Improved Data Driven technique with Answer Selection

[1]Snehal Ukhale, [2]Ruhi Sodani

[1]ME Student, [2]Professor
Computer Science & Engineering Department
MSSCET College Jalna, Maharashtra, India

***Abstract*: Similar queries from historical archives have been used to answer questions with good theoretical foundations and good practical accomplishments. However, each question in the pool of respondents is often related to several answers, and therefore the user needs to look thoroughly before finding the correct answer. To alleviate the problem, we offer new projects to rank candidates who respond via a double comparison. In particular, it consists of an offline learning component and an online search component. In the offline learning component, we will automatically and automatically generate positive and negative training examples in terms of priority pairs introduced by our data-driven observations. We will then present a new model to include all three training examples. The closed form of this form is derived. In the online search component, we collect the answers to the questions listed by finding similar questions. Then we will sort the candidate answers by upgrading the offline model to the reference order. Extensive tests on the vertical and horizontal community questionnaire have demonstrated durability and compelling performance. We have also published code and information to help other researchers.**

***Index Terms*: Community-based Question Answering, Answer Selection, Observation-guided Training Set Construction**

I. INTRODUCTION (HEADING 1)

Verification systems are intended to extract point-to-point solutions over flooding with documents, or even to match pathways, while most data systems do. For example "Who is the Iron Man of India?" The exact answer the user expects is (Sardar Balabh bhai patel), but does not intend to read the text or documents that match the words of the first Indian steelman. Keywords used by search engines are mostly searchable on the web. Answer this question. Recent successes have been reported in a series of survey evaluations that began in 1999 as part of the Retext Text Conference (TREC). The other inconvenience of keyword search queries is the amount of unrelated information retrieved. information Currently, the best systems can answer the facts in more than two-thirds of this assessment. The combination of user needs and interesting results has spurred international attention and activity in answering questions.

We needed a system that allowed users to ask questions in everyday language and get answers quickly and concisely, with enough context to check answers. The current search engine can list the classified documents. It does not provide the answer to the user. Talking to computers in general English more. Natural language interfaces are not indirect user interviews: In a system log, we understand what people think when searching for any information on the web. One dimensional dimension - limited human language understanding - based on traditional ideas of multimedia. The World Wide Web has grown tremendously since its inception in 1992 as a global interconnection system for document sharing between researchers. It has over 130 million domains and has billions of unique URLs, and about 2 billion users reach them.

Currently, the best systems can answer the facts in more than two-thirds of this assessment. The combination of user needs and interesting results has spurred international attention and activity in answering questions. Answer this question. Recent successes have been reported in a series of question-answering assessments that began in 1999 as part of the text conferencing (TREC) conference. This special issue arises from an invitation to the research community to discuss. Performance, application requirements and challenges of the answering system. Answer one question. The search engine needs to analyze the question, possibly in the context of ongoing interaction. The system needs to find one or more answers by consulting on the links. Then, they must provide the perfect answer for the user in the appropriate form or material.

This section provides an overview of some dimensions of this research in terms of:

- Questions
- Classification of Questioners Levels
- Document Retrieval
- Answers
- Answer Generation
- Evaluation
- Clustering
- Framing
- Presentation

The views of this type of question may vary. But generally But the goal is to get precise answers from the system. We can distinguish the question: yes / no questions, "Wh" questions (Mahabodhi founder, what is the distance between Mumbai and lonavala), indirect requests (I want you to list ...) and commands (Name All historical places ...) These are questions. However, the system is

based on the use of the word "Wh" for clues (who needs someone's answer when they need answers in time). There may be problems processing this type of question.

Classification of Questioners Levels: This section presents a classification of different levels of Questioners.

- Casual Questioners: IR retrieval from documents or passages.
- Template Questioners: NLP techniques to parse the questions.
- Cube Reporter: Named Entity tagging and CE to relate entities.
- Professional Information Analyst: name entity tagging and CE and GT.

Document Information Retrieval

When a user posts a question to a seated system on a large database of unstructured data. (Text file) The first order of business is to reduce the stack to just a handful of documents that the answer would be.

Answers

The results are stored in a WX format. The answer may be either long or short. It may be a list or a subtitle. The results are then converted to the required text that the user wants and displayed to the user. For example, if a user needs this reason, they must use a long answer, but the short answer to reading the comprehension test requires a short phrase. This level separates the possible answers for the classification of questions. various There are also different methods for creating. This level separates the possible answers for the classification of questions. Finding Answers: From Extraction - Cut and Paste Samples from Original Document with Answers When answers are drawn from multiple sentences or multiple documents, the coherence of the isolated answers may be reduced by the need to synthesize these parts into a consistent holotype.

Answer Generation:

Usually, the answers to these analytical questions will require multiple data pages. Example 1 below shows the first answer HITIQA created for the Black Sea search query. The answer is composed of only passages messages from the zero conflict frame. The messages of these frames are sorted by date and delivered to the user. The current work is focused on creating answers.

Evaluation:

What makes a good answer? Is the answer better? The short answer may be better. Is there enough context to adjust the selection to the answer? Contexts are helpful if the system displays multiple candidate responses, as it will help the user find the correct answer, even if the answer is not a given answer. However, in other cases, the experience of the TREC questionnaire evaluates [1], indicating that it is easier to provide longer groups with more embedded solutions than shorter ones. We will discuss the issues, assessments and criteria for selecting questions and providing accurate answers in more detail. The reason is that the solution is based on accurate and accurate answers that will make the system user friendly. On the other hand, every user needs a time-saving and easy-to-use system.

Clustering:

We use n-gram grouping and fetching ideas to find the main topics, themes, and key entities in this set. Deleted documents are split into first-occurrences. Duplicate paragraphs are filtered out and the rest are grouped using hierarchical clustering and n-bin categorization. The topic labels are assigned to each group. Title labels may come from one of two positions: First, the messages in the cluster are compared to the list of key phrases extracted from the user's search query. Net words can be consulted to see if there are common ancestors. For example, "gun" and "machine gun" are "weapons" in Word Net. It allows for a match between the questions about the weapons weapon inspectors and the messages reported by the cops' "rifle" and "machine gun" officers.

Framing:

We use a frame-by-frame technique to define the gaps between the meaning of user and system questions. The "Understanding" of this question in the current version of the frame system is a fairly generic template that contains a small number of attributes such as location, person, country, organization, etc.

Presentation & Output:

Finally, in real-time acquisition scenarios, users are interacting with the system in real-time. Users often start with common (and unspecified) questions, and the system displays direct or indirect feedback by returning too many documents. Then the user narrows down the search, thus engaging in talking to the system. Enhance user experience and user satisfaction. In addition, if the interface can handle speech input and conversation, the answering system can be used to access data from the web using information from the web, which is a particularly attractive area of commerce. Especially for telecom providers and web content providers. When a user refreshes the Query to our QA system, the Query is preprocessed by tokenization, pausing, deleting, and storing information. Query then enters the token object object to specify the type of response (example: If the question has keywords such as distance, then the answer should be words such as km, miles, kilograms, miles, etc.). Then use the programming scheme for specific keywords where our system will find the answer with a dash. Effects of natural language In the system we offer, we have developed a QA system for the travel domain. By using the crawler, we have collected web pages, parsed the pages of many travel websites, and processed web pages in advance, by synchronizing information, stopping words and keeping them in the file system.

It identifies and records the corresponding answer strings in the database such as miles, miles, yards, yards, as well as queries for any distance related questions, then associates these data with the token to find answers in the data. Collected from there, the automated token system will take all of these files and retrieve the keywords and store them with their own file URLs in the database, or they can be done manually and automatically. Save the actual database file which will configure the token. To date there has been

little research on the interface for answering questions. There are few systematic assessments of how best to present information to users, how many answers will be offered to users, context to provide services, or complete answers to short answers, along with a summary. This is an area that will get more attention since the commercial response interface is enabled.

Enhance user experience and user satisfaction. This is an area that will get more attention as the commercial response interface is enabled. Users often start with common (and unspecified) questions, and the system displays direct or indirect feedback by returning too many documents. In addition, if the interface can handle speech input and conversation, the answering system can be used to make the discussion of accessing information from the web interesting, especially for Telecommunication providers and web content providers. These systems include respondents who have front-page conversations to data stores and structured systems that try to find answers to questions from text sources such as encyclopedias. The well known BASEBALL questionnaire (Green et al., 1961) is a program for answering questions about baseball games played in the American League during one season. Ask questions such as where the Red Sox lost on July 5? Or how many Yankees play in July? Or even when playing an eight-day team in July, BASEBALL has analyzed the 3 questions set out here to get a natural-language questioning session and try to answer questions by searching for stored information. Is complex, although the current standard applies to the grammar and meaning of the question, but it is limited in terms of {baseball {domain} only and with the fact that This target is the interface to the structure. BASEBALL is the first set of software packages that are designed to be "Natural language interface with database"

II. Related Work

Wherever Developing a system that interacts with human users in natural language is the goal of the artificial intelligence research community. From the 1960s to early childhood, a wide variety of natural language databases have been created, both interactive and language-based. The current QA system can evaluate answers from complex information systems. Many current quality monitoring systems for specific domains, such as specific topics, such as scientific topics or specific types of questions, such as descriptive questions. The problem with current quality assurance is the problem of recovery. Answers to questions are also limited to predefined categories. [1]

Use a broad statistical analysis tool that aims to generate full-scale analysis. The analysis of the constituent elements of the questionnaire is transformed into a meaningful representation of interdependence in the question. [2] The current trend in answering open domain questions as a result of TREC-QA Track.

However, open domain QA systems lack the special domain management for all types of queries, since there is no restriction on user question types or terminology, and it is difficult to build an ontology for open domains. Auto-FAQ [4] and FAQ Finder [5] are two systems that use navigation system through the set of questions found.

They have three common core features:

- The QA systems use a natural language based interface – a user asks his or her question in ordinary English.
- The QA systems answer it by one or several prestored related questions and their answers, if any.
- All system interact with their users through WWW (initially FAQ Finder did not have a Web-based user interface).

The high accuracy of the extraction solution has been greatly achieved by using heuristics. [6] Analyze all questions, and then use many rules to parse the data to classify questions. On the other hand, the learning methodology can automatically generate highly automated high-performance classification programs that take advantage of thousands of questions.

Provide sufficient training information. The effectiveness of the learned classifier program will increase. In addition, the learning categorization program is more flexible than the first one because it can be easily adapted to new areas, and there are many documents describing how to learn the system in question classification [7]. Support for learning by machine approach. [8] Use language format for question description. Or limited identification process may involve parsing questions with different complexity grammar [9], or using a full query extension technique, for example, creating a query based on keywords from an encyclopedia. [10] When specifying the entity type, the remaining task of the query is to determine the type of entity. Additional restrictions on entities that match the description to follow. This step can only be done by separating the keywords from the rest of the questionnaire to match the sentence that is the candidate. This keyword set may be expanded using synonyms and / or morphological forms. [11] QA with limited domains has a long history, starting with systems running through databases such as BASEBALL [12]. And LUNAR [13] use some effective segmentation, which is intended to determine the grammatical relationship in which the question can be answered. In the case where these relationships are linked to the entity that identifies the desired entity, they will be passed as a constraint to bring as considered during extraction the answer.

Proposed QA System is the one of highly enriched and inseparable part of information Retrieval (IR) System. There are many types of IR system protocols are been using in the present day Scenario's, like

Vector Space Model (VSM): This is an algebraic form for representing text documents. [13] The entire document format in collections and query strings is a vector in a limited Euclidean vector space.

Probability Retrieval Models: The first idea The probabilistic pull is proposed by Maron and Kuhns [14] and depends on the probability that the document is related to the query.

Inference Network Model: In this model, data retrieval is modeled as a process of inference in network inference. [15] Most techniques used by IR systems can be used under this model.

Information extraction (IE) is a new technology that enables relevant content to be extracted from existing electronic data. Data extraction is primarily in natural language processing and computer-based linguistics. It also involves an area called a query, which is a way to find information in a similar way to a question answer. In general, the IE process has two main parts. First, the system will distinguish. "Facts" leave the text of the document using local text analysis. The second is to combine these data to create new

facts or facts (through inference). The facts are combined. The relevant information is translated into the desired output format. Many IE systems are present and used in the field. Some types of IE are discussed below.

Automated Content Extraction: Automatic Content Extraction (ACE) is a large-scale evaluation effort for IE systems conducted by the National Institute of Standards and Technologies (NIST). ACE challenges participating systems to find people's reference geographic units. - Politics such as city, state and country, location, physical boundaries, organization and facilities, new messages and live broadcasts.

Named Entity Recognition : The Named Entity identifier (NE) is a unique form of IE that is intended to identify the phrase in the text referring to entities such as organization, organization, date, date, and currency amount and facility. And the meaning of those words..

Template matching is generally known as text-based targeting. Fetching is the search for information within the free text that corresponds to the template provided.

III. Proposed Method

The system that offers new parallel learning to the RANk model, PLANE, which can rank candidates who answer questions from a group of related questions. Specifically, it consists of two components, an offline learning component and an online search component.

In the offline learning component, we set up positive, neutral, and negative training examples in the form of priority pairs suggested by observing our information-driven data. We have demonstrated a new paradigm to incorporate all three of these arrangements and the closed solution of this model.

In the online search component, we collect a set of answers to specific questions by finding comparative questions or similar questions. At that point, we will select candidates to answer the appropriate questions by using an offline training model to judge.

Here, we describe our approach of Question Answering System with a heuristic approach for the steps shown in figure 1. As shown in figure there are 9 main steps in our approach.

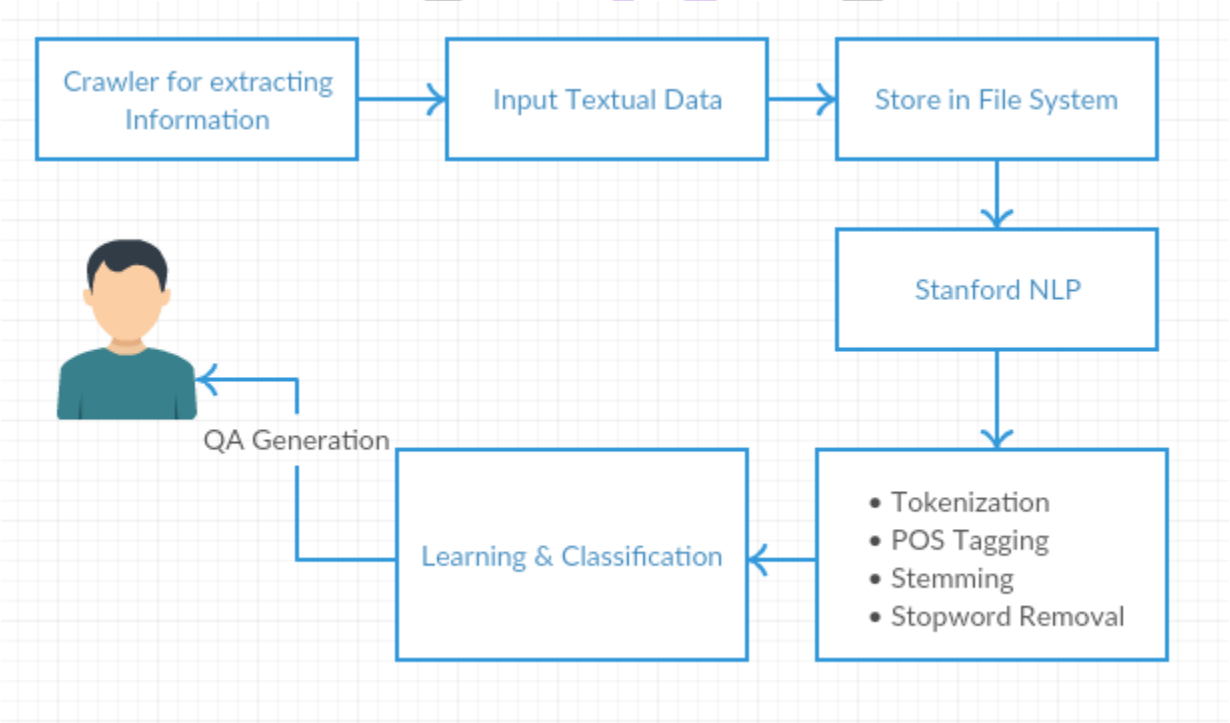

Figure 1 Proposed Architecture

Procedural Steps in Proposed Architecture

Step 1: Users enter questions through the user interface in linguistic form.

Step 2: We are processing user data previously. The query entered by the user leads to the basic meaning of the word, divided into 4 main parts: Tokenization, word deletion, and deletion. The word Tokenization is extracting words into words. The division of sentences is to examine the boundaries and separate the original text into sentences. Next, deleting a stop word is a term that often appears in search terms. It gives less meaning to the content of important documents, such as "a", "an", "the" etc. Pre-processing is Word Stemming; Word separation is the process of removing the prefix and suffix of each word.

Step 3: Key steps in our answering process This is to identify the tokens that answer multiple domain tokens, which make our system more efficient for querying questions. For example, for search queries that are always associated with units of historical places, cathedral hotels, etc.

Step 4: At this stage, decide the quality of the answers offered by the system we offer. Here we choose a number of relevant domain sites that are properly configured. For our guide, we will look at the webpage of the Jala City of Maharashtra in India.

Step 5: Here we are creating a web crawler, which gets the seed URL of the travel domain and finds all the links.

Crawlers are an important component in search engines. Running a web crawler is a challenging task. There are issues with performance and reliability, and more importantly, there are social issues. Data collection is the most vulnerable because it involves contacting hundreds of thousands of web servers and name servers outside the control of the system. The crawl rate is controlled by the speed of its own Internet connection, including the speed at which the site crawls. Especially if one site is crawled by multiple servers, the total crawl time is greatly reduced if multiple downloads are made in parallel. There are many applications available for web crawlers. But the core is basically the same. Here are some web crawler processes that work:

- Download the webpage.
- Parse from the download page and browse all links.
- For each retrieved link, repeat this step.

Web crawlers can be used for crawling across all sites on Inter- / Intranet. When we specify a seed URL, Crawler follows all the links found on that HTML page, which often leads to more links. Been treated again, and so the site can be seen as a root tree structure as a seed URL; All links are in rootHTML-page Direct Roots In the way we propose, we have developed a web crawler using the java programming language, which we use. multithreading Widely used java html parser to parse web pages. And finally, we will keep all web links in the database.

Step 6: One of the most important steps of our testing, where our system interacts with the actual web page of the domain URL. Then by using the child crawler, our system can retrieve page information and then parse all HTML tags from the page. Only human readable data is extracted from the web page, and there are also many advertisements being vomited at this stage.

Step 7: The information collected in step 6 is sent to the method. preprocessing Of step 2 to take information in a simple format and this information is saved in a specific location in the file format.

Step 8: At this stage, our system of tools, tokens and search terms, is working to get answers to specific questions.

Step 9: Now the answer is gathered as one word or several words, then sorted into lists and displayed per user.

Here are the general steps below, which are followed by our system.

- Generates a base vector that contains a set of tokens and keywords. (Eg, miles, distance)
- Separate the number of sentences in the document.
- For each sentence, specify the elements of the main vector.
- Identify the sentence in the sentence. (Here we use a dictionary file to do so)
- If the number of tokens in the sentence increases, specify the nearest token as the noun of the question.
- Divide the answer and separate it from the sentence. Here we will explain how to answer a system question using the solution for the procedure shown in Figure 1. These steps are represented in the form of an algorithm.

Algorithm :

Our approach //
input: Question Qn
//input: Dictionary Set Dc= {d1, d2, d2….dn}
Where dn is dictionary words
// output: Answer An
1: Set Mv = {Tk, Kw} (Master vector, token, keyword)
2: For each sentence Si i=1 to N
3. If Mv Є Si then
4. tag Si as Simp
5. Simp ≠ Dc →Pn (Proper Noun)
6. (Words of Simp) Wi→ Pn → An
7. return

Conclusion

In this work we present a new framework for the selection of questions. Includes offline classes and online search components. In offline learning components, instead of using time-consuming and labor-intensive annotations, we automatically generate positive, neutral and negative training examples in the form of pairs of values defined by the data driven. With our data We then offered some interesting learning partnerships in order to classify these three types of training. In the online search component for a given question, we collect the respondents by asking similar questions. We then use offline learning models to rate respondents through pairwise comparisons.

REFERENCES

[1] Liqiang Nie, Xiaochi Wei, Dongxiang Zhang, Xiang Wang, Zhipeng Gao, and Yi Yang Data-driven Answer Selection in CommunityQA Systems, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING JUNE 2016

[2] Green, W., Chomsky, Laugherty, C. K.(1961). BASEBALL: An automatic question answerer, Proceedings of the Western Joint Computer Conference, pp219- 224.

[3] Woods, W. A.(1973). Progress in natural language understanding: An application to lunar geology, AFIPS Conference Proceedings, Vol. 42, pp.441-450.

[4] Whitehead S. D. (1995) Auto-FAQ: an Experiment in Cyberspace Leveraging. Computer Networks and ISDN Systems, Vol. 28, No. 1-2, pp. 137-146.

[5] Hammond K., Burke R., Martin C., and Lytinen S. (1995) FAQ Finder: a Case-Based Approach to Knowledge Navigation. Proceedings. The 11th Conference on Artificial Intelligence for Applications, Los Alamitos, CA, USA, IEEE Comput. Soc. Press, pp. 80-86.

[6] Hermjakob, U.(2001). Parsing and question classification for question answering, In Proceedings of the Workshop on Open-Domain Question Answering at ACL-2001, Toulouse, France. July 6-11.

[7] Zhang, D. & Lee, W.(2003). Question classification using support vector machines, In Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp.26-32.

[8] Li, W.(2002). "Question classification using language modeling," Technical Report IR-259, Center for Intelligent Information Retrieval, University of Massachusetts.

[9] Srihari, R. & Li,W. (2000). "Information extraction supported question answering" In Proceedings 8th Text Retrieval Conference (TREC-8), NIST Special Publication 500-246.

[10] Ittycheriah, A., Franz, M., Zhu, W, J. & Ratnaparkhi, A. (2001). IBM"s "statistical question answering system." In Proceedings 9th Text Retrieval Conference (TREC-9), NIST Special Publication 500- 249.

[11] Harabagiu, S., Moldovan, D., Pasca, M., Mihalcea, R., Surdeanu, M., Bunescu, R., Girju, R., Rus, V. & Morarescu, P. (2000). FALCON: Boosting knowledge for Answer Engines, The Ninth Text REtrieval Conference (TREC 9), NIST Special Publication, 500- 249.

[12] Scott, S. & Gaizauskas, R. (2001). University of Sheffield TREC-9 Q&A System. In Proceedings 9th Text Retrieval Conference (TREC-9), NIST Special Publication 500-249.

[13] Salton, G., Wong, A. & Yang, C. (1975). A vector space model for automatic indexing, Communications of the ACM, 18(11), pp.613-620.

[14] Maron, M. E. & Kuhns, J. L.(1960). On relevance, probabilistic indexing and information retrieval, Journal of the ACM, 7, pp.216-244.

[15] Turtle, H. R. (1991). Inference Networks for Document Retrieval, PhD thesis, CIIR, University of Massachusettes.

[16] Ellen, M. V.(2001). Overview of the TREC 2001 question answering track, In Proceedings of the 2001 Text Retrieval Conference (TREC 2001).

[17] Franklin, Curt. How Internet Search Engines Work, 2002. www.howstuffworks.com

[18] Grossan, B. "Search Engines: What they are, how they work, and practical suggestions for getting the most out of them," February1997. http://www.webreference.com.