More complicated algorithms could have instructions encoded within the genetic code, such as a simple syntax. In this case behaviours could be evolved. The fitness function is the other core component of Genetic Algorithms; it takes the genetic information and runs a calculation or simulation to determine how fit the genetic code is. It can be as simple as a small maths equation or as complex as a physical simulation. Once fitness has been determined for each participant

the weaker participants are removed from the gene pool and the strong ones are cross bred (have their genetic code swapped with other succeeding actors) and randomly mutated. This allows for additional convergence on the true optimal solution without getting stuck in local minima.

Comments:

1. Yes, these are a solution to optimization problems.

Relevant content generated: The fitness function is the other core component of Genetic Algorithms; it takes the genetic information and runs a calculation or simulation to determine how fit the genetic code is. Genetic algorithms are a solution to optimisation problems inspired by biological evolutionary adaption. Genetic algorithms usually involve a generational approach to evolving a set of genetic information to achieve the best result at a specific task. For instance (in an absurdly simple example) an algorithm evolving the optimum angle to throw a ball may have only the angle of the throw as it is genetic code.

For example one may wish to evolve a solution for optimising the distribution of resources.

Example 3: URL: http://blog.betafamily.com/2014/07/08/testing-techniques-black-white/

Title: Testing techniques – difference between white box and black box testing

Post content: We explain the difference between white box and black box testing, followed by some helpful and efficient testing tricks and techniques. Along the way we also define some names and words.
A common distinction between two vastly different types of software testing, is that between *white box* and [glossary slug='black-box-testing']*black box*[/glossary]. You may associate the latter term with plane crashes, but in this case it denotes that you are testing

the software without any inside knowledge about source code, architecture or internal design. This is most likely the case with any app you may test on The Beta Family.

White box testing, on the other hand, is mainly used by developers with access to the source code itself. Distinct code units can be separately tested by writing *unit tests* that assert expected output for given input data. These tests often strive to cover as many *if statements* and *code branches* as possible. The measurement of this is called *code coverage*, often defined as the percentage of code lines covered by test cases.

While the general consensus is that developers theoretically would like 100 % code coverage, it is also recognized that it is realistically very hard to achieve – it would require defining the expected results for all combinations of input, variables and states. An advantage however is the reusability of unit tests, which makes them very suitable for automation. At this point I should also point out that there are other types of white box tests, even those that are based on access to documentation rather than actual source code.

Let us move the focus back to black box testing, as this is more relevant to tests and testers on The Beta Family. When testing from a black box perspective, much revolves around the functionality itself. As Wikipedia defines the technique: "The tester is aware of *what* the software is supposed to do but is not aware of *how* it does it". So if you are an app developer publishing your app for testing on The Beta Family, it is extremely helpful for the tester if you attach some kind of documentation over expected functionality. Without it, you are essentially relying on the tester's personal opinion of how the app should work. This may be good or bad: it could produce more "bug" reports than necessary, but also give you unbiased first impressions.

Relevant content generated: A common distinction between two vastly different types of software testing, is that between *white box* and [glossary slug='black-box-testing']*black box*[/glossary]. *We explain the difference between white box and black box testing, followed by some helpful and efficient testing tricks and techniques.* Let us move the focus back to black box testing, as this is more relevant to tests and testers on The Beta Family. When testing from a black box perspective, much revolves around the functionality itself. White box testing, on the other hand, is mainly used by developers with access to the source code itself.

Example 4: URL: http://javahungry.blogspot.com/2013/04/scheduling-algorithm-first-come-first.html

Title: Scheduling Algorithm: First Come First Serve (fcfs) Java Program Code

Post content: Scheduling algorithm is used by CPU scheduler to select a process. There are many types of scheduling algorithm but we will discuss about the most common algorithm FCFS i.e. First come and First Serve. By applying this scheduling algorithm , the CPU makes sure that the process which is run by user are lined in queue , just like the queue for buying tickets of movie . The person who comes first , will have the chance to get the ticket , similarly , if CPU is idle and CPU is using First come and First Serve algorithm then ,it executes the process which arrives first. .
Read Also: Round Robin Scheduling Algorithm with Example.
Here , User can calculate the average turnaround time and average waiting time along with the starting and finishing time of each process
Turnaround time : Its the total time taken by the process between starting and the completion
Waiting time : Its the time for which process is ready to run but not executed by CPU scheduler.

Relevant content generated: There are many types of scheduling algorithm but we will discuss about the most common algorithm FCFS i.e. First come and First Serve. The person who comes first will have the chance to get the ticket. Similarly, if CPU is idle and CPU is using First come and First Serve algorithm then ,it executes the process which arrives first. Scheduling algorithm is used by CPU scheduler to select a process. By applying this scheduling algorithm , the CPU makes sure that the process which is run by user are lined in queue , just like the queue for buying tickets of movie . Read Also: Round Robin Scheduling Algorithm with Example.

APPENDIX-5

Survey

The survey has been conducted to know the factors people find relevant for the answer types.

What you need to know about:

(Please tick mark)

1. *Location-*
    - I. *Where it is located?*
    - II. *Whether it is a city, state , country etc.*
    - III. *What is its population?*
    - IV. *It is nearby to……*
    - V. *Its historical significance*
    - VI. *Its current temperature*
    - VII. *Others please specify……*
2. *Person-*
    - I. *Person's Name*
    - II. *Education*
    - III. *Birth place/native place*
    - IV. *His/Her contribution to the country as a leader, employer, scientists or others*
    - V. *When he/she was born/died*
    - VI. *Others please specify……*
3. *Organization-*
    - I. *Name*
    - II. *Owner*
    - III. *Location*
    - IV. *Year in which it was established*
    - V. *Whether recognized internationally or not?*
    - VI. *Its historical significance*
    - VII. *Its product*

VIII. *Others please specify……*

4. *Day (Tuesday, Wednesday etc.)-*
   I. *It is 1st,2nd ,… or 7th day of the week*
   II. *How it is pronounced?*
   III. *Its spelling*
   IV. *It comes after……..*
   V. *It may be known for something in India/other countries*
   VI. *Its significance –religious, historical etc.*
   VII. *Rahukaal of that day*
   VIII. *Others please specify……*
5. *Month (Jan, Feb etc.)-*
   I. *It is 1st,2nd ,… or 12th month of the year*
   II. *Number of day in this month*
   III. *How it is pronounced?*
   IV. *Its spelling*
   V. *It comes after……..*
   VI. *It may be known for something in India/other countries*
   VII. *Its significance –religious, historical etc.*
   VIII. *Others please specify……*
6. *Number (one, two etc.)*
   I. *It is a number/amount/count etc.*
   II. *How it is pronounced?*
   III. *Its spelling*
   IV. *How it is formed?*
   V. *What it signifies actually?*
   VI. *Others please specify……*
7. *Year (1997, 2007, 1947 etc.)*
   I. *It is an year*
   II. *What event took place in the year?*
   III. *Its significance in history*
   IV. *It comes after……*

V. *Total number of days in this year*

VI. *Others please specify……..*

8. *Abbreviation (NAT,WHO etc.)*
   I. *Its full form*
   II. *It is short form of*
   III. *What it is actually?*
   IV. *Its basic concept*
   V. *If an organization; its location, owner, year and for what it works*
   VI. *Others please specify……..*
9. *Description of a term (Computer, object, protocol, apple etc.)*
   I. *Its definition*
   II. *Its concept*
   III. *Its meaning in different contexts*
   IV. *Its short form*
   V. *Its synonyms/ hypernyms/ antonyms/ meronymns etc.*
   VI. *Its spelling*
   VII. *Its pronunciation*
   VIII. *Its use as a noun, verb, adjective ,its plural, its singular or some other form*
   IX. *Others please specify……..*
10. *If your question starts with how, then what you need to know from the answer?*
    I. *Process/procedure*
    II. *Others please specify……..*
11. *If your question starts with why, then what you need to know from the answer?*
    I. *Reason*
    II. *Others please specify……..*

There were sixty participants in total. See table A5.1 for the responses given.

Table A5.1 Most relevant factors for the answer tpes

Answer type	**Most relevant factors**	**Number of participants that find the factors as relevant**
Location	• *Where it is located?*	60
	• *Whether it is a city, state , country etc.*	58
	• *What is its population?*	47
	• *It is nearby to……*	58
	• *Its historical significance*	50
	• *Its current temperature*	38
Person	• *Person's Name*	60
	• *Education*	58
	• *Birth place/native place*	57
	• *His/Her contribution to the country as a leader, employer, scientists or others*	54
	• *When he/she was born/died*	52
Organization	• *Name*	59
	• *Owner*	58
	• *Location*	58
	• *Year in which it was established*	55
	• *Whether recognized internationally or not?*	50
	• *Its historical significance*	48
	• *Its product*	56

Day	• *It is 1st,2nd ,… or 7th day of the week* • *How it is pronounced?* • *Its spelling* • *It comes after……..* • *It may be known for something in India/other countries* • *Its significance –religious, historical etc.* • *Rahukaal of that day*	56 40 31 55 54 54 52
Month	• *It is 1st,2nd ,… or 12th month of the year* • *Number of days in that month* • *How it is pronounced?* • *Its spelling* • *It comes after……..* • *It may be known for something in India/other countries* • *Its significance –religious, historical etc.*	55 56 23 28 40 54 56
Number	• *It is a number/amount/count etc.* • *How it is pronounced?* • *Its spelling* • *How it is formed?* • *What it signifies actually?*	56 25 28 56 54
Year	• *It is an year* • *What event took place in the year?* • *Its significance in history* • *It comes after……..*	49 55 54 48

	• *Total number of days in this year*	56
Abbreviation	• *Its full form* • *It is short form of* • *What it is actually?* • *Its basic concept* • *If an organization; its location, owner, year and for what it works*	55 53 48 55 53
Description of a term	• *Its definition* • *Its concept* • *Its meaning in different contexts* • *Its short form* • *Its synonyms/ hypernyms/ antonyms/ meronymns etc.* • *Its spelling* • *Its pronunciation* • *Its use as a noun, verb, adjective ,its plural, its singular or some other form*	57 56 46 48 54 40 38 54
How	*Process/procedure*	57
Why	*Reason*	58

APPENDIX-6

Fig. A6.1 to Fig. A6.14 shows the snapshots of searching in *Question classified index* for *Question Answering:*

Example 1: for "who" Question class

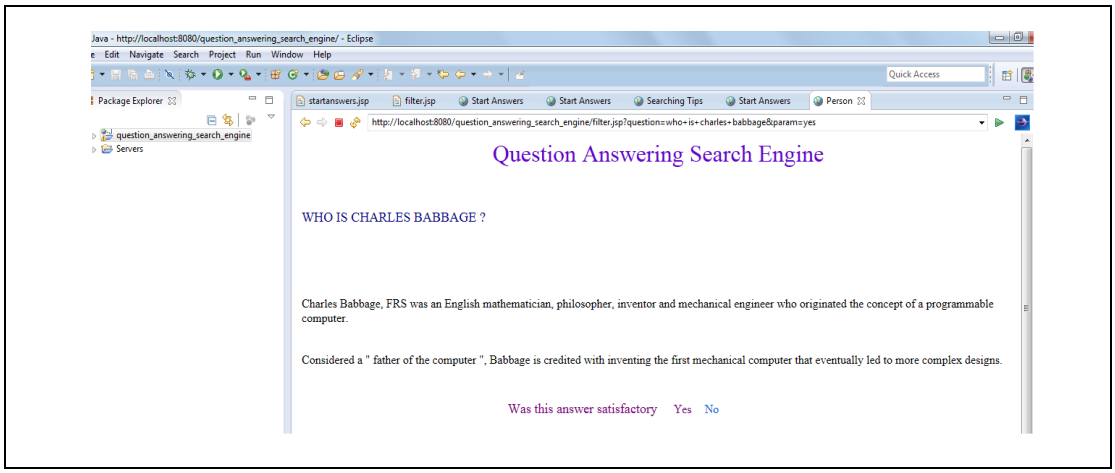


Fig. A6.1 For question "who is Charles babbage"

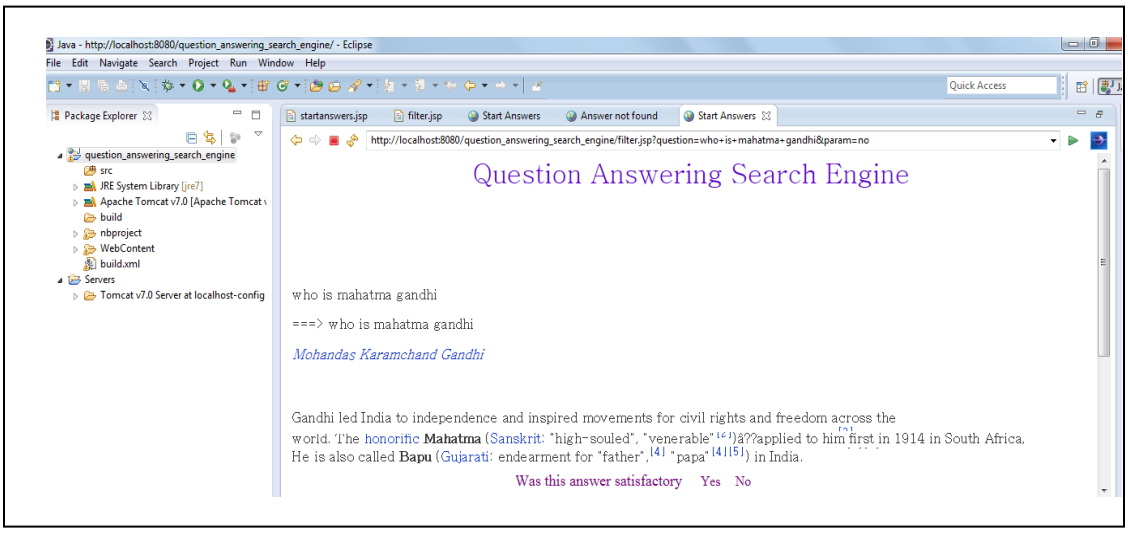


Fig. A6.2 For question "who is Mahatma Gandhi"

Example 2: for "how" question class

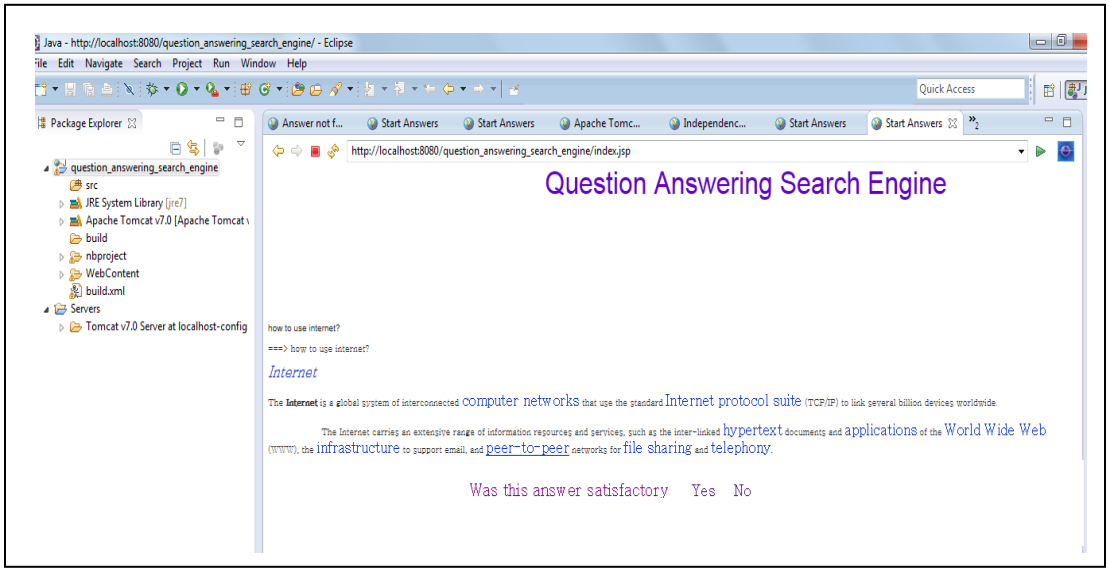


Fig. A6.3 For question "how to use Internet"

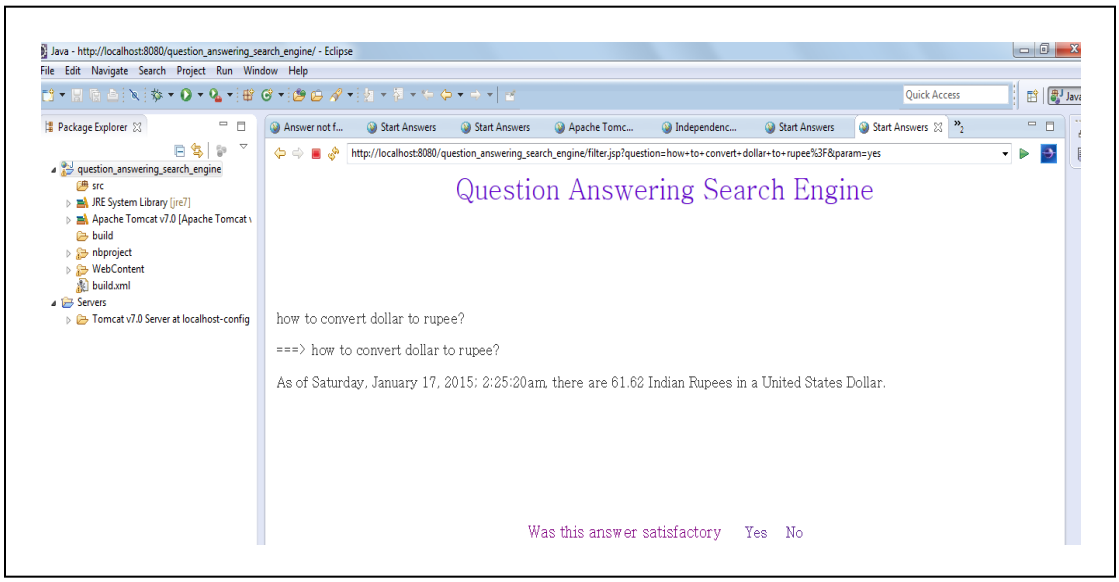


Fig.A6. 4 For question "how to convert dollar to rupee"

Example 3: for "what" question class

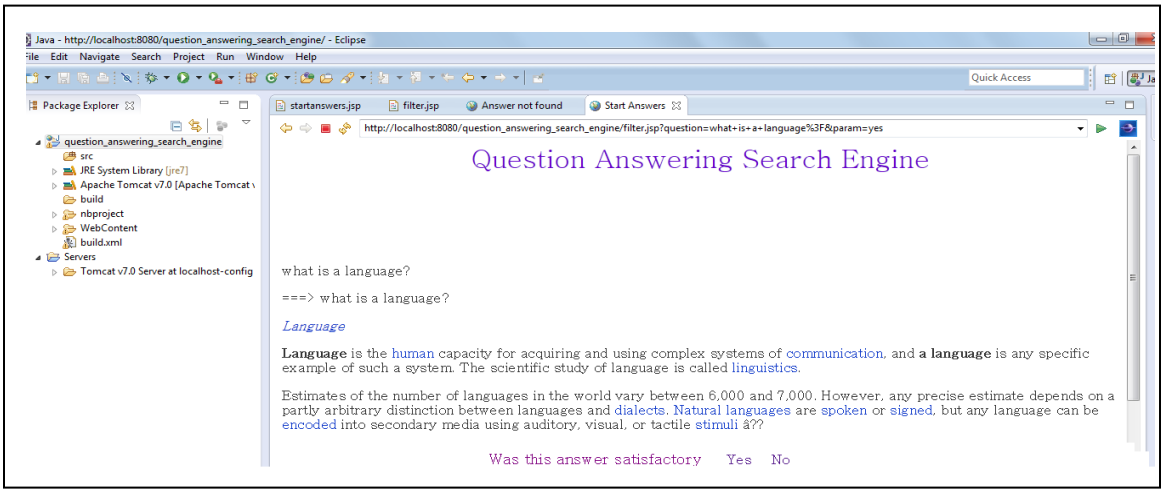


Fig.A6. 5 For question "what is a language"

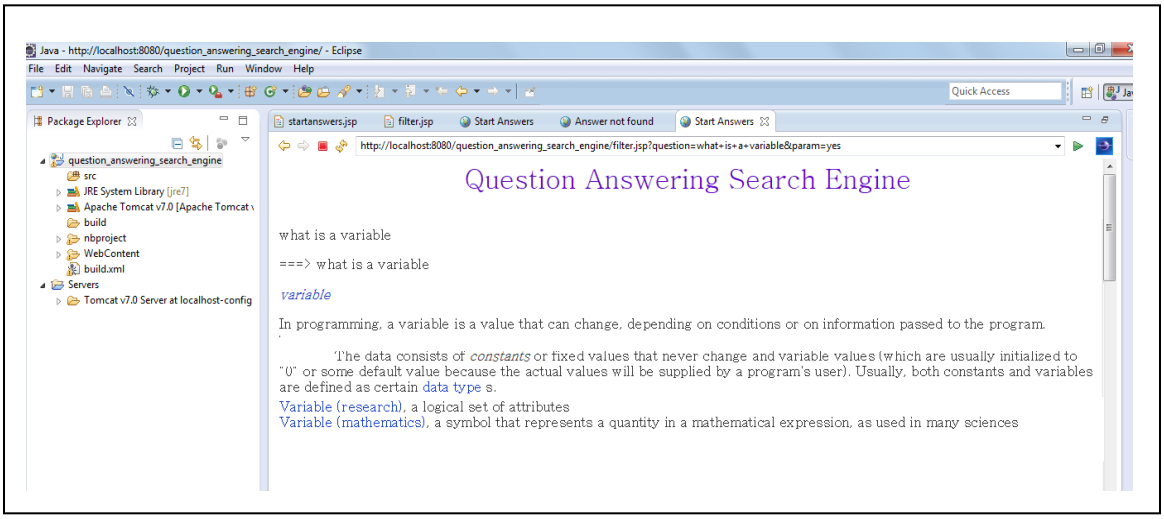


Fig.A6. 6 For question "what is a variable"

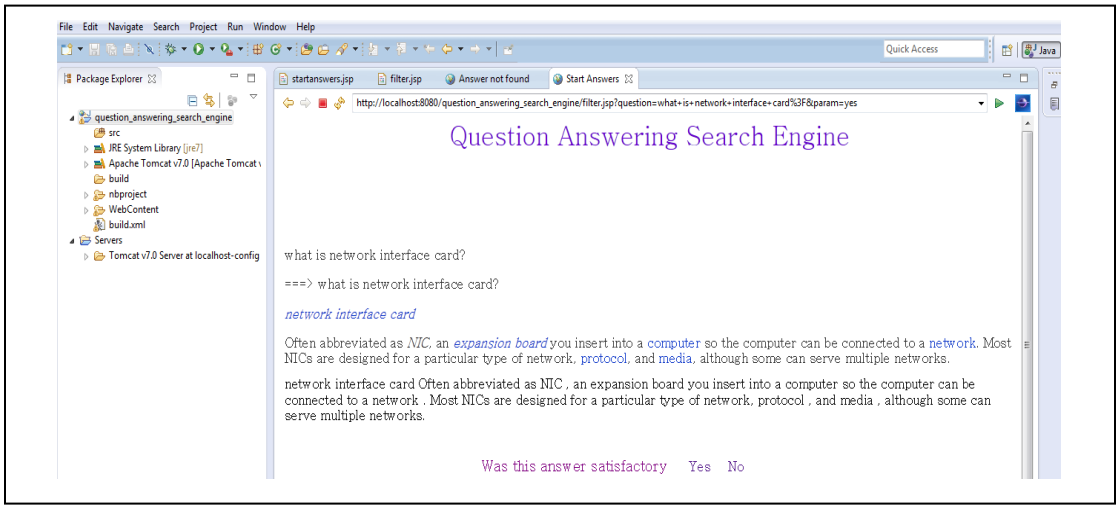


Fig.A6. 7 For question "what is network interface card"

Example 4: for "where" question class

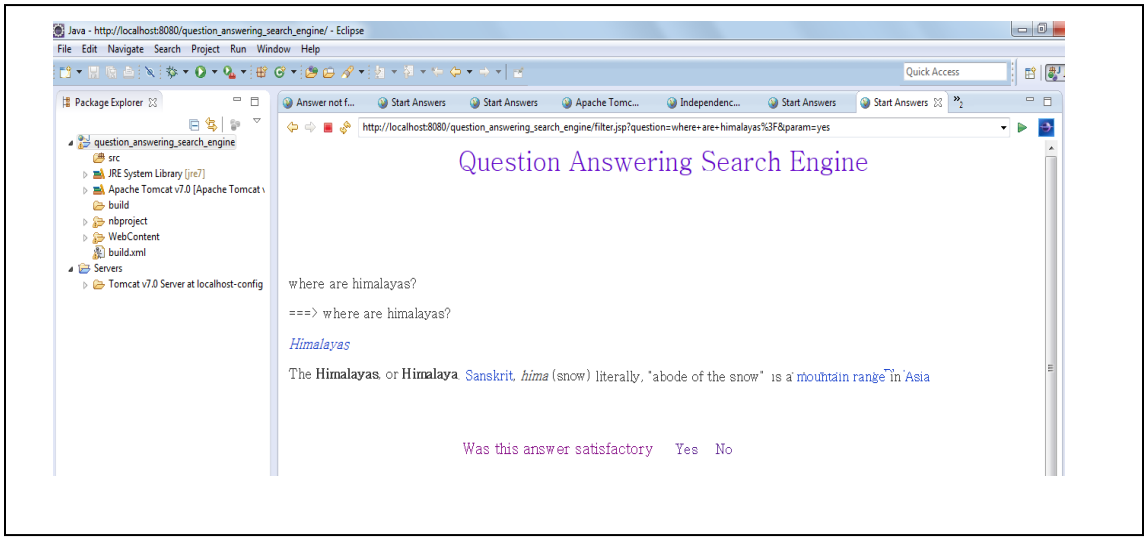


Fig.A6. 8 For question "where are himalayas"

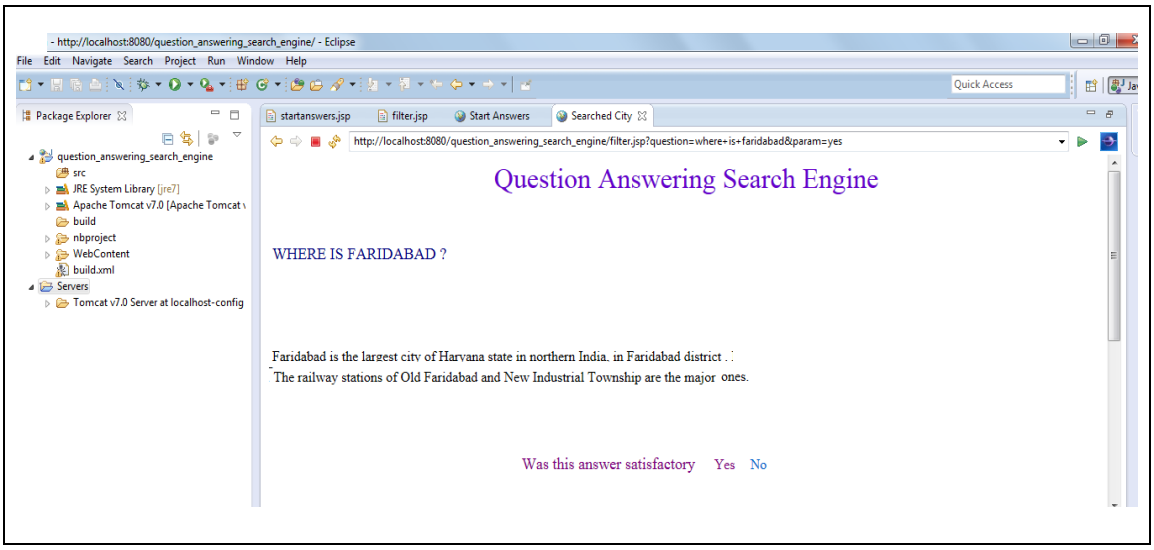


Fig. A6.9 For question "where is faridabad"

Example 5: for "when" question class

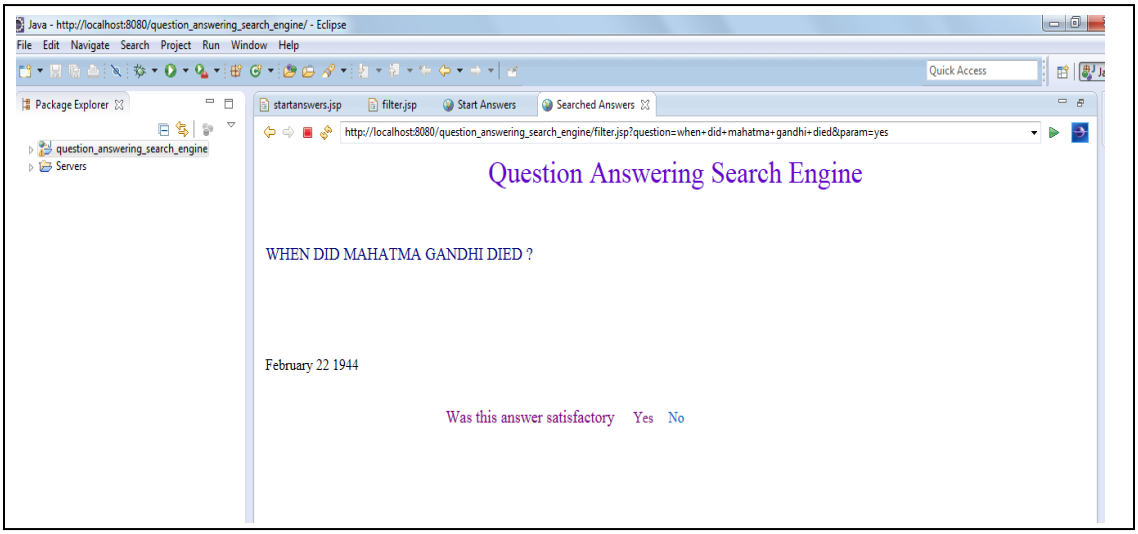


Fig. A6.10 For question "when did Mahatma Gandhi died"

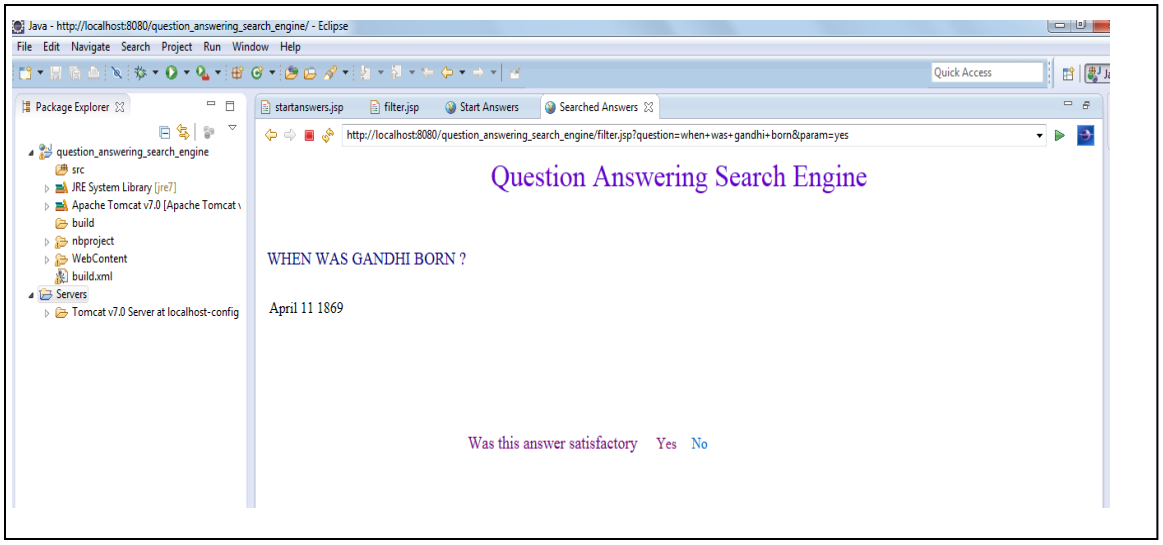


Fig. A6.11 For question "when was Gandhi born"

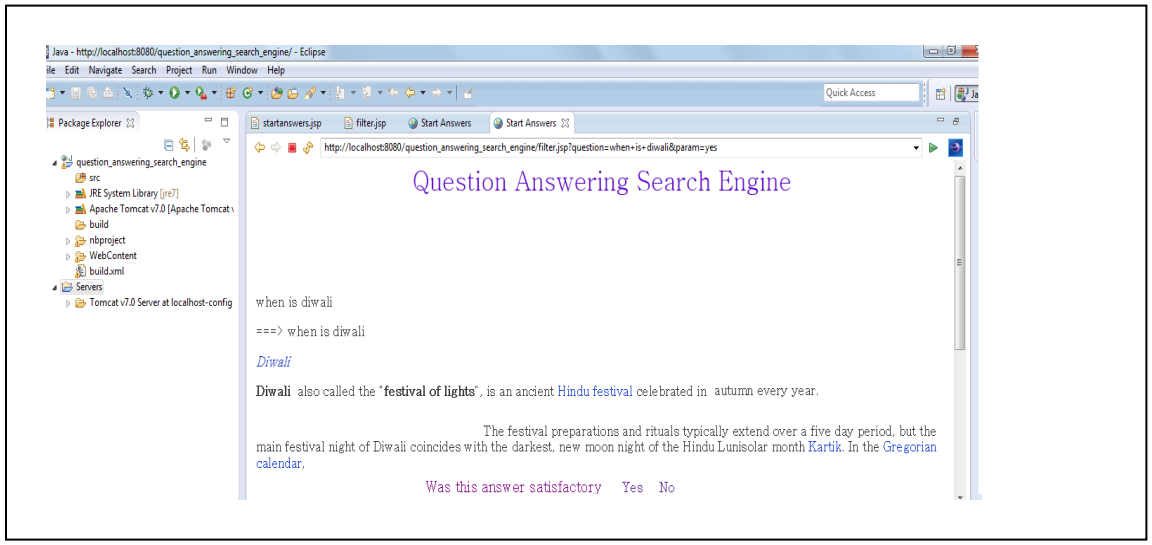


Fig. A6.12 For question "when is Diwali"

Example 6: for "which" question class

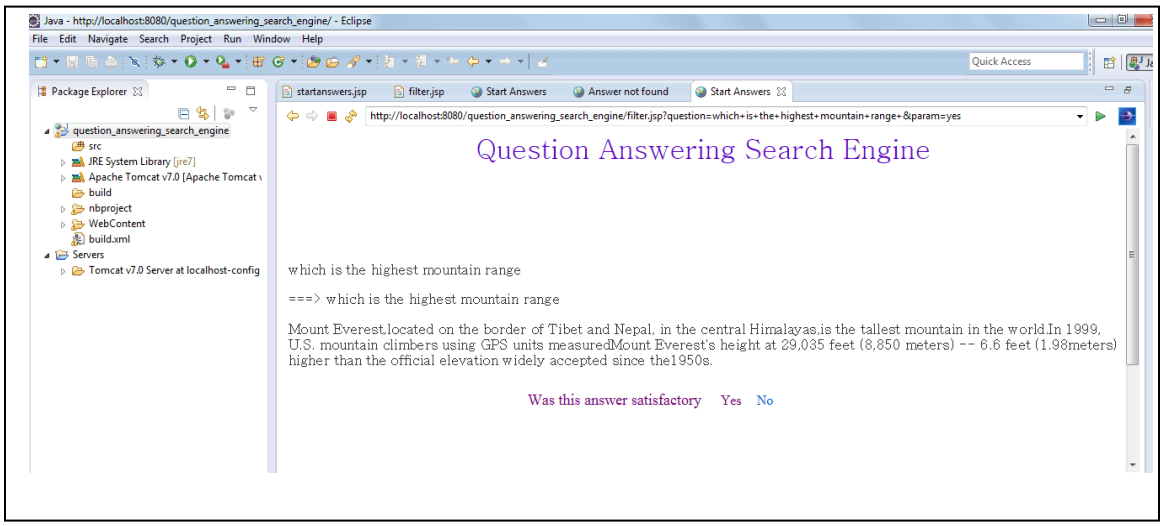


Fig. A6.13 For question "which is the highest mountain range"

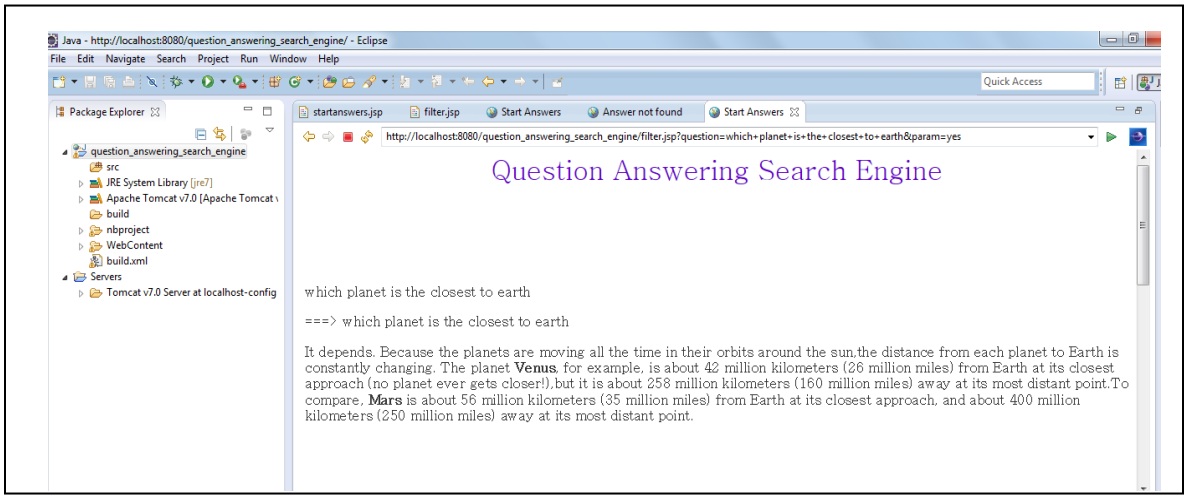


Fig. A6.14 For question "which planet is closest to the Earth"

APPENDIX-7

The following (Fig. A7.1 to Fig. A7.5) are the snapshots of some sample blog pages with popularity features:

Sample 1: URL: insidetech.monster.com/benefits/articles/8537-10-best-tech-blogs

Fig. A7.1 Sample page 1

Sample 2: URL: http://thenextweb.com/apps/2013/08/16/carrot-alarm-for-iphone-whatever-you-do-dont-hit-the-snooze-button/

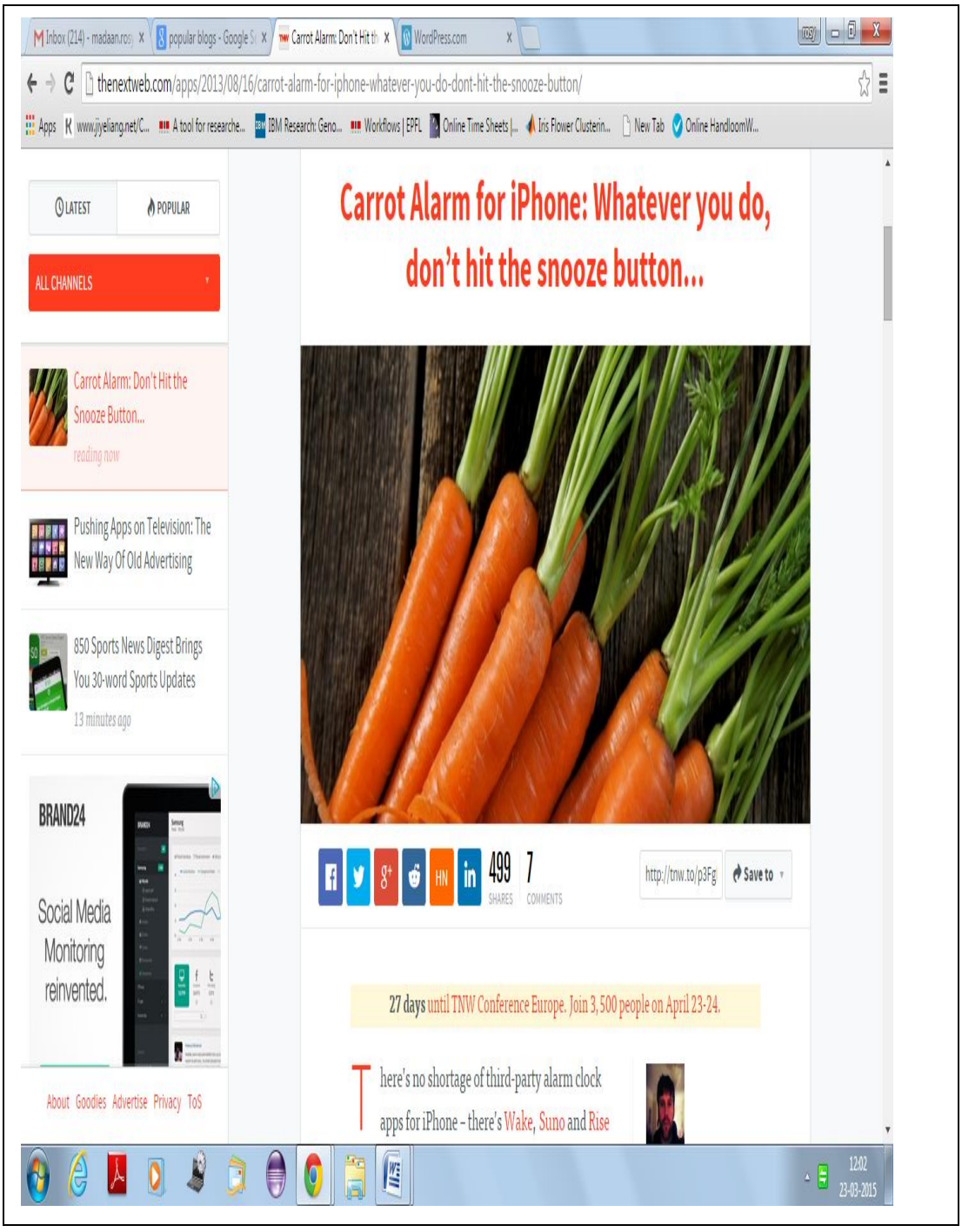


Fig. A7.2 Sample page 2

Sample 3. URL: http://www.huffingtonpost.in/ajay-shah/whats-interesting-in-budg_b_6920768.html?utm_hp_ref=india#

Fig. A7.3 Sample page 3

Sample 4. URL: http://annie.rydholm.org/thoughts-about-test

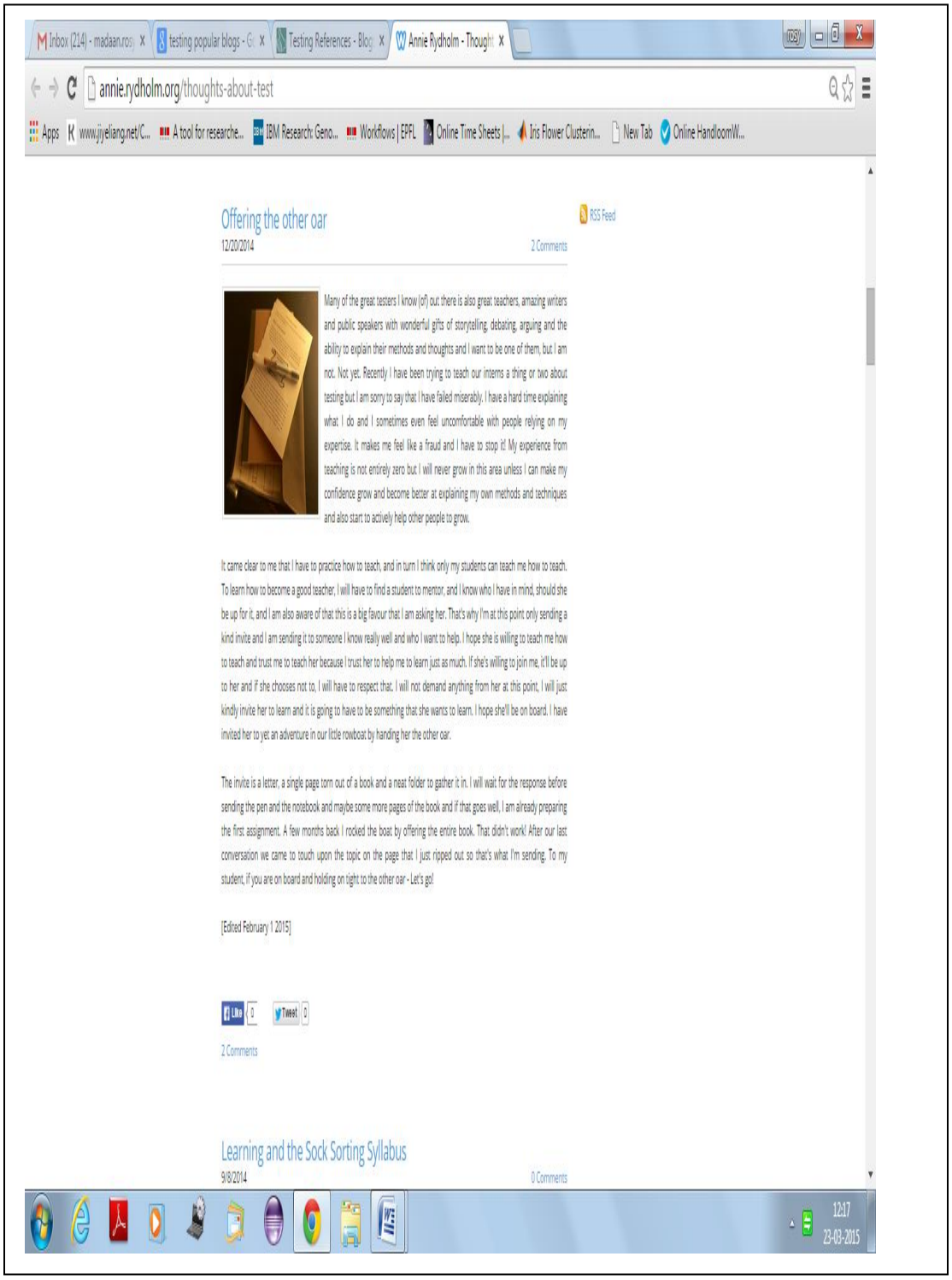


Fig. A7.4 Sample page 4

5. URL: http://blog.xebia.com/2014/11/06/mutation-testing-how-good-are-your-unit-tests/

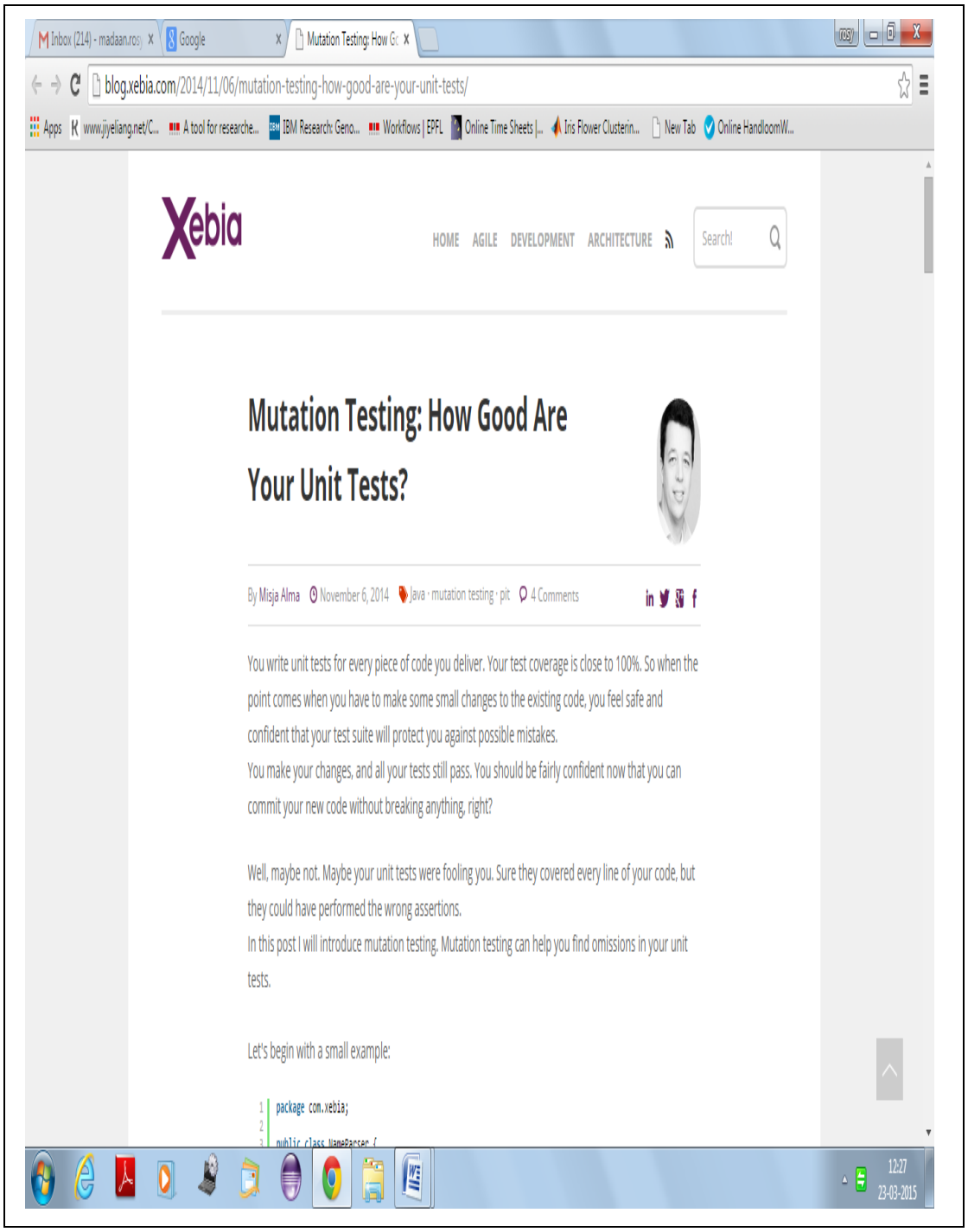
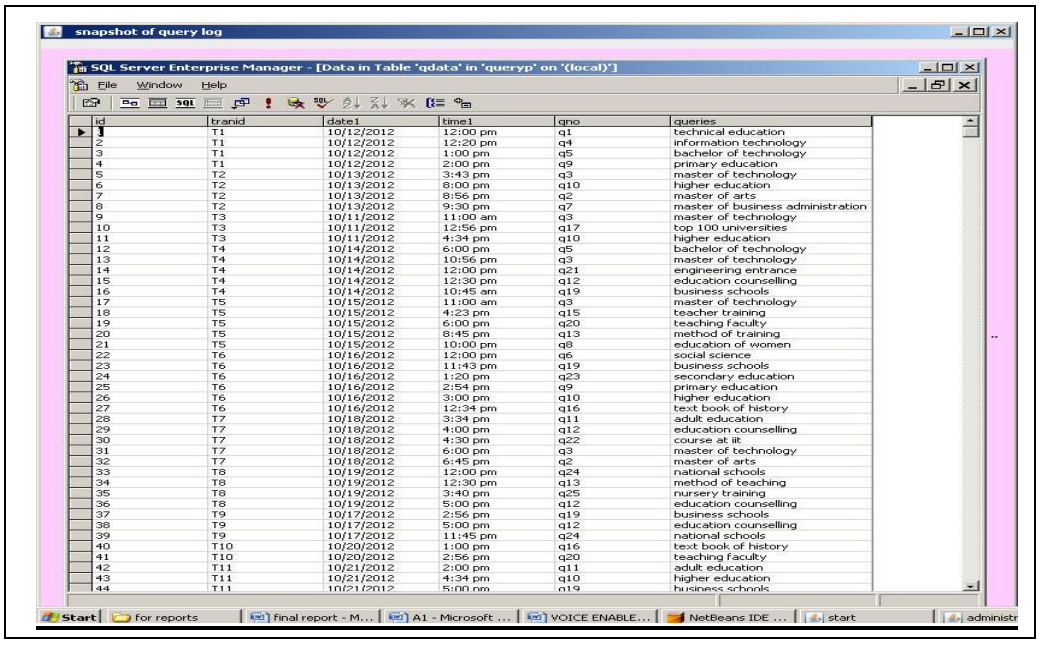


Fig. A7.5 Sample page 5

APPENDIX-8

The snapshots of the implementation of the proposed system for next question prediction are shown as follows:

Fig. A8.1 shows all user questions and their interactions as stored in the *Questions log.*



id	tranid	date1	time1	qno	queries
1	T1	10/12/2012	12:00 pm	q1	technical education
2	T1	10/12/2012	12:20 pm	q4	information technology
3	T1	10/12/2012	1:00 pm	q5	bachelor of technology
4	T1	10/12/2012	2:00 pm	q9	primary education
5	T2	10/13/2012	3:43 pm	q3	master of technology
6	T2	10/13/2012	8:00 pm	q10	higher education
7	T2	10/13/2012	8:56 pm	q2	master of arts
8	T2	10/13/2012	9:30 pm	q7	master of business administration
9	T3	10/11/2012	11:00 am	q3	master of technology
10	T3	10/11/2012	12:56 pm	q17	top 100 universities
11	T3	10/11/2012	4:34 pm	q10	higher education
12	T4	10/14/2012	6:00 pm	q5	bachelor of technology
13	T4	10/14/2012	10:56 pm	q3	master of technology
14	T4	10/14/2012	12:00 pm	q21	engineering entrance
15	T4	10/14/2012	12:30 pm	q12	education counselling
16	T4	10/14/2012	10:45 am	q19	business schools
17	T5	10/15/2012	11:00 am	q3	master of technology
18	T5	10/15/2012	4:23 pm	q15	teacher training
19	T5	10/15/2012	6:00 pm	q20	teaching faculty
20	T5	10/15/2012	8:45 pm	q13	method of training
21	T5	10/15/2012	10:00 pm	q8	education of women
22	T6	10/16/2012	12:00 pm	q6	social science
23	T6	10/16/2012	11:43 pm	q19	business schools
24	T6	10/16/2012	1:20 pm	q23	secondary education
25	T6	10/16/2012	2:54 pm	q9	primary education
26	T6	10/16/2012	3:00 pm	q10	higher education
27	T6	10/16/2012	12:34 pm	q16	text book of history
28	T7	10/18/2012	3:34 pm	q11	adult education
29	T7	10/18/2012	4:00 pm	q12	education counselling
30	T7	10/18/2012	4:30 pm	q22	course at iit
31	T7	10/18/2012	6:00 pm	q3	master of technology
32	T7	10/18/2012	6:45 pm	q2	master of arts
33	T8	10/19/2012	12:00 pm	q24	national schools
34	T8	10/19/2012	12:30 pm	q13	method of teaching
35	T8	10/19/2012	3:40 pm	q25	nursery training
36	T8	10/19/2012	5:00 pm	q12	education counselling
37	T9	10/17/2012	2:56 pm	q19	business schools
38	T9	10/17/2012	5:00 pm	q12	education counselling
39	T9	10/17/2012	11:45 pm	q24	national schools
40	T10	10/20/2012	1:00 pm	q16	text book of history
41	T10	10/20/2012	2:56 pm	q20	teaching faculty
42	T11	10/21/2012	2:00 pm	q11	adult education
43	T11	10/21/2012	4:34 pm	q10	higher education
44	T11	10/21/2012	5:00 pm	q19	business schools

Fig. A8.1 Snapshot showing *Questions log*

Then an appropriate value of t is chosen say t=10. In the gap of t units of time, then the users' sessions are extracted. The questions extracted from the user sessions, are converted to form queries. Considering the queries as transactions, then the system applies *A-priori algorithm* to extract the *frequent itemsets* as shown in Fig. A8.2. From the *frequent itemsets*, the association rules are generated.

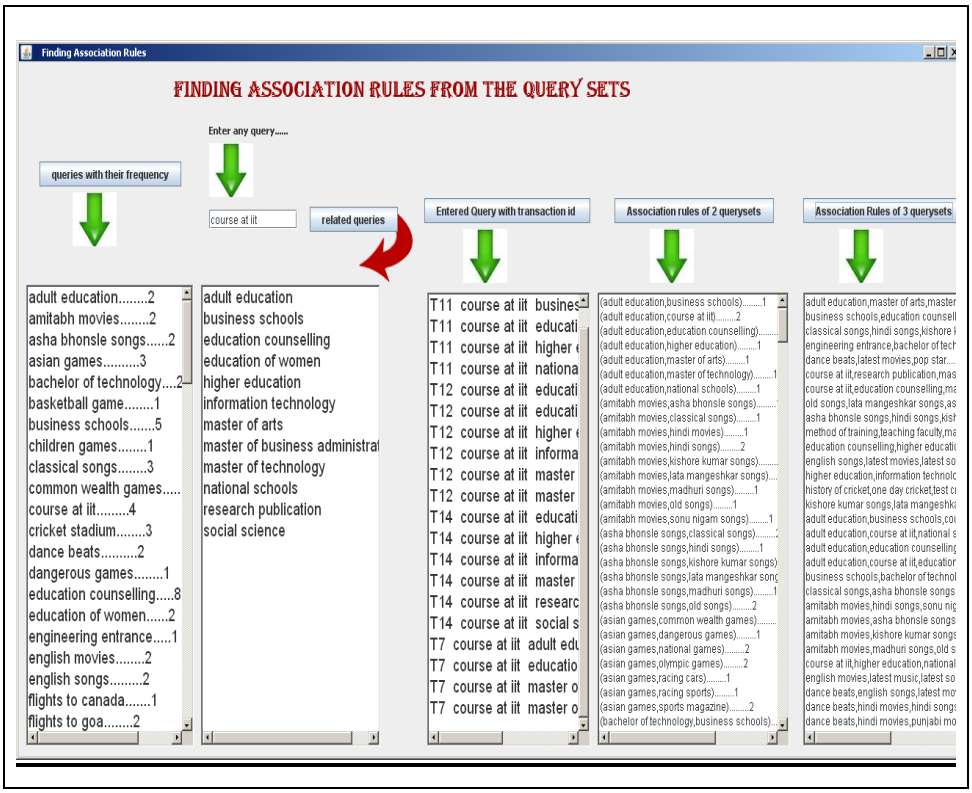


Fig. A8.2 Snapshot showing the frequent itemsets and association rules

The predicted questions are shown in Fig. A8.3.

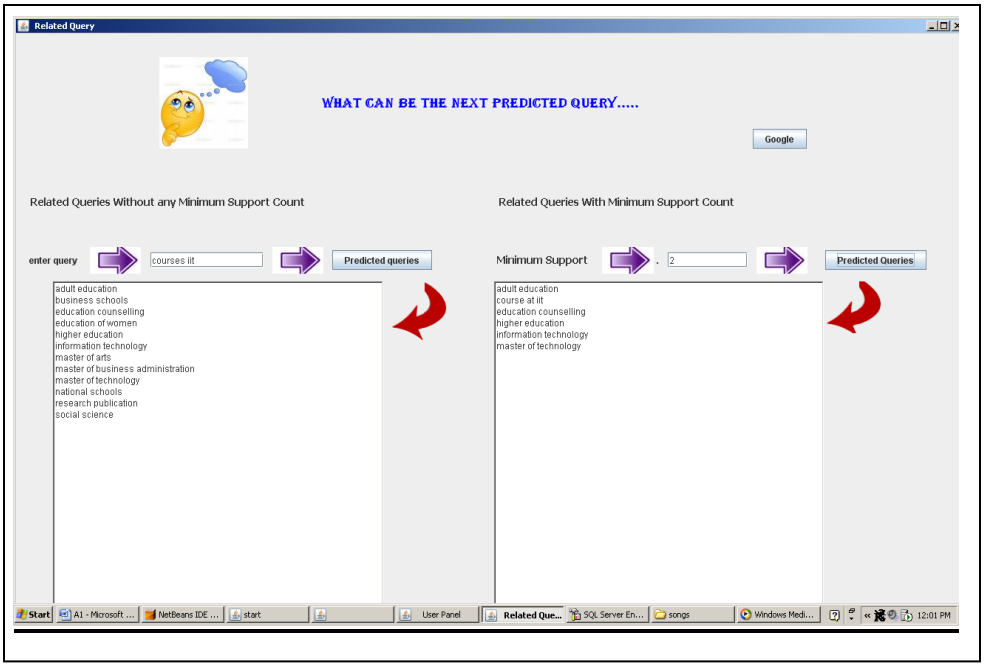


Fig. A8.3 Snapshot showing the next predicted questions

APPENDIX-9

The following figures show some snapshots of the implementation of the PQAS:

Example 1. Snapshots for some sample questions belonging to "who" question class

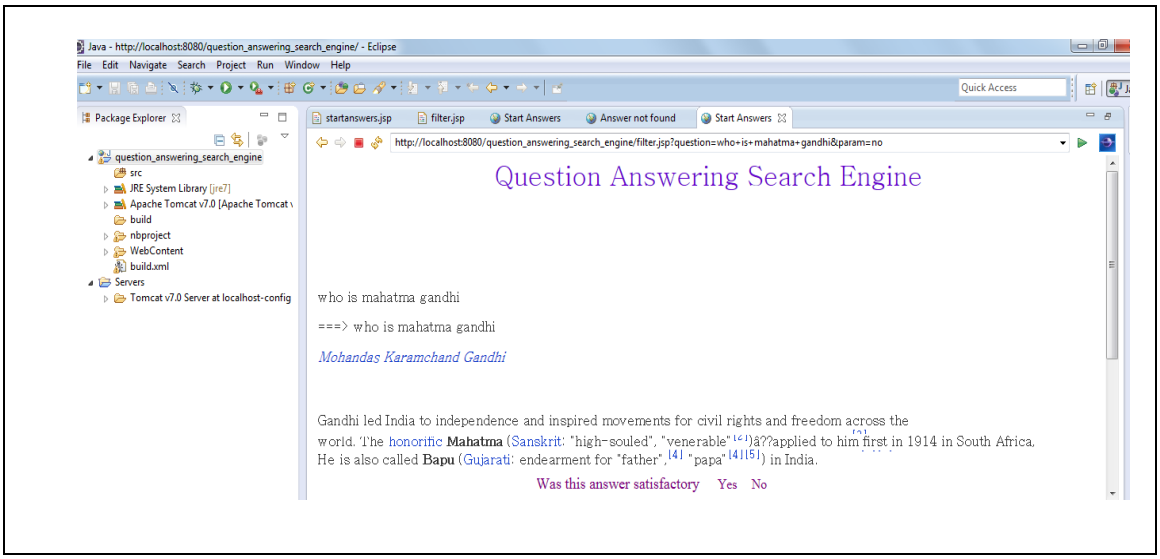


Fig. A9.1 Sample Question 1 starting with "who"

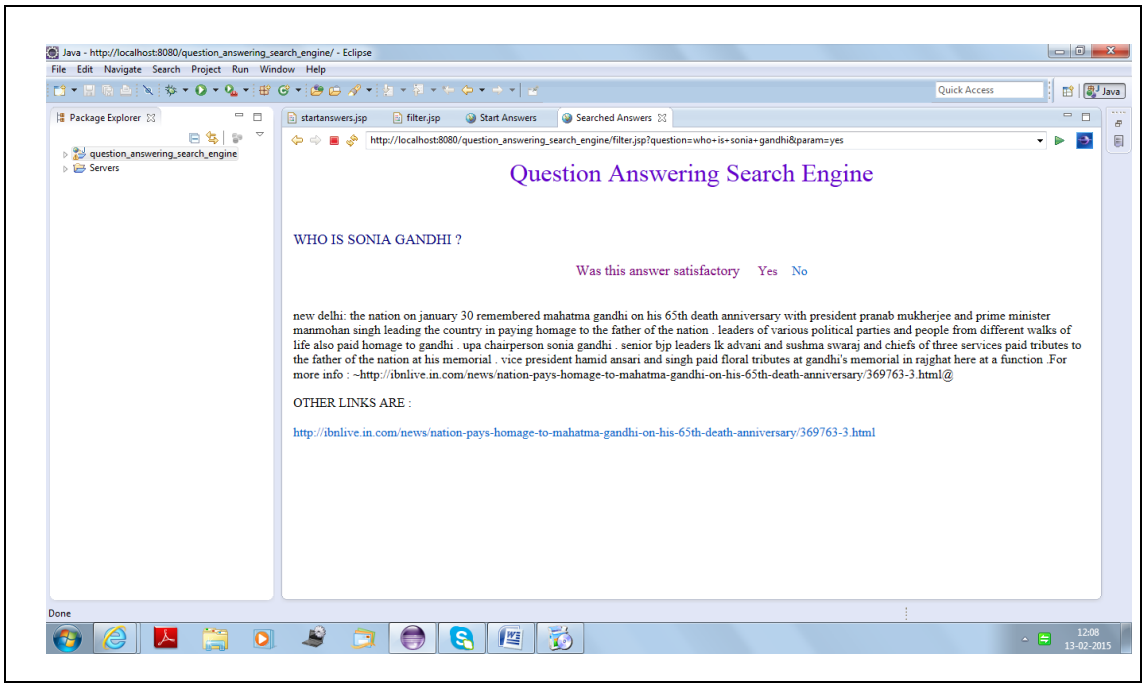


Fig.A9.2 Sample Question 2 starting with "who"

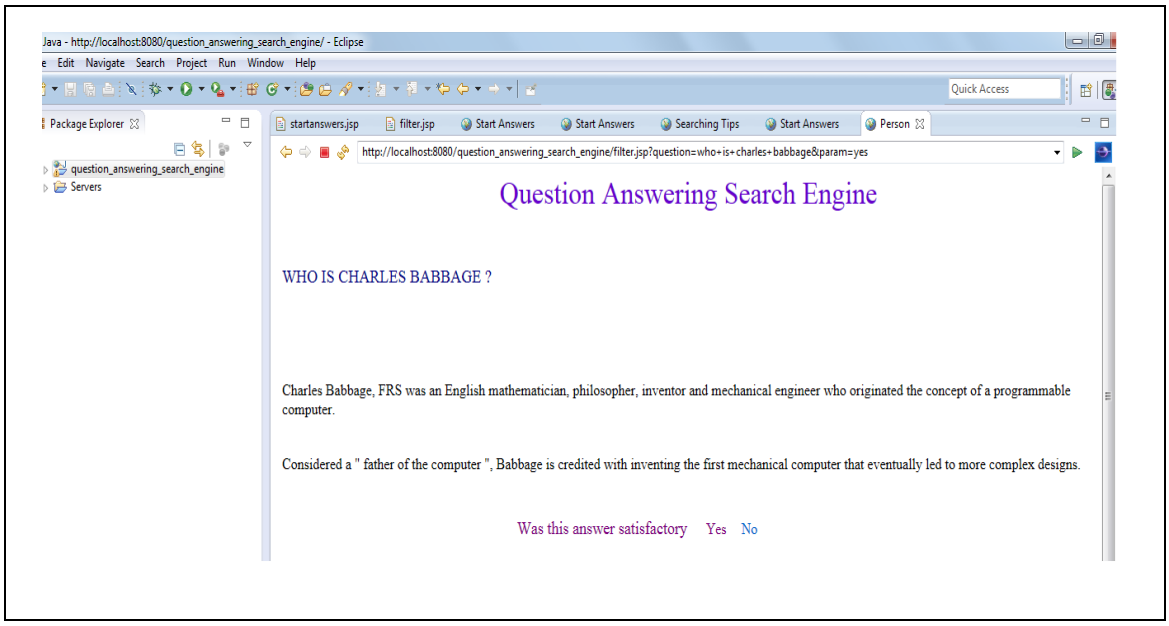


Fig. A9.3 Sample Question 3 starting with "who"

Example 2. Snapshots for some sample questions belonging to "where" question class

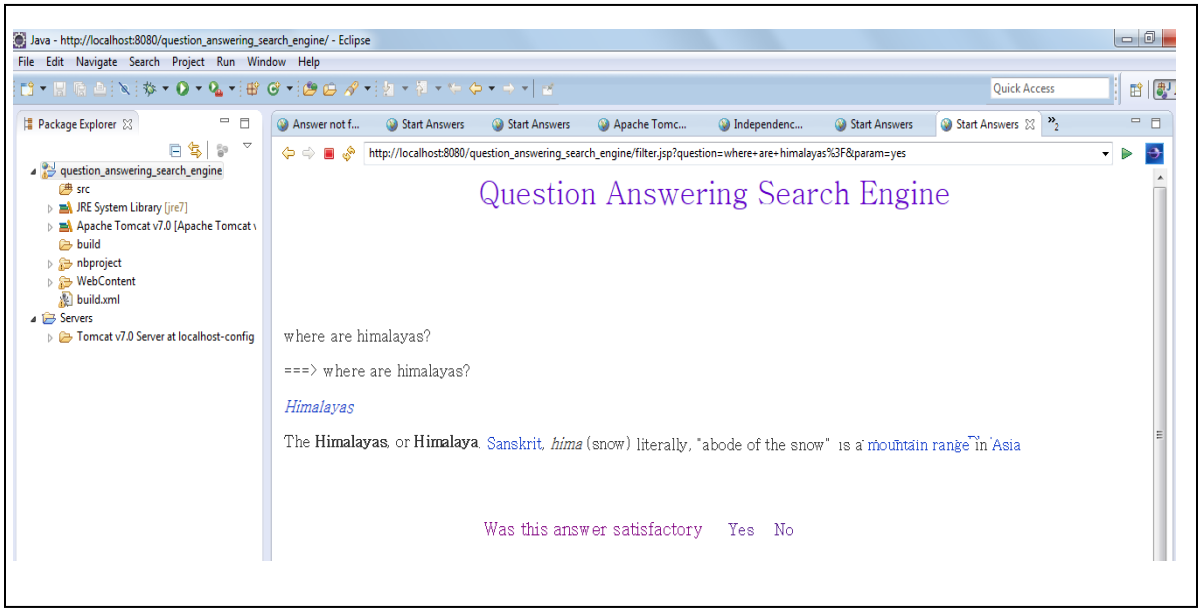


Fig.A9.4 Sample Question 1 starting with "where"

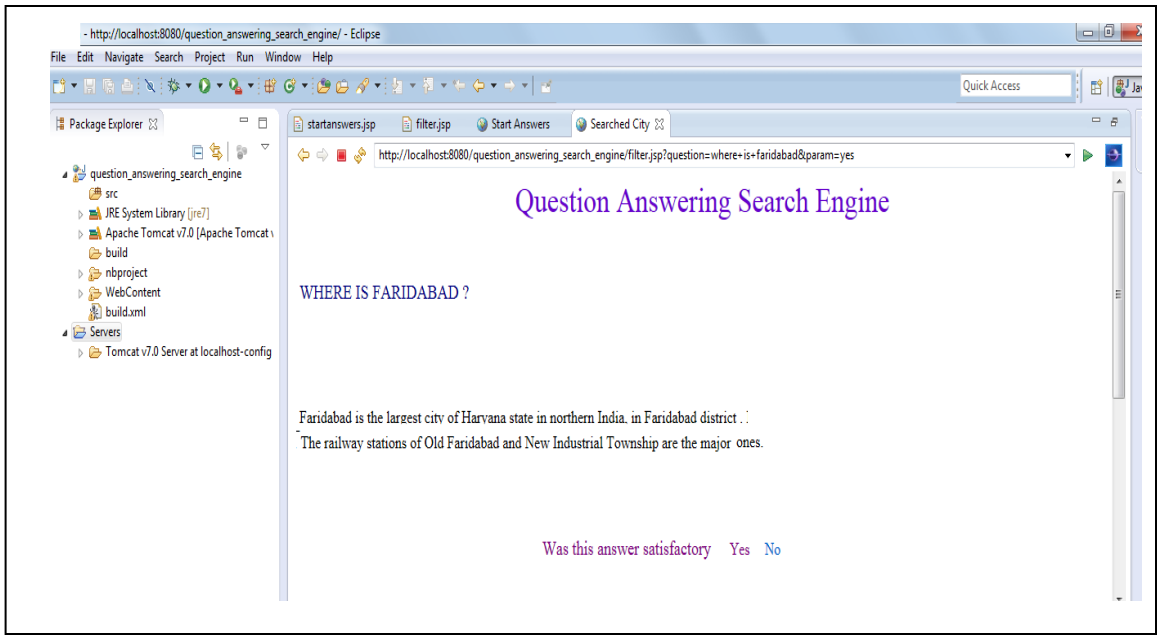


Fig.A9.5 Sample Question 2 starting with "where"

Example 3. Snapshots for some sample questions belonging to "when" question class

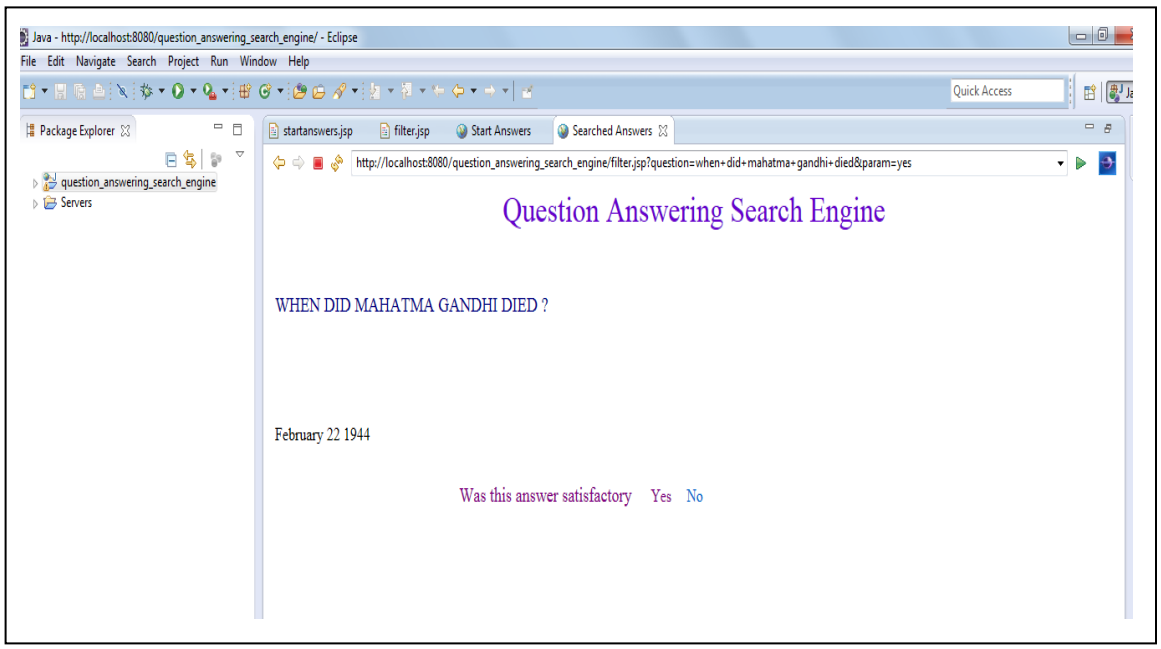


Fig. A9.6 Sample Question 1 starting with "when"

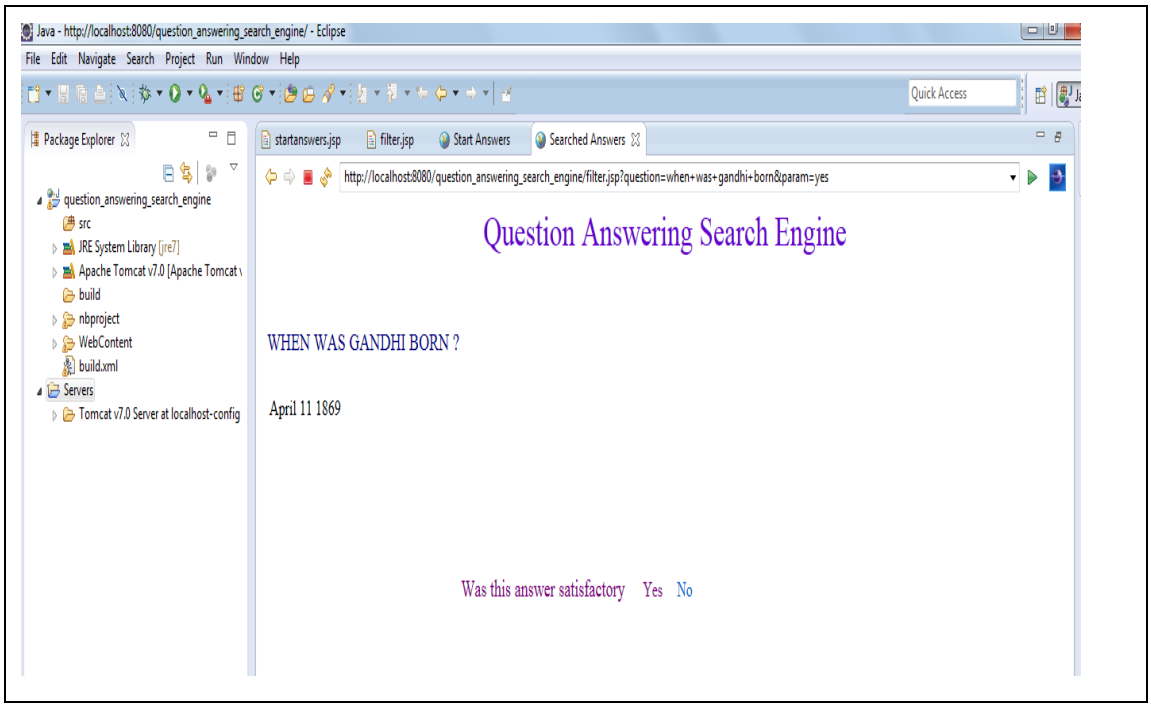


Fig. A9.7 Sample Question 2 starting with "when"

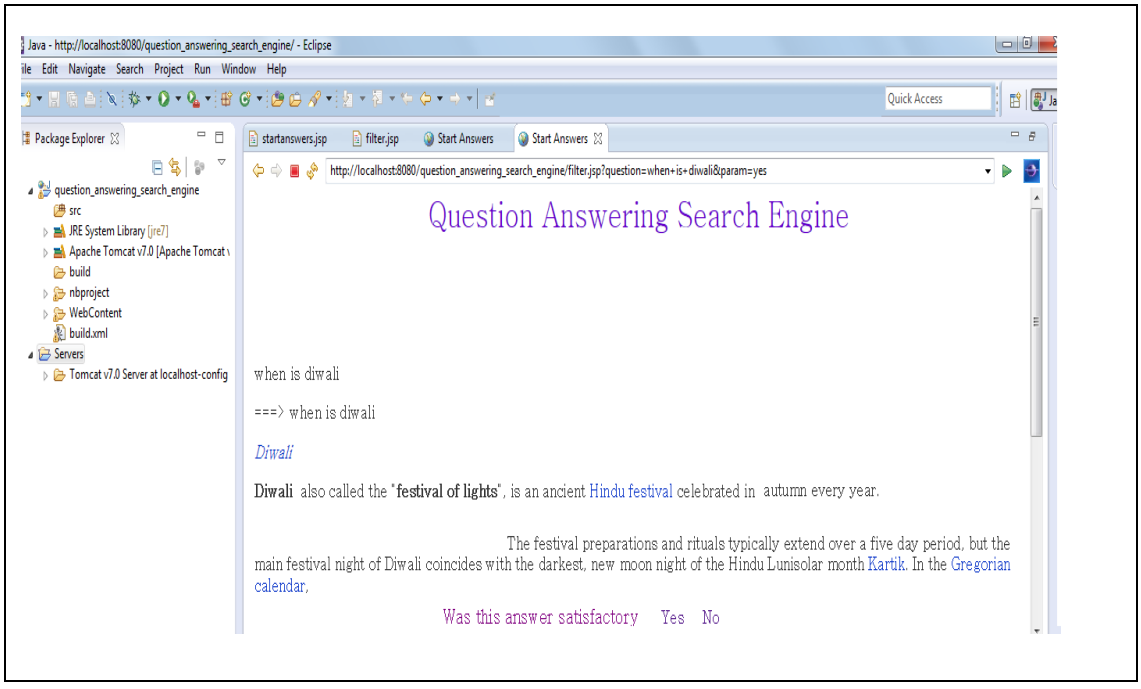


Fig. A9.8 Sample Question 3 starting with "when"

Example4. Snapshots for some sample questions belonging to "what" question class

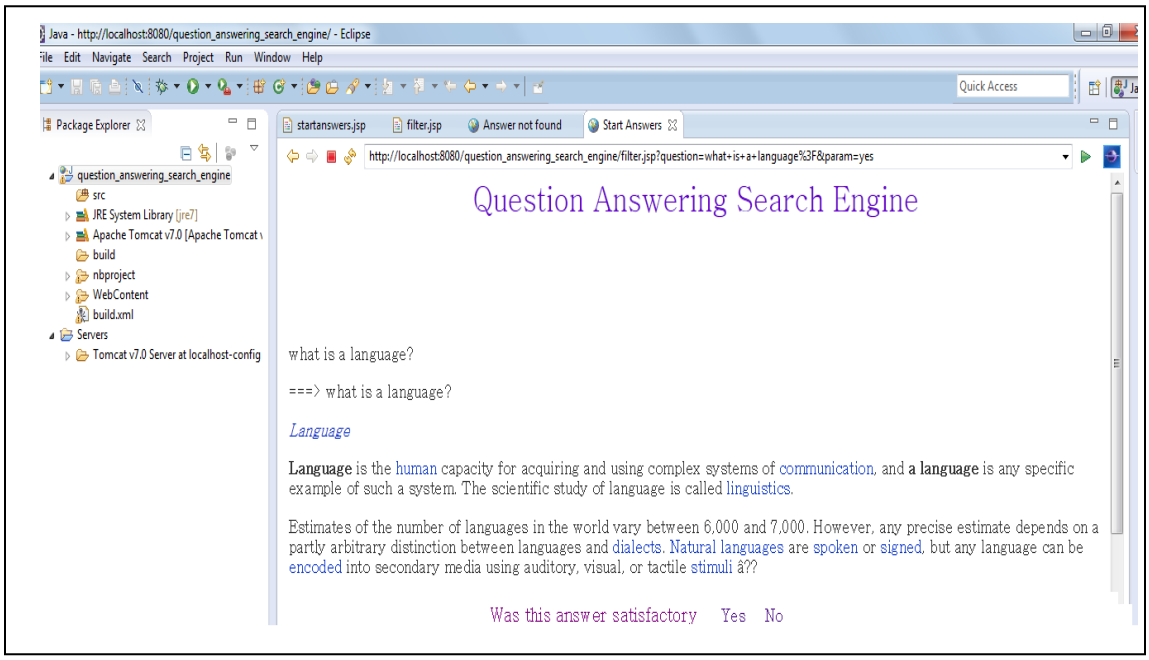


Fig. A9.9 Sample Question 1 starting with "what"

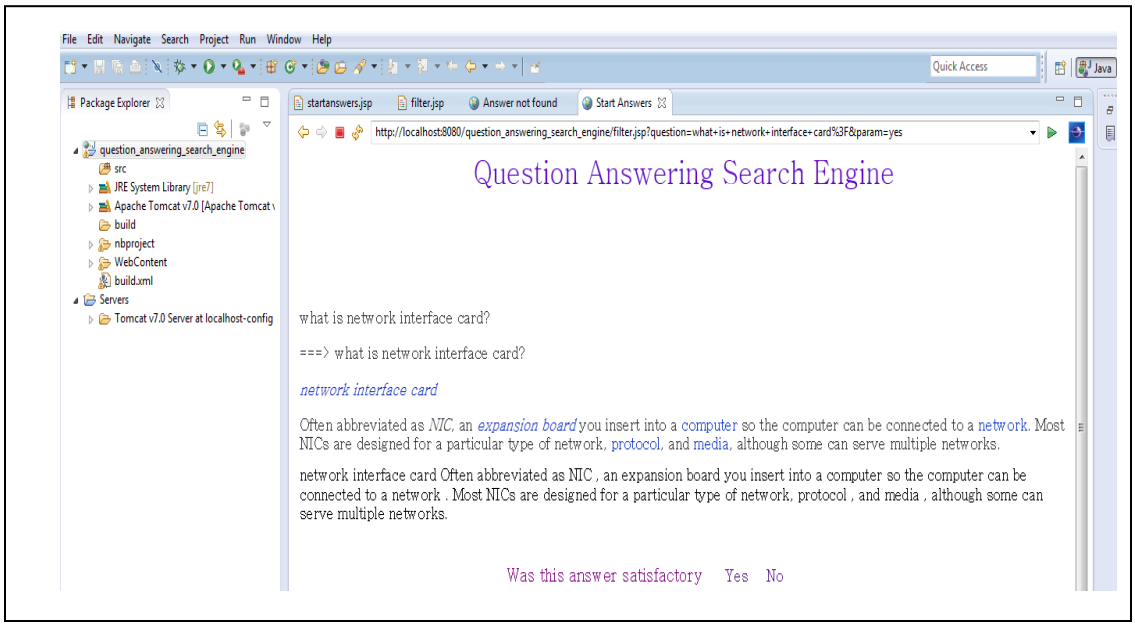


Fig. A9.10 Sample Question 2 starting with "what"

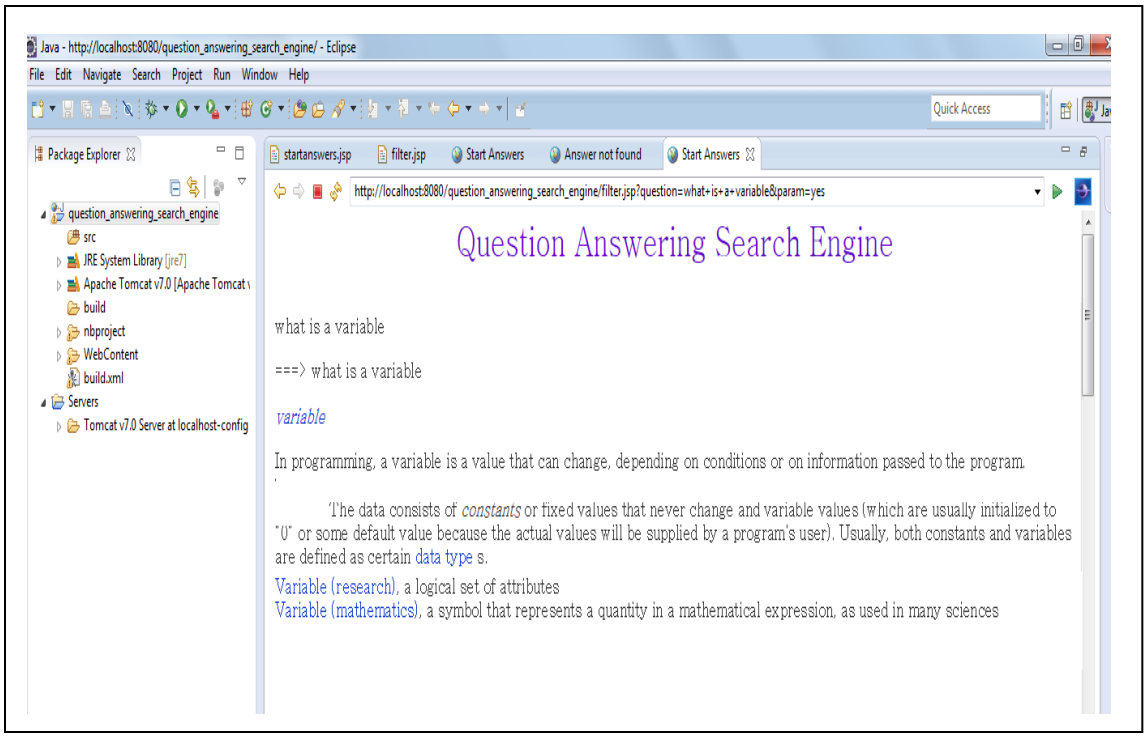


Fig. A9.11 Sample Question 3 starting with "what"

Example5. Snapshots for some sample questions belonging to "which" question class

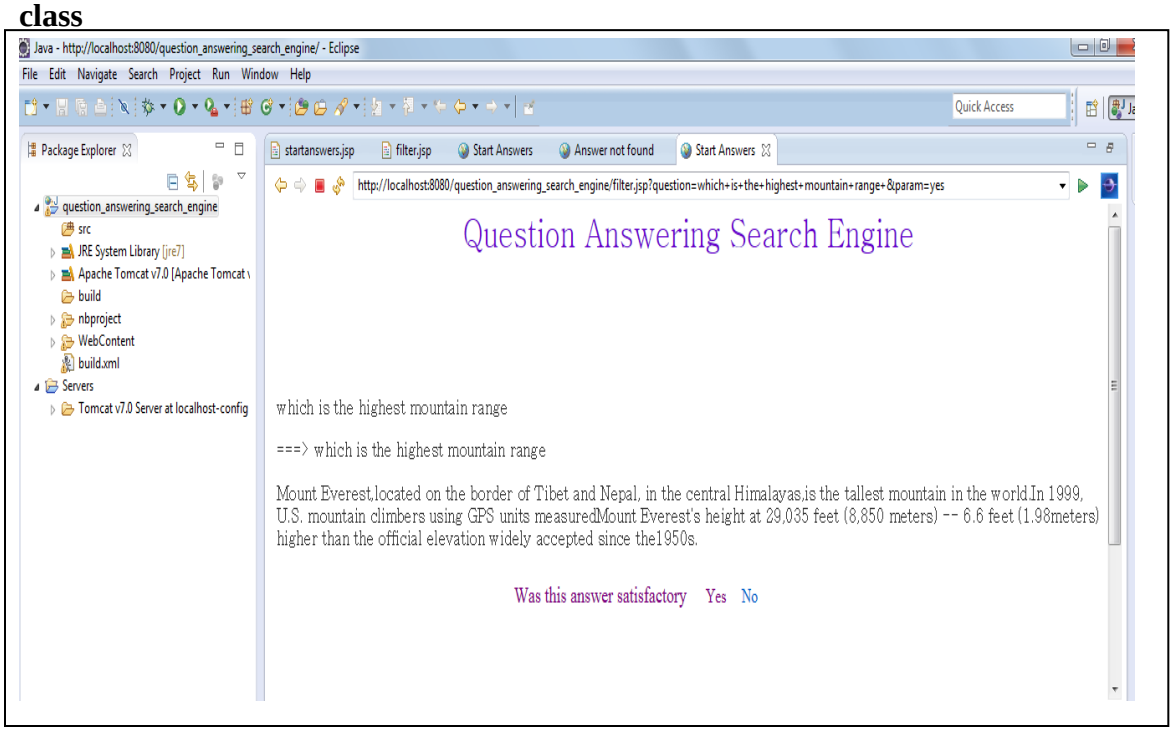


Fig. A9.12 Sample Question 1 starting with "which"

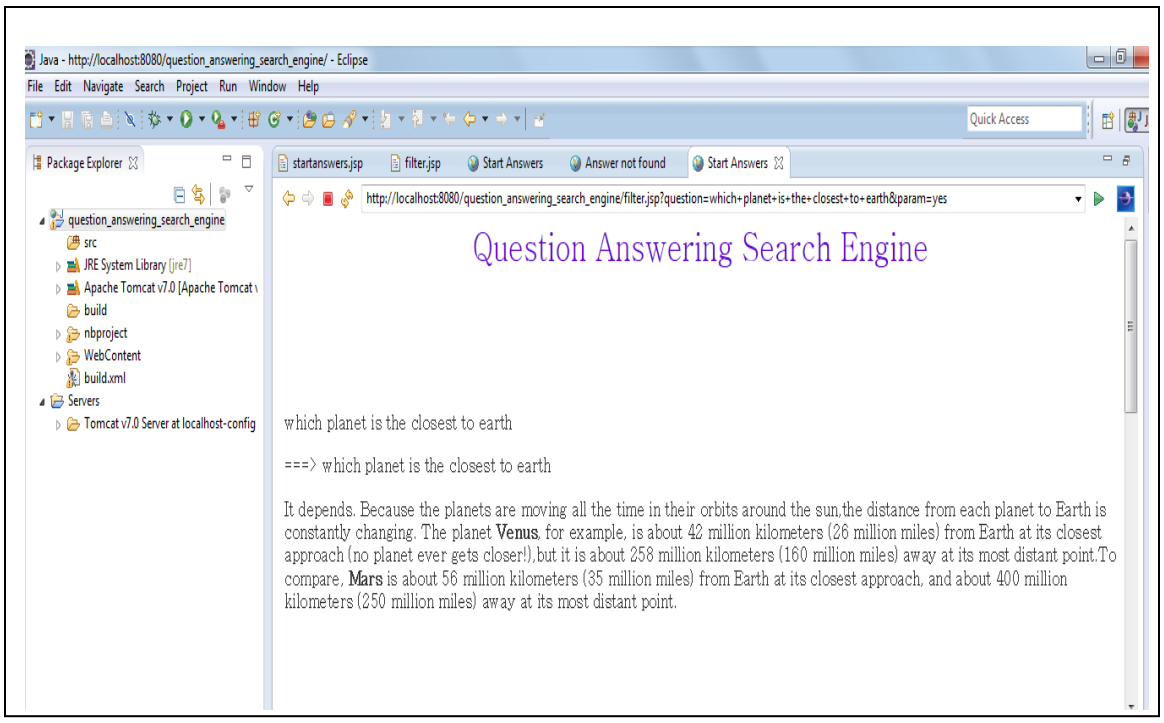


Fig. A9.13 Sample Question 2 starting with "which"

Example6. Snapshots for some sample questions belonging to "why" question class

Fig. A9.14 Sample Question 1 starting with "why"

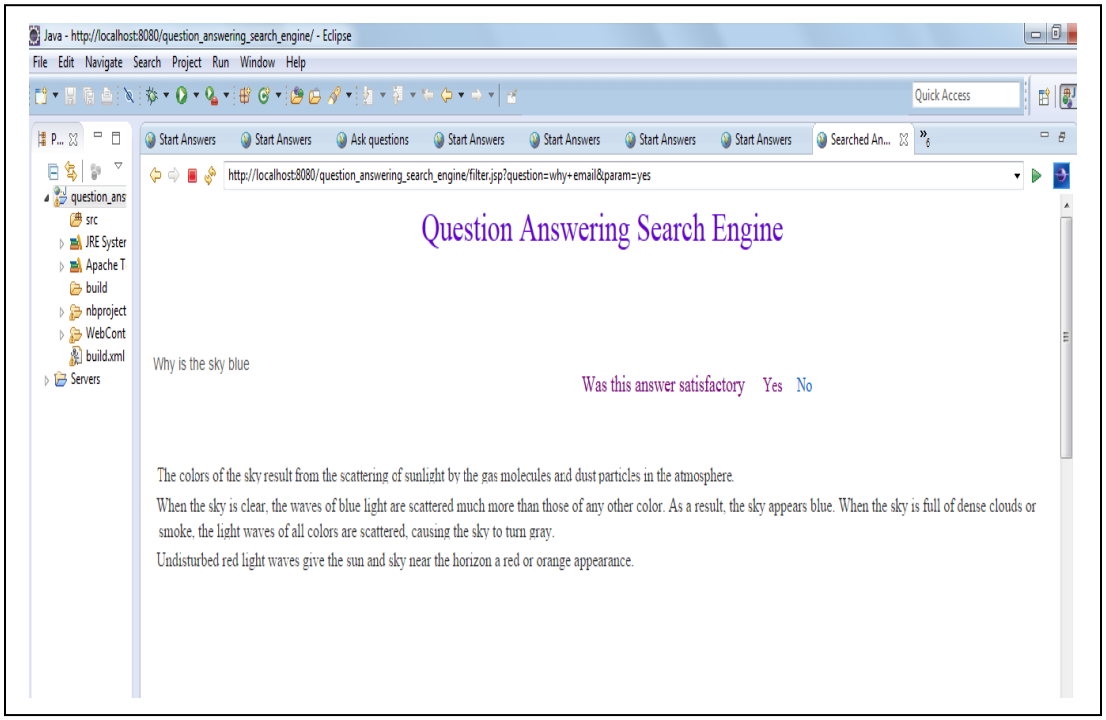


Fig. A9.15 Sample Question 2 starting with "why"

Example 7. Snapshots for some sample questions belonging to "how" question class

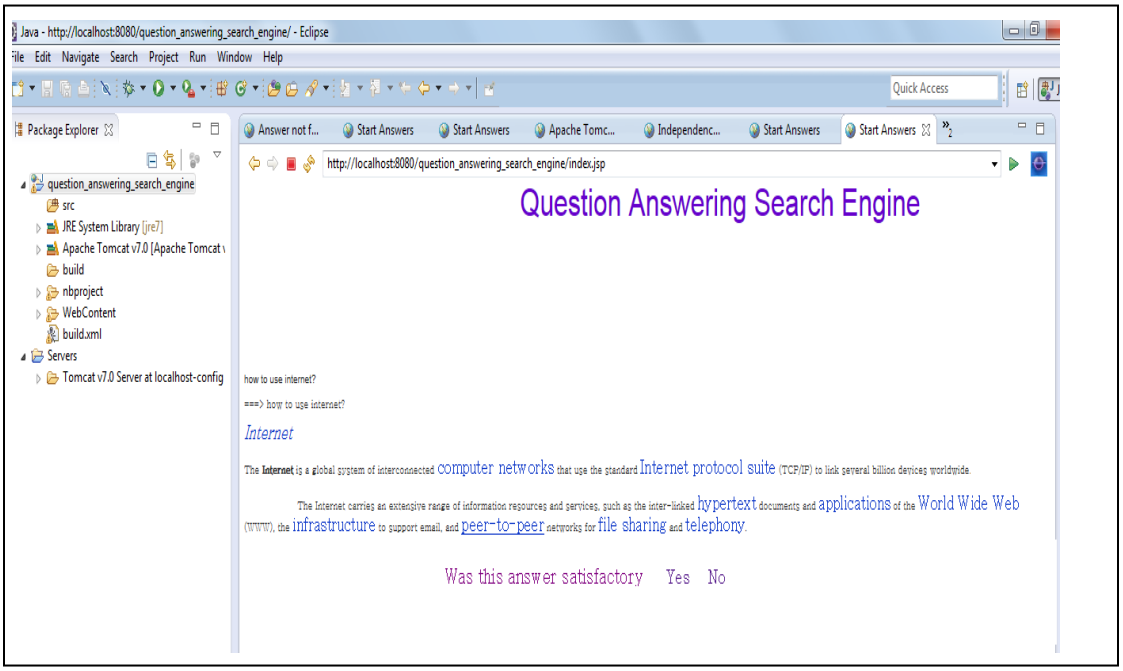


Fig. A9.16 Sample Question 1 starting with "how"

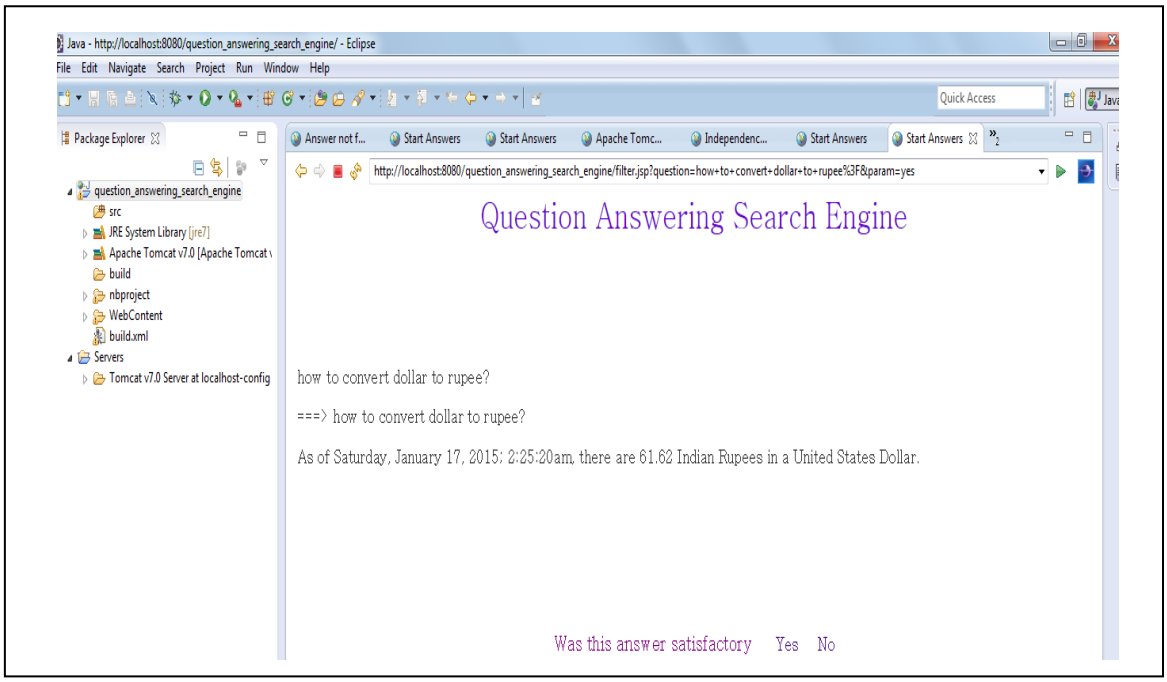


Fig. A9.17 Sample Question 2 starting with "how"

Table 9.15 Comparison of PQAS with Existing Question Answering systems

Characteristics	PQAS	Ask.com	Answers.com	START
History	The proposed system PQAS comprises of six functional components and it is able to respond to the user's questions posed in a natural language with accurate answers. It is able to answer the questions that start with who, where, what, when, which, how and why.	Ask.com [100] (originally known as Ask Jeeves) was founded in 1996 by Garrett Gruener and David Warthen in Berkeley, Calif. It allows online searchers to get answers to questions posed in everyday, natural language.	Answers.com [101] is an Internet-based knowledge exchange, which includes WikiAnswers, ReferenceAnswers, VideoAnswers, and five international language Q&A communities. The Answers.com domain name was purchased by entrepreneurs Bill Gross and Henrik Jones at Idealab in 1996.	START [102], the world's first Web-based Natural language question answering system, has been on-line and continuously operating since December, 1993. It has been developed by Boris Katz and his associates of the InfoLab Group at the MIT Computer Science and Artificial Intelligence Laboratory.
Crawling	The system uses its own Blog crawler to download the blog pages.	Doesn't use crawler	Uses a general crawler	Uses a general crawler
Quality data	The system maintains the quality data by ranking the blog posts and ignoring those that have rank lower than the decided threshold value.	*	*	*

Answer generation	The proposed downloads the blog pages and then extracts the relevant content from these pages. The pages are then indexed for searching corresponding to user's questions. The system constructs and uses Question classified index for indexing and responding to user's questions. The system also enriches its repository with the data obtained from other alternate sources containing user satisfactory answers.	Uses Google to answer to the user's question with a list of web pages as result. It also provides an option that puts the question on community and asks its members to respond. The answers given by the members are then sent to the user who asked the question by email.	Uses wikianswers, ReferenceAnswers, VideoAnswers, and five international language Q&A communities to respond to user's questions. Also, puts the question asked under a suitable category and ask the members of the corresponding community to respond.	Uses some web sources like Wikipedia, some books, dictionaries, projects and some web sources to answer the user's question. START parses incoming questions, matches the queries created from the parse trees against its knowledge base and presents the appropriate information segments to the user.
Answer rating	Asks, if the user is satisfied?-yes or no	No	Asks, if the answer is useful?-yes, no or somewhat	No
Source of answers shown	No	Yes	Yes	Yes
User oriented results	If the user says that the answer to his question is not satisfactory, then the system looks for an alternate data source to answer to user's question.	No	Sometimes	Sometimes

Answer accuracy	High	*	*	*
Extensible	Yes	Can't say	Can't say	Can't say
Scalable	Yes	Can't say	Can't say	Can't say
Limitatons	1. Shows somewhat low performance in case of the questions that start with why and how.	1. User has to search for the answer in the result page given by the Search engine, thus aim the objective of Question answering not fulfilled. 2.Also, the user has to wait for the answer until it is provided by some member of the QA community of Ask.com	1. Question has to be categorized into an appropriate category and then is then forwarded to the members of the QA community for answering. 2. The user waits until it is responded by someone else.	1. The grammatical structure of the question asked needs to be accurate otherwise the system fails to answer the question. 2. Also, the question that mean the same as an already answered question but has few more terms may not be responded by the system.

* *not claimed.*

REFERENCES

[1] Ricardo Baeza-Yates and Berthier Ribeirmobileo-Neto, "Modern Information Retrieval", ACM Press/ Addison-Wesley, 1999.

[2] Arvind Arasu, Junghoo Cho, Hector Garcia-Molina, Andreas Paepcke, and Sriram Raghavan, "Searching the Web", ACM Transactions on Internet Technology (TOIT), 1(1):2–43, August 2001.

[3] Terrence A. Brooks, "Web Search: How the Web has changed information retrieval", Information Research, April 2003.

[4] Sergei Brin and Lawrence Page, "The anatomy of a large-scale hypertextual Web search engine", Computer Networks and ISDN Systems, 30(1–7):107–117, April 1998.

[5] Ricardo Baeza-Yates, "Challenges in the interaction of information retrieval and natural language processing" In Proceedings of 5th international conference on Computational Linguistics and Intelligent Text Processing (CICLing), volume 2945 of Lecture Notes in Computer Science, pages 445–456. Springer, February 2004.

[6] Ricardo Baeza-Yates and Carlos Castillo, "Crawling the infinite Web: five levels are enough", In Proceedings of the third Workshop on Web Graphs (WAW), volume 3243 of Lecture Notes in Computer Science, pages 156–167, Rome, Italy, Springer, October 2004.

[7] Thanaa M. Ghanem and Walid G. Aref, "Databases deepen the Web", Computer, 37(1):116 –117, 2004.

[8] R. Kumar, P. Raghavan, S. Rajagopalan, D. Sivakumar, A. Tomkins, and E. Upfal, "Stochastic models for the web graph", In Proceedings of the 41st Annual Symposium on Foundations of Computer Science (FOCS), pages 57–65. IEEE CS Press, 2000.

[9] Steve Lawrence and C. Lee Giles, "Accessibility of information on the web. Intelligence", 11(1):32–39, 2000.

[10] Lipyeow Lim, Min Wang, Sriram Padmanabhan, Jeffrey Scott Vitter, and Ramesh Agarwal, "Characterizing Web document change" In Proceedings of the Second

International Conference on Advances in Web-Age Information Management, volume 2118 of Lecture Notes in Computer Science, pages 133–144, London, UK, Springer-Verlag, July 2001.

[11] B. E. Brewington and G. Cybenko. “How Dynamic is the Web?” In Proceedings of the International World-Wide Web Conference, Amsterdam, The Netherlands, 2000.

[12] J. Cho and H. Garcia-Molina. “The Evolution of the Web and Implications for an Incremental Crawler.” In Proceedings of the Twenty-Sixth VLDB Conference, pp. 200–209, Cairo, Egypt, 2000.

[13] J. Cho and H. Garcia-Molina. “Estimating Frequency of Change.” Technical report, DB Group, Stanford University, Nov 2001.

[14] J. Cho and A. Ntoulas. “Effective Change Detection Using Sampling.” In Proceedings of the International Conference on Very Large Databases (VLDB), 2002.

[15] Brightplanet’s searchable databases directory. http://www.completeplanet.com.

[16] Junghoo Cho, Hector Garcia-Molina, “Parallel Crawlers”, WWW2002, May 7–11, 2002, Honolulu, Hawaii, USA.

[17] S. Chakrabarti, K. Punera, and M. Subramanyam, “Accelerated focused crawling through online relevance feedback”, In Proc. of WWW, pages 148–159, 2002.

[18] S. Chakrabarti, M. van den Berg, and B. Dom, “Focused Crawling: A New Approach to Topic-Specific Web Resource Discovery” Computer Networks, 31(11-16):1623–1640, 1999.

[19] A.K. Sharma, J. P. Gupta, “Design of a Parallel Crawler based on Augmented Hypertext Documents (PARCAHYD)”, Ph.D. Thesis, IIIT & M, Gwalior, Aug. 2003.

[20] Christopher D. Manning, Prabhakar Raghavan and H. Schutze, http://freecomputerbooks.com/Introduction-to-Information-Retrieval .html.

[21] Rong Zhou and Eric A. Hansen, ”Breadth-First Heuristic Search”, 14th International Conference on Automated Planning and Scheduling (ICAPS-04), Whistler, British Columbia, Canada , June 3 - 7, 2004.

[22] J. Cho, H. Garcia-Molina, and L. Page, "Efficient crawling through URL ordering", In Proc. of the 7th International WWW Conf., 1998.

[23] http://rss.softwaregarden.com/aboutrss.html.

[24] A. Bergholz and B. Chidlovskii, "Crawling for Domain-Specific Hidden Web Resources", Proc. of the Fourth International Conference on Web Information Systems Engineering (WISE'03), IEEE, 2003.

[25] Dinesh Sharma, A.K. Sharma, Komal Kumar Bhatia, "Web crawlers: a review", Proc. of NCTC-2007.

[26] Dinesh Sharma, A.K. Sharma, Komal Kumar Bhatia, "Search engines: a comparative review", Proc. of NGCIS-2007.

[27] Profusion's search engine directory. http://www.profusion.com/nav.

[28] Z. Zheng, "AnswerBus question answering system", in Proceedings of the Human Language Technology Conference (HLT 2002), San Diego, CA, 2002.

[29] Michele Banko, Eric Brill, Susan Dumais and Jimmy Lin, "AskMSR: Question Answering Using the Worldwide Web", AAAI Spring Symposium on Mining Answers from Texts and knowledge Bases, California, March 2002.

[30] Eric Brill, Susan Dumais and Michele Banko, "An Analysis of the AskMSR Question-Answering System", In Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP 2002), 2002.

[31] Susan Dumais, Michele Banko, Eric Brill, Jimmy Lin and Andrew Ng, "Web Question Answering: Is More Always Better", SIGIR'02, August 2002, Finland, ACM, Copyright 2002.

[32] Lorand Dali, Delia RUSU, Blaz Fortuna, Dunja Mladenic and Marko Grobelnik, "Question Answering Based on Semantic Graphs", WWW 2009, Spain, April 2009.

[33] Bookbaby presents, http://www.gc.astd.org/Resources/Documents/Forums/Writers/Blogging101.pdf.

[34] Daniel W. Drezner and Henry Farrell, "The power and politics of blogs", American Political Science Association. August 2006.

[35] Rodrygo L. T. Santos, Richard McCreadie and Iran Soboroff, "Information Retrieval on the Blogosphere", Foundations and Trends in Information Retrieval Vol. 6, No. 1 (2012) 1–125, 2012.

[36] Anne Helmond and Geert Lovink, "Blogging for Engines", Ph.D. Thesis, University of Amsterdam, January 2008.

[37] Educase Learning Initiative, https://net.educause.edu/ir/library/pdf/ELI7006.pdf, August 2005.

[38] National centre for technology in education, http://www.pdsttechnologyineducation.ie/en/Technology/Advice-Sheets/Blogs.pdf, November 2008.

[39] Prachi Prashar Panday, "Why Blogs", https://net.educause.edu/ir/library/pdf/CSD4902.pdf, Copyright 2007.

[40] R. Kumar, J. Novak, P. Raghavan, and A. Tomkins, "On the bursty evolution of blogspace", World Wide Web, vol. 8, no. 2, pp. 159–178, 2005.

[41] G. Mishne and M. de Rijke, "A study of blog search", in Advances in Information Retrieval, 28th European Conference on IR Research, ECIR 2006, London, UK, April 10-12, 2006, Proceedings, ser. Lecture Notes in Computer Science, vol. 3936, 2006, pp. 289–301.

[42] Li Wei-jiang, Ru Hua-suo, Hong Kun, Luo Jia, "A New Algorithm of Blog-oriented Crawler", International Forum on Computer Science-Technology and Applications, 2009.

[43] S. Shanmugapriyaa, K.S. Kuppusamy, G. Aghila, "BLOSEN: Blog Search Engine Based on Post Concept Clustering", International Journal of Ambient Systems and Applications (IJASA), Vol. 1, No.3, 2013.

[44] Sachio Hirokawa, Chengjiu Yin, Tetsuya Nakatoh, "Component-Based Search Engine for Blogs", IEEE International Conference on Fuzzy Systems, June 2011.

[45] Biplab Ch. Das, "A Survey on Question Answering System", Thesis, department of Computer science & Engineering, Indian Institute of Technology, Bombay.

[46] Andrew Lampert, "A Quick Introduction to Question Answering", 2004.

[47] https://en.wikipedia.org/wiki/Question_answering.

[48] K. C.-C. Chang, B. He, C. Li, M. Patel, and Z. Zhang, “Structured Databases on the Web: Observations and Implications”, SIGMOD Record, 33(3):61–70, 2004.

[49] Fonseca Bruno M. , Golgher Paulo B., de Maura Edleno S., Ziviani Nivio, “Using association rules to discover search engine related queries”, Proceedings of the First Latin American Web Congress (LA-WEB), IEEE, 2003.

[50] Gupta Deepti, Puniya Antima, Bhatia kumar komal,”Prediction of the Query of the Search Engine using Back propogation Algorithm”, IJCSE, 2011.

[51] Lin, K.H, “Predicting Next Search Actions with Search Engine Query Logs”, Web Intelligence and Intelligent Agent Technology (WI-IAT), IEEE/WIC/ACM International Conference on (Volume: 1), 2011.

[52] Georges Dupret and Marcelo Mendoza, “Recommending Better Queries Based on Click-Through Data”, LNCS, Springer, 2005.

[53] X. Song, Y. Chi, K. Hino, and B. L. Tseng,” Summarization System by Identifying Influential Blogs”, ICWSM, 2007.

[54] S. Mithun, and L. Kosseim, “Discourse Structures to Reduce Discourse Incoherence in Blog Summarization”, Proceedings of Recent Advances in Natural Language Processing, September 2011.

[55] B. Sharifi, M.A. Hutton, and J. Kalita,” Automatic Summarization of Twitter Topics”, 2010.

[56] S. Sun, “A New Approach to Blog Post Summarization Using Fast Features”, Proceedings of the 2008 Fifth International Conference on Fuzzy Systems and Knowledge Discovery, 2008.

[57] C.M. Hu, A. Sun, and E. Lim, “Comments-Oriented Blog Summarization by Sentence Extraction", CIKM, Hurst, M.; Maykov, A., ”Social Streams Blog Crawler”, ICDE '09. IEEE 25th International Conference on data Engineering, 2009.

[58] Philipp Berger, Patrick Hennig, Justus Bross, Christoph Meinel, “Mapping the Blogosphere-Towards a Universal and Scalable Blog-Crawler”, IEEE International Conference on Privacy, Security, Risk, and Trust, and IEEE International Conference on Social Computing, 2011.

[59] Mehdi Naghavi1 and Mohsen Sharifi1, “A proposed architecture for continuous Web monitoring through online crawling of blogs”, International Journal of UbiComp (IJU), Vol.3, No.1, January 2012.

[60] Richard E Fardig, Kaye D, “Content delivery in Blogosphere”, 2004.

[61] E. Adar, L. Zhang, L. Adamic, and R. Lukose, “Implicit Structure and the Dynamics of Blogspace,” In Proceedings of the Workshop on the Weblogging and Ecosystem at the 13th International World Wide Web Conference, 2004.

[62] http://www.thesitewizard.com/faqs/howtoreadsitefeeds.html.

[63] Apostolos Kritikopoulos, Martha Sideri, and Iraklis Varlamis, “Blogrank:ranking weblogs based on connectivity and similarity features”, ACM, 2006.

[64] Fujimura, K., Inoue, T., Sugisaki, M., (2005). “The EigenRumor Algorithm for Ranking Weblogs”, 2nd Annual Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, WWW 2005.

[65] Justus Bross,Keven Richly,Matthias Cohnen,Christoph Meinel, “Identifying the Top dogs of Blogosphere”, Springer-Verlag, 2011.

[66] Kritikopoulos, Apostolos, Martha Sideri, and Iraklis Varlamis. "BLOGRANK: Ranking on the blogosphere." Proceedings of the International Conference on Weblogs and Social Media (ICWSM 2007), Boulder, Colorado, USA, 2007.

[67] Shen, Jie, et al. "A content-based algorithm for blog ranking." Internet Computing in

Science and Engineering, 2008. ICICSE'08. International Conference on. IEEE, 2008.

[68] Tayebi, Mohammad A., S. Mehdi Hashemi, and Ali Mohades. "B2Rank: An algorithm for ranking blogs based on behavioral features." Web Intelligence, IEEE/WIC/ACM International Conference on IEEE, 2007.

[69] http://www.babelfish.com/

[70] http://blogflux.com/

[71] http://urlm.co/www.topblogarea.com

[72] Mani, I., MayBury, M.T., “Advances in Automatic Text Summarization”, The MIT

Press, 1999.

[73] David Pinto, Michael Branstein, Ryan Coleman, W. Bruce Croft, Matthew King, Wei Li and Xing Wei, "QuASM: A System for Question Answering Using Semi-Structured Data", 2nd ACM/IEEE-CS joint conference on Digital Libraries, Amherst, MA, 2002.

[74] Dell Zhang, Wee Sun Lee, "Question Classification using Support Vector Machines", proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval, 2003.

[75] Xin Li and Dan Rot, "Learning Question Classifiers: The Role of Semantic Information", Proceedings of the 19th international conference on Computational linguistics, 2004.

[76] Dunwei WEN , Shen JIANG , Yangjian HE, "A Question Answering System Based on VerbNet Frames", International Conference on Natural Language Processing and Knowledge Engineering, Athabasca University Athabasca, AB, 2008.

[77] Zhiheng Huang, Marcus Thint, "Question Classification using Headwords and their Hypernyms, in Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing, pages 927-936, 2008.

[78] Joseph Chang, Tzu-Hsi Yen, Richard Tzong-Han Tsai, "Minimally Supervised Question Classification and Answering based on Word Net and Wikipedia", 21st Conference on Computational Linguistics and Speech Processing (ROCLING21), 2009.

[79] Fan Bu, Xingwei Zhu, Yu Hao, Xiaoyan Zhu, et.al., "Function Based Question Classification for general QA", proceedings of the 2010 Conference on Empirical Methods on Natural Language Processing (EMNLP 2010),pp.1119-1128, MIT, Boston U.S., 2010.

[80] Hakan Sundbald, "Question Classification in Question Answering Systems", Thesis No. 1320, Department of Computer and Information Science Linköpings University, Sweden 2007.

[81] Ashutosh Dixit and A.K Sharma, "Self Adjusting Refresh Time Based Architecture for Incremental Web Crawler", International Journal of Computer Science and Network Security (IJCSNS), Vol 8, No12, Dec 2008.

[82] www.cs.uic.edu/~liub/teach/cs583-fall-05/CS583-association-rules.ppt.

[83] www.cs.bilkent.edu.tr/~guvenir/courses/CS558/Association%20Rules.ppt.

[84] https://www.creighton.edu/fileadmin/user/HSL/docs/ref/Searching_-Recall_Precision.pdf

[85] http://www.seas.gwu.edu/~bell/csci243/lectures/performance.pdf.

[86] webhome.cs.uvic.ca/~thomo/seng474/precision-recall.ppt.

[87] web.stanford.edu/class/cs276/handouts/lecture8-evaluation.ppt.

[88] http://technorati.com/

[89] Rosy Madaan, A.K. Sharma, Ashutosh Dixit, "A Novel Architecture for Blog Crawler, 2nd IEEE International Conference on Parallel, Distributed and Grid Computing, ISBN: 978-1-4673-2922-4, Jaypee Institute of Information Technology & University of Florida, Noida, December 2012.

[90] http://vitobotta.com/using-google-define-search-feature-terminal/

[91] dictionary.reference.com/The world's most popular dictionary

[92] https://wordnet.princeton.edu/

[93] Rosy Madaan, A.K. Sharma, Ashutosh Dixit, "Presence Factor-Oriented Blog Summarization", International Journal of Advances in Computing & Information Technology (IJACIT) ISSN: 2277-9140 Volume-2, Issue-2, May 2013.

[94] Rosy Madaan, A.K. Sharma, Ashutosh Dixit, "A novel architecture for relevant content extraction from Blog pages", International Journal of Scientific and Engineering research (IJSER), ISSN: 2229-5518, Volume-4, Issue-5, May 2013.

[95] http://darylkinsman.ca/tools/wordfreq.shtml

[96] Renu Mudgal, Rosy Madaan, A.K. Sharma, Ashutosh Dixit, "A Novel architecture for question classification based indexing scheme for efficient question answering", International Journal of Computer Engineering & Applications (IJCEA), ISSN: 2321-3469, Volume-2, Issue-2, June 2013.

[97] Deepti Kapri, Rosy Madaan, A.K. Sharma, Ashutosh Dixit, "A Novel architecture for relevant blog page identification", International Journal of Computer Engineering & Applications (IJCEA)", ISSN: 2321-346, Volume-2, Issue-2, June 2013.

[98] Rosy Madaan, A.K. Sharma, Ashutosh Dixit, " A Data Mining approach to predict user's next question in a QA system", 9th INDIACom 2nd International Conference on Computing for Sustainable Global, BVICAM, IEEE Xplore, March, 2015.

[99] Rosy Madaan, A.K. Sharma, Ashutosh Dixit, "Design of a novel search engine for prospective question answering", International Journal of Information retrieval research (IJIRR)(free journal), ISSN: 2155-6377, Volume 4 Issue 2, ACM Digital Library, 2014.

[100] N. V. Pardakhe, Prof. R. R. Keole, "Analysis of Various Web Page Ranking Algorithms in Web Structure Mining", International Journal of Advanced Research in Computer and Communication EngineeringVol.2, Issue 12, December 2013.

[101] Rosy Madaan, A.K. Sharma, Ashutosh Dixit, Deepti Kapri, Renu Mudgal "A comparative study of Web based and IR based Question Answering systems, International Journal of Advances in Computing and Information Technology (IJACIT), ISSN: 2277-9140, Volume-1, Issue-2, April 2012.

[102] Rosy Madaan, A.K. Sharma, Ashutosh Dixit, "Prediction of Next Search Query using Association Rule Mining", International Conference on Data Acquisition, Transfer, Processing & Management, ISBN:978-81-924212-6-1, Lingaya's University, Nachauli, Faridabad, March-2014.

[103] Rosy Madaan, A.K. Sharma, Ashutosh Dixit, "A Comparative study of Question Answering Systems" at Recent Trends in Computer Science & Information Technology (RTCSIT-2013), ISBN: 978-93-82880-75-2, Echelon Institute of Technology, Faridabad, India, October 2013.

[104] Rosy Madaan, A.K. Sharma, Ashutosh Dixit, Deepti Kapri, Renu Mudgal, "Ranking blog pages", Recent Trends in Engineering & Engineering Education

(RTEEE-2012), ISBN:978-93-82062-93-6, Echelon Institute of Technology, Faridabad, India, October 2012.

[105] Rosy Madaan, A.K. Sharma, Ashutosh Dixit, Renu Mudgal, Deepti Kapri "Web Page Summarization Techniques: A Comparative Review", Recent Trends in Computer Science and Information Technology (RTCSIT-2012) ISBN: 978-93-82062-34-9, Echelon Institute of Technology, Faridabad, India, June 2012.

[106] http://www.ask.com/

[107] http://www.answers.com/

[108] http://start.csail.mit.edu/index.php.

[109] http://press.princeton.edu/chapters/s8216.pdf.

[110] Poonam Bhatia, Rosy Madaan, A.K. Sharma, Ashutosh Dixit, "A Comparison Study of Question Answering Systems", International Conference on Recent Trends in Computer and Information Technology Research, 2015.

[111] Rosy Madaan, A.K. Sharma, Ashutosh Dixit, Poonam Bhatia, "Indexing of semantic web for efficient question answering", CSI-2015.

[112] Shu-Jing Lin, Yi-Chung Chen, Don-Lin Yang, Jungpin Wu, "Discovering long maximal frequent pattern", Eighth International Conference on Advanced Computational Intelligence (ICACI), 2016.

[113] Syahida Hassan; Janet Toland; Mary Tate, "From Blogosphere to Social Commerce: A Laddering Analysis of Sellers' Motivation", 49th Hawaii International Conference on System Sciences (HICSS), 2016.

LIST OF PUBLICATIONS

List of Published Papers in International Journal

S.No	Title of the paper along with volume, Issue No, year of publication	Publisher	Impact Factor (as on journal's website)	Referred or Non-Referred	Whether you paid any money or not for publication	Remarks
1.	Design of a novel search engine for prospective question answering, Volume 4, Issue 2, April-June 2015	International Journal of Information retrieval research (IJIRR)		Referred	Free	Indexed on ACM Digital Library
2.	A Novel architecture for relevant blog page identification, Volume-2, Issue-2, June 2013	International Journal of Computer Engineering & Applications (IJCEA)	2.84	Referred	Paid	Indexed on DBLP, google scholar
3.	A novel architecture for relevant content extraction from Blog pages, Volume-4, Issue-5, May 2013	International Journal of Scientific and Engineering research (IJSER)	1.4	Referred	Paid	
4.	A Novel architecture for question classification based indexing scheme for efficient question answering, Volume-2, Issue-2, June 2013	International Journal of Computer Engineering & Applications (IJCEA)	2.84	Referred	Paid	Indexed on DBLP, google scholar
5.	Presence Factor-Oriented Blog Summarization, Volume-2, Issue-2, May 2013	International Journal of Advances in Computing & Information Technology (IJACIT)		Referred	Free	Indexed on DBLP, google scholar
6.	A comparative study of Web based and IR based Question Answering systems, Volume-1, Issue-2, April 2012	International Journal of Advances in Computing & Information Technology (IJACIT)		Referred	Free	

List of Published Papers in Conferences

S.No	Title of the paper along with volume, Issue No, year of publication	Publisher	Impact Factor	Referred or Non-Referred	Whether you paid any money or not for publication	Remarks
7.	A Novel Architecture for Blog Crawler	2nd IEEE International Conference on Parallel, Distributed and Grid Computing				Indexing on IEEE Xplore
8.	A Data Mining approach to predict user's next question in a QA system	9th INDIACom 2nd International Conference on Computing for Sustainable Global				Indexing on IEEE Xplore
9.	Prediction of Next Search Query using Association Rule Mining	International Conference on Data Acquisition, Transfer, Processing & Management, 2014				
10.	A Comparative study of Question Answering Systems	Recent Trends in Computer Science & Information Technology (RTCSIT-2013)				
11.	Ranking blog pages	Recent Trends in Engineering & Engineering Education (RTEEE-2012)				
12.	Web Page Summarization Techniques: A Comparative Review	Recent Trends in Computer Science & Information Technology (RTCSIT-2012)				
13.	A Comparison Study of Question Answering Systems	International Conference on Recent Trends in Computer and Information Technology Research, 2015				
14.	Indexing of semantic web for efficient question answering	CSI-2015				Springer

BRIEF PROFILE OF RESEARCH SCHOLAR

Rosy Madaan is a PhD scholar in Computer Engineering department of YMCA University of Science & Technology, Faridabad. She has completed her B.E. from M.D. University in Computer Science & Engineering in 2005. In 2010, she completed M.Tech., in Computer Engineering from YMCA University of Science & Technology, Faridabad. She has more than 10 years of teaching experience. She has worked in various engineering colleges and currently she is working as an Asst. Prof. in CSE department of G.D. Goenka University, Sohna-Gurgaon. Her areas of interest include information retrieval, search engines and crawlers. Her dissertation explores unresolved issues in the field of Question Answering and she has proposed the solutions to resolve them.