SEARCH BY QUESTION

Software Requirements Specification

Cankaya University
08/12/2017

Koray Danisma- 201511409,
Mehmet Şerefoğlu- 201611688,
Ali Dorukhan Karakaya- 201311027

Table of Contents

List of Figures

1.Introduction

1.1 Purpose

The purpose of this document is describing Search by Question: Find results of asked question on the internet and list them. It aims to access an information faster even almost all category; history, science, sport, education etc. Moreover, the SRS document explains algorithms of how the system find answers of questions.

1.2. Overview of the Project

Search by Question is basis of computers that will try to answer textual questions. The main point here is natural language processing, NLP for short, is a method for computers to analyze, understand, and make meaning from human language in a smart way. Through utilizing NLP, developers can promote and structure knowledge to perform works such translation, automatic summarization, named entity recognition, sentiment analysis, relationship extraction, speech recognition, topic segmentation..

Basis of Search by Question is also question answering. For systems implementing, there were two significant process of question answering: IR-based question answering and knowledgebased question answering.

Test about question answering was the computer that worked on answer the question that was “the Great Question Of Life The Universe and Everything” . Computer’s answer was 42, but the details of the this asked question were never unclosed. This took place in TV program that is called the Hitchhiker's Guide to the Galaxy. Another test was IBM’s Watson question-answering system, which is used for this project, won the game is called Jeopardy. The system answer the following question: What is your most inspired Novel the Author’s of "An account of the principalities of Wallachia and Moldovia”. The system answer both faster and logical to this question then human.

Search by Question more focuses on factoid questions. That is one of the significant difference between Question Answering and Search by Question. Factoid questions can be answered with basic facts expressed in any short text answer. For example, the following factoid questions can be answered with a short string expressing a date, personal name, or location:

(i) When was Ataturk born?
(ii) Who is the founder of Microsoft?
(iii) Where is Başar Soft based?

Analysis of one of the significant processes that is used for Search by Question is in the below, with focusing on its application to factoid questions.

2. Overall Description

2.1. Search by Question with IR-based Factoid Questions

The aim of IR-based search by question is finding short text parts on the Internet or some other collection of documents for answer a question. There are three states of Search by Question with IR-based Factoid Questions: question processing, passage retrieval, answer processing.

2.1.1. Question Processing

The purpose of the question processing state is extracting a number of components of information from the question. The answer type defines the type of entity the answer consists of (person, date, location, etc.). The query defines the keywords that is used for IR system for using in searching for documents. Some systems extract a focus, that is the string of words in the question that are replaced in any answer string found by the answer. Some systems also categorize the question kind that could be a definition question, a y/n question, a math question. For example, for the following question:

Which universities have computer engineering departments in Turkey?

The process of query produce results as the following:

Answer Type: university

Query: universities, computer engineering department, in Turkey

Focus: computer engineering department

2.1.2. Passage Retrieval

The query which was created in the question processing state is next used to query an information retrieval system. It is either a general IR engine on a exclusive set of indexed documents or a Web search engine. The result of this document retrieval phase is a set of documents.

Even though set of documents is usually ranked by relevance, the answer to the question is probably not top ranked document. Because documents are not an convenient unit to rank with respect to the purpose of a search by question system. A highly relevant and large document that does not distinctly answer a question is not an optimal candidate for further processing.

Hence, the next state is extracting a set of possible answer passages from the retrieved set of documents. The description of a passage is necessarily system dependent, but the characteristic units contain paragraphs, sentences, and sections.

After these processes, passage retrieval could be performed. First, passages that in passage retrieval the returned documents which don't include possible answers filtered out and then rank the rest by similarity of them to include an answer to the question. The first stage in this process is runing a named entity or answer kind classification on the retrieved passages. The answer kind which is determined from the question specify that the potential answer kinds that are expected to see in the answer. Therefore, documents which don't include any entities of the right kind can be filtered out.

The remaining passages are ranked, generally by supervised machine learning, trusting a small set of properties that could be smoothly extracted from a probable large number of answer passages, like: The rank of the document from that passage was extracted, the count of question keywords in the passage, the longest exact set of question keywords that takes place in the passage, the count of named entities of right kind in the passage.

So far search by question from the collective documents is analyzed. For search by question from the web also, instead of extracting passages from returned documents, doing passage extraction for us depends on Web search. This occurs by using snippets generated by Web search engine as the returned passages. For example, all results in the result list of google search are snippets.

2.1.3. Answer Processing

The final state of search by question with IR-based factoid questions is extracting a particular answer from the passage. So, the user with an answer like 4,2 light years to the question "How many light years there are between Earth and Proxima b?" could be presented.

Two classes of algorithms have been implemented to the answer processing task, one of them based on answer type pattern extraction and another one based on N-gram tiling.

In the answer type pattern extraction methods for answer processing, information about the expected answer type together with regular expression forms is used. For example, for questions that's answer type is a music , the answer type or entity that is named tagger on the candidate passage or sentence and return which entity is tagged with type music are run. Thus, examples that are in the below, the entities which are bold named are extracted from the candidate answer passages for the answer to the music and location questions:

"What is the name of Tarkan's last album?"

In September Tarkan's last album **10** will be on the market.

"What is the capital of Turkey?"

This international conference will be take place in **Ankara**, capital of Turkey.

On the other hand, the answers to some questions, like definition questions, are not similar to be a spesific named entity type. For these type of questions, instead of using answer types, hand written regular expression forms are used to support for extracting the answer. Moreover, these forms are useful in situations in that a passage includes multiple examples of the equal named entity type. Following examples show some forms for the question phrase (QP) and answer phrase (AP) of definition questions:

Form: <AP> such as <QP>

Question: What is react native?

Answer: Developers use this type of tool that used for building native mobile apps with JavaScript such as react native

or

Form: <QP> ,a <AP>

Question: What is blimp?

Answer: Because of defecting of blimp, a housing attached to a camera which reduces the sound caused by the shutter click, plans are postponed.

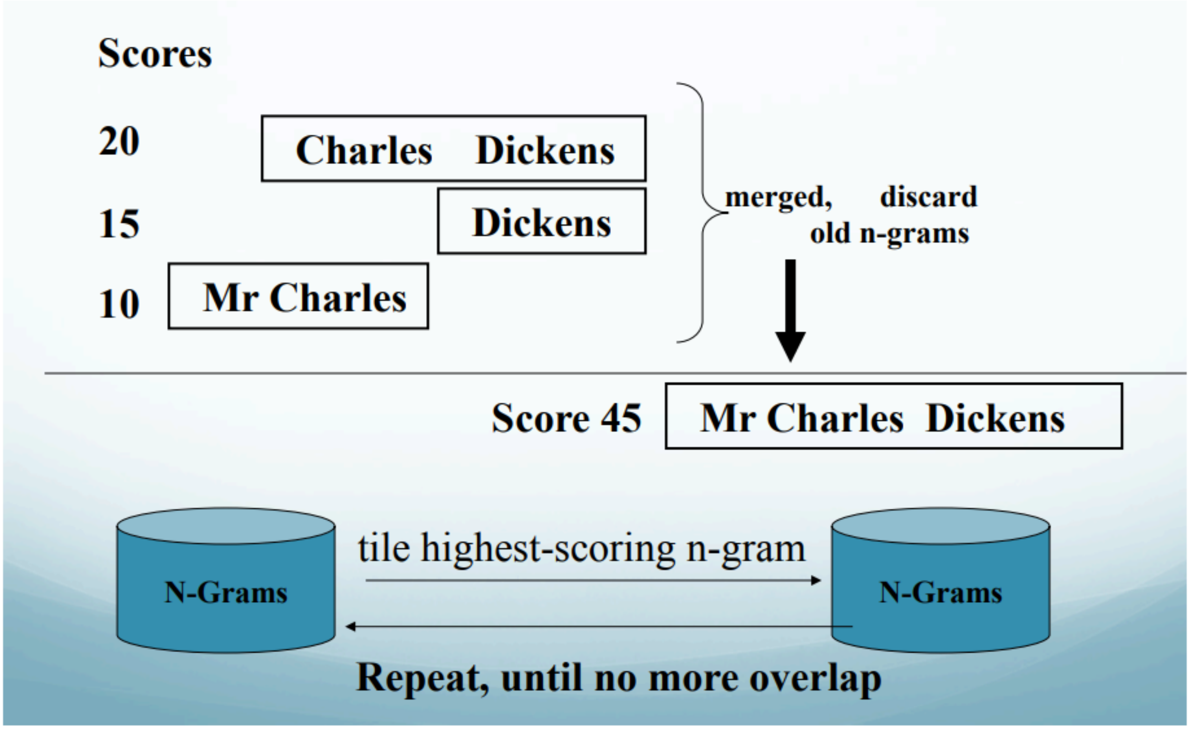

Figure-1: Process of N-gram Tilling

There is a another way to answer extraction, use Web search that is based on N-gram tiling, or its other name is redundancy based approach[1]. First step of this method is that snippets returned from the Web search engine, generated by a formulated query. Then, N-gram mining, every unigram, bigram, and trigram taking place in the snippet is extracted and weighted. The weight is a function of the count of snippets in that the N-gram taking place, and the weight of the query formulation forms that returned it. In the N-gram filtering state, N-grams are ranked by relevancing they match the expected answer type. Scores which are result of ranked are sum of weights of the query that retrieved coincident pages that build for each answer type. Then, canditates forms should be found. This occurs that for a given question kind, detect an question with question term and answer term, submit to search engine, download top X web documents, select relevant sentences for answer term, detect all substring and their numbers, select states with question term and answer term and replace them. The following example shows N-tilling process:

Question: When was Alan Turing Born?

Answer: Alan Turing (1912)

Question Term: Alan Turing

Answer Term: 1912

- Alan Turing (1912-1954) was highly influential in the development of theoretical computer science
- Alan Turing (1912-1954) devised a number of techniques for speeding the breaking of enemy ciphers during second world war.

Phrase: Alan Turing (1912-1954), count= 2

Convert to: <Name> (<Answer>)

2.2. Using Multiple Information Sources: IBM's Watson

There are 4 stages of the DeepQA system, question-answer system of Watson, that is the search by question component of Watson.

The first state is question processing. The DeepQA system runs investigating the grammers, named entity tagging, and correlation extraction on the question. Then, such as the text based systems in Section 2.1, the DeepQA system extracts the focus, the answer kind, other name is the lexical answer type or LAT, and implement question classification and question sectioning.

After that, DeepQA extracts the question focus. The focus is the component of the question that co refers with the answer, and it is used to arrange with a supporting passage. Hand written rules extract the focus. This process is such as a rule extracting for any noun phrase with adverb “this”, and rules extracting pronouns that are such as he, she, him, her.

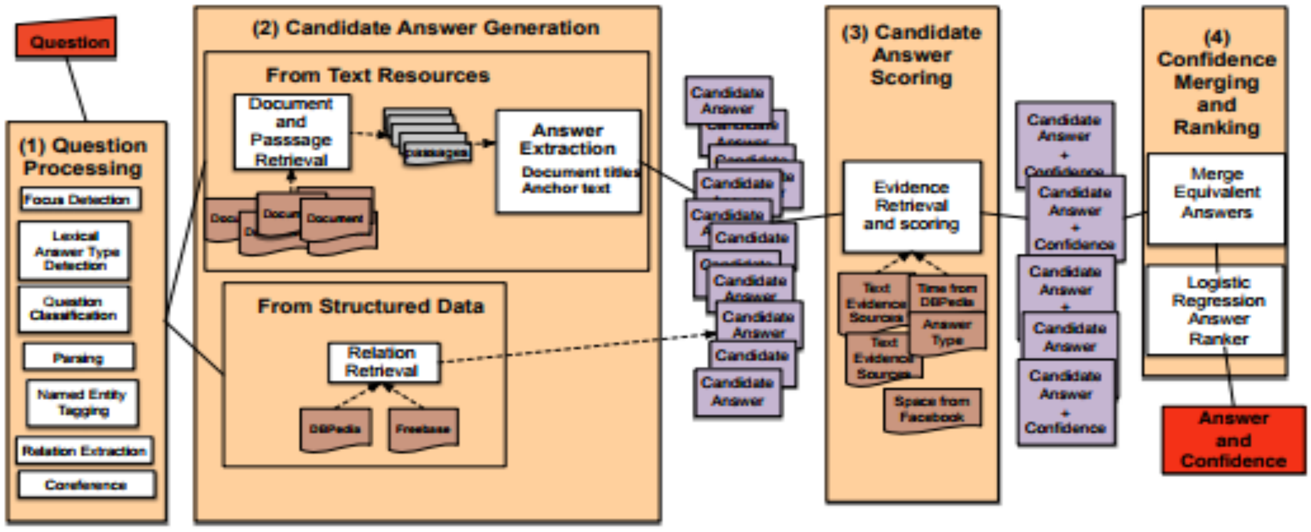

Figure-2: The 4 general states of Watson

The lexical answer type is a one word or words that explain lexical answer type about the semantic kind of answer. Lexical answer types are also extracted by rules: the default rule is to select the syntactic headword of focus. Other rules evolve this default selection. For example extra lexical answer types can be words in question which are have a specific syntactic relation with the focus, like headwords of predicative nominatives of focus. In some cases, category can proceed as a lexical answer type, if it refers to a type of entity that is appropriate with the other lexical answer types. Additionally, using the rules straight as classifier, they could instead be used to return a possibility as well as a lexical answer type.

Difference between DeepQA and IR-based factoid question answerers is that in IR-based question answerers, first answer type is determined, then using a strict filtering algorithm for taked notice text strings which have that type. On the contrary, In DeepQA, lots of answers are extracted, determine a set of answer types, then turn of ‘candidate answer scoring’ state, finally, simply score how well each answer appropriate the answer types to be one of many sources of proof.

In the second state, the operated question with external documents are combined. These candidate answers are extracted from text documents as section 2.1, first generate a query from the question, for DeepQA this is usually completed by sifting stop words, then upweighting any terms that take place in any correlation with focus.

The third state is that use a lot of sources of proof to score the candidates. One of the most significant is the lexical answer type. DeepQA contains a system which attracts a candidate answer and a lexical answer type and returns a score representing whether the candidate answer could be interpreted as subclass or model of the answer type. For example, the candidate “chronic headache” and the lexical answer type “permanent illness”. DeepQA matches these words one by one with probable entities in internal medicine on web sites like DBpedia and Wikipedia. Hence, the candidate “chronic headache” is matched with the DBpedia entity “migraine”, then that model is designed to the Wikipedia kind “symptom”. The answer type “permanent illness” is designed to the Wikipedia kind “indication”.

In the final state, which is answer merging and scoring stage, first candidate answers that are equivalent are merged. Hence, if two candidate answers are extracted like A. Turing and Alan Turing, this state would merge two into a single candidate. For proper nouns, automatically produced name dictionaries could support this task. One useful type of resource is large synonym dictionaries which are composed by listing all anchor text strings which point out to same Wikipedia page; this kind of dictionaries give large numbers of synonyms for each Wikipedia title, for example, A. Turing, Alan Turing, Computer Scientist Alan Turing , Alan Turing who hacked Enigma etc. For widespread nouns morphological parsing to merge candidates could be used that are morphological variants.

In conclusion, in four states of DeepQA which it draws on systems of the IR-based paradigms are analyzed. Actually, Watson's architectural improvement is its confidence on recommended a large count of candidate answers from text based, then progressive a a lot of variety of proof properties to score these candidates again text based. Certainly, the Watson system includes much more components for dealing with complicated and uncommon questions.

3. External Interface Requirements

3.1. User Interfaces

The user interface will be worked on any operating system, the user interface will be worked on web-based systems. In addition to this,help button,shortcut key,standarts for fonts,images,color chemes and message display convention.

3.2. Hardware interfaces

Enough fre space for database size,required web address to Access this system(web adress).

3.3. Software interfaces

The proje needs to run a web space to Access any users,this proje need any web browser such as Google Chrome,Internet Explorer,Mozilla Firefox…In addition to this, the proje needs to analysis some sentences so the system will take to pieces user's question thanks to IBM Watson Library.In addition to this,The system need a database to take some information.

3.4.Communications interfaces

Admin comminication adress,data transfer rates,network communication protocols,live chat services,web browser.

4. Functional Requirements

4.1.Assistant Tools Use Case for Searching

Use Cases

- Virtual Keyboard.
- Question Based Search.
- Answer Sharing.
- Copy Answer.

Diagram

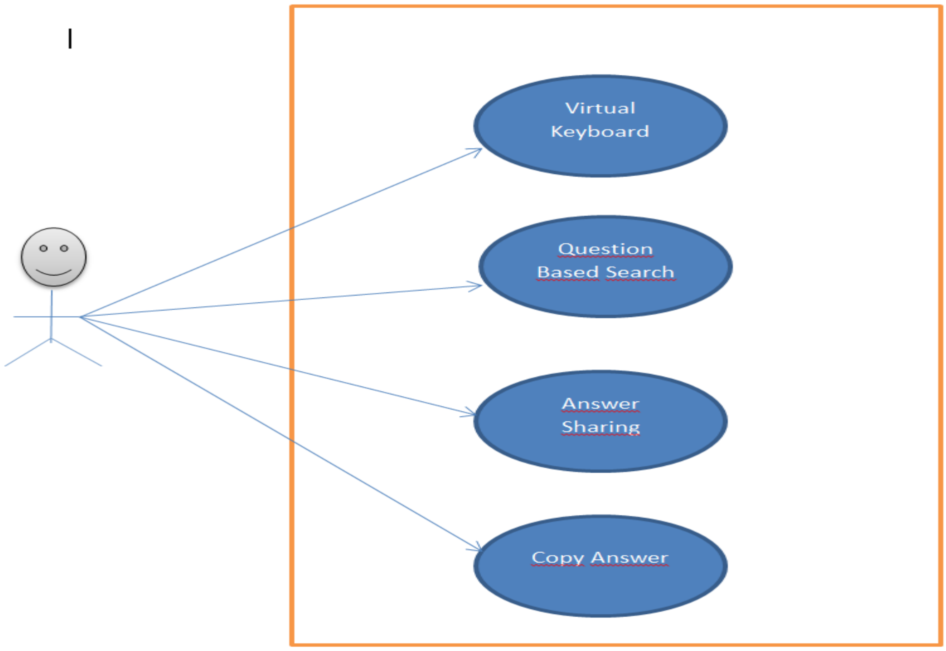

Figure 3 Assistant Tools Use Cases for Searching and happy user

Brief Description:

In Assistant Tools Use Case Diagram(*Figure 1*) explains the basic operations for users. Users are able to use the following functions:

- Virtual Keyboard
- Question Based Search
- Answer Sharing
- Copy Answer

Initial Step by Step Description:

1. User can use virtual keyboard if user doesn't have physical keyboard or users are using mobile phone.
2. If user wants to use Search by Question system, user should ask question to system, otherwise system will not return logical answers.
3. User can share answer of questions on other social medias(Facebook,Twitter,Gmail,etc.).
4. User can copy answer of questions with copy button.

5. Performance Requirement

System requirements for IBM Watson Explorer:

1. Operating system: Mac OS X 10.1 and above,Windows 10(64-bit) Enterprise,Education or Pro editions.
2. 4 CPU cores.
3. 8 GB of memeory.

6. Software System attributes

6.1.Performance

- We should train to increase performance to find true keys onto asked question on IBM Watson Tool.
- As Watson is trained, accuracy and performance of responses will increase on the system.

6.2.Usability

- If user use 5W 1H(how,who,what,where,when,why) question type, user will obtain a answering centences.

6.3.Adaptability

- Questions and answers should accommodate to desired format like 5W1H format.

6.4.Scalability

- There is no scalability for appealing to the general public.

7. Functionality Requirement

The entered text should be question template.When entered text is not question template,the system can run but it does not return expected result.

8. Development Tools

- Bluemix - IBM Cloud System for use Watson.[2]
- Natural Language Understanding.
- Watson Knowledge Studio - This tool will use train to Natural Language Understanding tool.
- Watson Explorer - This tool will use analytics for unstructured data.

9. References

[1] Ta-Cheng Chen, C, “IAs Based Approach for Reliability Redundancy Allocation Problems”, science direct

http://www.sciencedirect.com/science/article/pii/S0096300306005042

Last access date: 08.12.2017

[2] IBM, “IBM Cloud”.

https://console.bluemix.net/catalog/

Last access date: 07.12.2017