A Cognitive approach in Question Answering System

Shirish Kadam
Amrutvahini College of Engineering,
Savitribai Phule Pune University
shirishkadam35@gmail.com

Amit Gunjal
Amrutvahini College of Engineering,
Savitribai Phule Pune University
amitgunjal26@gmail.com

Abstract

Cognitive Computing is the future of computing and is rapidly taking over the industry. With the amount of information growing exponentially it also poses a challenge to search engines to extract more relevant and contextual information. A question answering system outperforms the conventional search engines in such scenarios. This paper discusses a cognitive approach in question answering systems and proposes an architecture which follows human problem-solving techniques to answer questions. An example is also discussed and explained along with its underlying operations.

1 Introduction

Over the 20th century Human Computer Interaction (HCI) went through lot a of changes and now it has come to a new era. The era where computers will interact with human beings in natural language. But this presents us with challenges such as making computers understand the idiosyncrasies, the idioms, nuances and ambiguities of natural languages. It is quite obvious that with the rise of computers capable to interact and communicate with human beings in natural language, a question answering system will play an important role in it.

A question answering system specifically deals with understanding the user query, extracting the knowledge from different structured and unstructured data sources, drawing logical relationships between facts, generating candidate answers and selecting the most optimal answer from the candidate answer set (D. A. Ferrucci, 2012). A search engine, on the other hand, returns a list of documents which might contain the answer instead of a concise answer. The Question Answering systems in the early 1980s were rule based where these rules had to be manually identified and coded. But clearly, this defined certain undesirable constraints on the systems. In the later period, machine learning techniques were used to identify some patterns in a question to map it to specific rules. But all these systems were domain specific or closed domain and failed to deal with outlier questions, examples of such systems are BASEBALL (Green R. F. et al., 1961) and LUNAR (Woods W. A. et al., 1972). BASEBALL answered questions on specific leagues of US baseball and LUNAR was restricted only to the geological analysis of rocks collected from the Apollo mission.

With the exponential rise of the world wide web in the 21st century, the need of extracting information from web-based documents took over and hence the development of web-based question answering systems became more relevant. The internet made it easy to develop open domain question answering systems as opposed to closed domain systems. START is one such system developed around 1993 and is available at http://start.csail.mit.edu answering natural language questions by presenting textual snippets and multi-media information extracted from the Internet (Boris Katz et al., 2006). It tries to match questions to candidate answers using natural language annotations. LogAnswer is another open domain question answering system implementing 'Theorem Provers' to derive correct answers to the questions. It does so by extracting answers from a logical knowledge representation using precise inference methods (Ulrich Furbach et al., 2008).

IBM Research took a novel approach to solving the challenge of open domain question answering

since 2007. It started its initial ground work based on its existing QA system PIQUANT (J. M. Prager et al., 2004; J. M. Prager et al., 2006). This new QA system called IBM Watson was based on an entirely new architecture called DeepQA defining various stages of analysis in a processing pipeline. It is able to generate multiple candidate answers for a question and assign scores to these answers based on evidences along different dimensions. The DeepQA also trains its statistical machine learning algorithms on prior question sets and their respective answers in order to improve its accuracy (D. A. Ferrucci, 2012).

IBM undeniably is the industry leader in Cognitive Computing and it shook the world when IBM Watson won the Jeopardy! challenge in 2011 overtaking the then world champions. IBM Watson today assists a lot of experts of health care, cancer research, finance, customer care and such industries, in decision making, information extraction, pattern recognition using its cognitive abilities.

Cognitive Computing is the computerized model representation of human thought process or problem-solving. With the help of cognitive computing techniques more contextual, comprehensive information can be extracted and different relationships among different facts in the knowledge base can be derived to obtain precise answers. Various ambiguities and natural language complexities can be resolved.

This paper describes a cognitive approach in question answering system following the four basic principles of problem-solving stated by George Polya in his book published in 1945, "How To Solve It". The architecture proposed here tries to follow these four principles of problem-solving and integrate the processing pipeline with these four principles.

2 Proposed Architecture

George Polya in his book states that solving a problem is a practical skill which can be acquired by imitation and practice. He briefly distinguishes the four phases of problem-solving and states that by following this approach to compute the solution of a problem, mistakes can be identified and eliminated at individual phase (George Polya, 1957).

2.1 Understand the problem

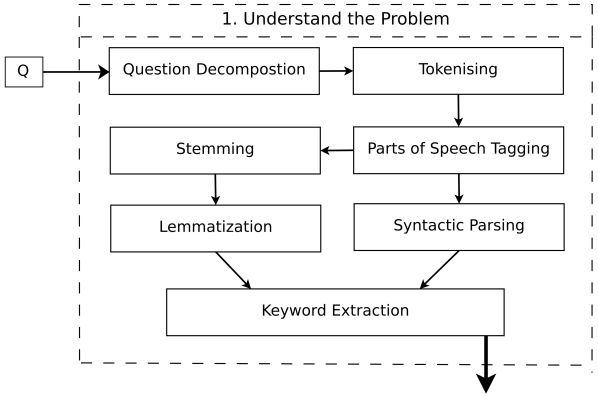

Figure 1. Understand the Problem

Although this might seem an obvious statement but still remains the most vital principal of the problem-solving technique. The goal here is the clear understanding of unknowns and knows. The lack of understanding of the problem leads to a misguided or irrelevant solution. The principal parts of a problem that are important in leading to a solution that needs to be identified are the unknown data, the known data and the conditions or constraints over the requirements (George Polya, 1957). With the help of this information, it is possible to identify the type of the expected answer and its extent. This information can be stored and represented using some internal notations or structures.

```
Question : "Which country has
Hindi as an official language
other than India?"
```

Consider the above question as an input to the question answering system. To properly understand the question, the system should first analyze the grammatical structure of the sentence. It should be able to dissect complex or concatenated questions into sub-questions thus making it easier to answer them individually.

```
Which/WDT, country/NN,
has/VBZ, Hindi/NNP, as/IN,
an/DT, official/JJ,
language/NN, other/JJ,
than/IN, India/NNP, ?/.
```

The question is later tokenized and assigned with the appropriate part of speech tags using statistical methods like Hidden Markov Model (HMM) Tagging or Maximum Entropy Tagging. Assignment of parts of speech plays an important role in stemming, dependency parsing, word sense disambiguation and extracting the named entities. The above example uses the Penn Treebank tag-set for English (Mitchell Marcus et al., 2016). Named entity recognition deals with extracting the proper names occurring in a sentence and classifying them according to their type.

```
INDIA:(GPE), HINDI:(ORG)
```

Syntactic parsing defines the syntactic structure for a sentence in the form of parse trees. A parse tree plays an important role in representation the meaning of the linguistic expression. This process of meaning representation is called as semantic analysis. The parse tree generated below uses the CLEAR NLP tags to define the dependency between the nodes of the tree (Jinho D. Choi and Martha Palmer, 2012).

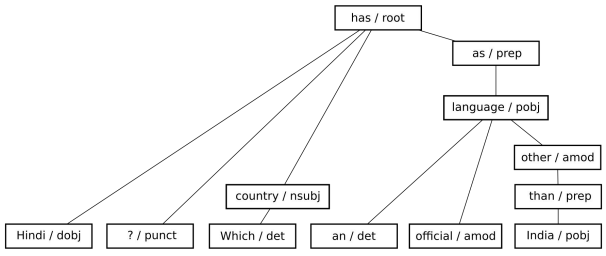

Figure 2. Parse tree

Features like known data, unknown data and constraints of the question can be extracted with the help semantic analysis to produce a meaning representation of the question. This meaning representation is important to construct the correct search query.

2.2 Devise a plan

The main focus of this phase is to devise a plan or an outline to solve the problem. This helps to identify what computations are necessary in order to reach to the final solution. It is suggested by Polya that to devise a good plan the past experiences and previously acquired knowledge plays an important role. The strategy that was used to solve a similar problem in the past can be applied to solve the current problem (George Polya, 1957).

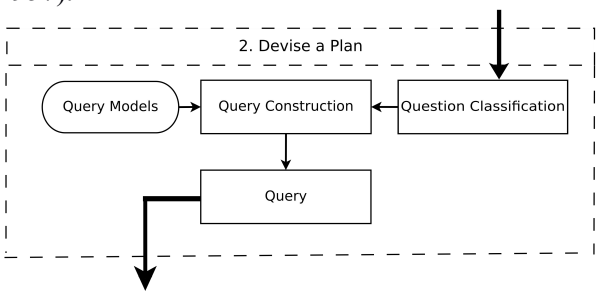

Figure 3. Devise a plan

The features extracted from the previous phase are used to construct a query or a plan which can be executed on the data repositories for extracting relevant documents in the next phase. A frame-based representation of these features can be constructed and stored. This intermediate representation in the form of frames enables us to bridge the gap between our input question and the query constructor. In the frame-based approach, the features are called as slots and the values filling these slots can be atomic values or embedded frames (Mary Elaine Califf and Raymond J. Mooney, 1999).

```
[country:[
      [official language:Hindi]
      other than:India
      ]
]
```

To identify a similar question previously answered by the system question classification is important. When a question is classified and a similar question belonging to the same class is previously answered, in some cases it is easier to follow the plan of the answered question than forming a new one (Zhiheng Huang et al., 2008). A question is classified based on the extracted features and its syntactic structure.

```
QType: (LOCATION)(country)
```

A sample query is constructed below using operators and the frame-based representation (Gideon Zenz et al., 2009). The hierarchy of the features in the frames play an important role to construct the query and define the known and

unknown data in relation to the operators. Here we can use boolean, logical and assignment operators to represent the relationship between the known and unknown features and the constraints. For concatenated questions, the sub-queries of the individual sub-questions can be joined to form the final query. Hence, the highest precedence has to be assigned to the brackets and then logical and boolean operators.

```
(Country)? (Official
Language)? (Hindi)? !=
(Country)?  (India)
```

2.3 Carry out the plan

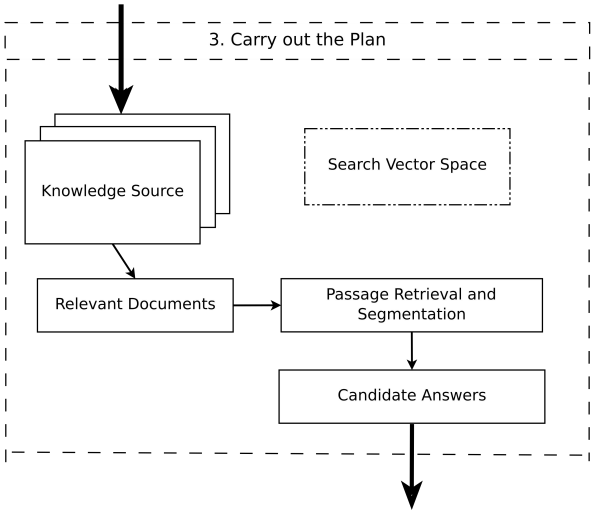

Figure 4. Carry out the plan

Once a plan is outlined, the information that was extracted in the first phase is fitted into the outline and the plan is executed accordingly.

The first two phases concentrate on question analysis and the last two phases on the answer extraction. The query generated in the previous phase is executed in this phase. Initially, the query is executed and all the relevant documents are extracted from the large data repositories. One approach to this is to do a domain analysis of the query and clustering the documents hierarchically according to the domain hierarchy. Frequent itemsets can be used to produce a hierarchical topic tree for clusters. By focusing on frequent items, the dimensionality of the document set is drastically reduced (Benjamin C. M. Fung, 2003).

```
(Country) > (Language) >
(Hindi)
```

This approach reduces the search space dramatically and enables us to focus on the relevant clusters of documents and
even search through the hierarchy levels. From these documents, relevant passages are extracted which might contribute in forming the candidate answers. In passage retrieval, passages in the relevant documents are filtered out that don't contain potential answers and then ranked according to how likely they are to contain an answer to the question. The ranking is based on a small set of features like named entities, keywords in the query and their proximity.

{ C1 : Some ancestral languages that are also spoken in Mauritius include Bhojpuri, Chinese, Hindi, Marathi, Tamil, Telugu and Urdu.} ,

{ C2 : Outside Asia, Hindi is an official language in Fiji as per the 1997 Constitution of Fiji, where it referred to it as "Hindustani", however in the 2013 Constitution of Fiji, it is simply called "Hindi"
The 1997 Constitution established Fijian as an official language of Fiji, along with English and Hindi, and there has been discussion about establishing it as the "national language", though English and Hindi would remain official } ,

{ C3 : nationals of Trinidad and Tobago of Indian heritage or descent. Linguistically they are collectively known as the speakers of the Indo-Aryan Hindustani languages typically Hindi } ,

{ C4 : Outside India, Hindi is an official language in Fiji, and is a recognised regional language in Mauritius, Suriname, Guyana, and Trinidad and Tobago} ,

{ C5 : Surinamese Hindi or Sarnami, a dialect of Bhojpuri, is the third-most used language, spoken by the descendants of South Asian contract workers from then British India. }

Figure 5. Candidate answers

A vector space model is used for information retrieval and answer extraction, where the documents and queries are represented as vectors of features representing the terms that occur within the relevant documents (V. V Raghavan and S. K. M. Wong, 1986). The operations like stemming and lemmatization carried out in the first phase are required here in the search operation to obtain an accurate frequency count for a given term. This

phase generates all the candidate answers for the query. The two classic approaches used in answer extraction are based on pattern extraction and N-gram tiling.

Here it generates five candidate answers using N-gram tiling, determining the accurate answer among the candidate set of answers in done in the next phase (Eric Brill et al., 2002). In N-gram tiling the first step is N-gram mining, where every unigram, bigram, and trigram from the filtered passages are extracted and weighted. In the next step of N-gram filtering, the N-grams are scored and the best-scoring concatenation is added to the set of candidate answers.

2.4 Look back

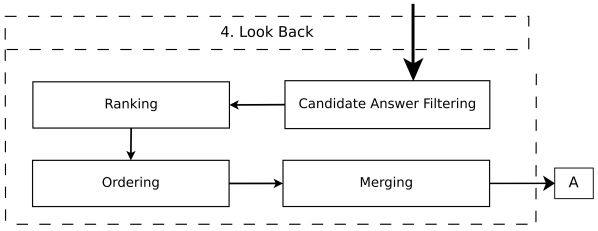

Figure 6. Look back

This is the last phase in problem-solving where each solution needs to be evaluated and ranked in the set of probable solutions. The path leading to the solution needs to be reconsidered and reexamined, to check if the current solution is the most optimal or accurate solution in the solution space (George Polya, 1957). This also provides with an opportunity to learn and understand the path followed to obtain the solution. All the evidence required to support or revert a particular solution is also gathered along the way. On the basis of gathered evidence, a solution can be accepted or rejected and ranked according to its accuracy. The verification of the solution enables the system to detect its shortcomings at a particular phase and improve upon them.

1) C2, C4 (0.9, 0.8) : Fiji
2) C1 (0.5) : Mauritius
3) C3 (0.4) : Trinidad
4) C5 (0.2) : Suriname

In this phase, the candidate answers are ranked based on the N-gram scores and sorted. The highest scoring candidate answers represent a higher probability of an accurate answer and the lower scoring candidate answers are iteratively removed until a single answer is constructed (D. C. Gondek et al., 2012).

```
Answer : "Fiji is the country
where Hindi is an official
language other than in
India."
```

To present the answer phrase in an appropriate sentence three main tasks are performed, content selection, information ordering and sentence realization (Dipanjan Das and André F. T. Martins, 2007).

In the content selection, only the phrases that contribute to the final answer are extracted. The major challenge in content selection is to avoid redundancy of information in multiple documents. A cluster of documents may have a significant amount of overlapping terms and concepts which may introduce an unwanted redundancy in the candidate answers and generate answers that seem to have repetitive information. Among the example candidate answers presented here the answer C2 has two sentences stating a similar fact and also the candidate answer C2 and C4 also state the same fact but are retrieved from different documents. A simple method to avoid redundancy is to explicitly include a redundancy factor in the scores for ranking the candidate answers. After the appropriate content is selected the next obvious process is information ordering where the information is concatenated into a coherent order using Coreference-based coherence algorithms and entity grid representations. Finally, sentence realization involves sentence fusion algorithms to combine phrases from different sentences of the passage containing the final answer phrase. This also involves pruning of unwanted terms from the question to construct the final answer.

The answers can be evaluated using metrics like mean reciprocal rank or MRR, where the evaluation score depends on the reciprocal of the rank of the first correct answers from the initial candidate answer set (EM Voorhees, 1999).

$$\mathrm{MRR} = \frac{1}{n}\left(\sum_{\mathrm{i}=1}^{n}\frac{1}{rank_{\mathrm{i}}}\right)$$

Figure 7. MRR

3 Conclusion

In this paper, a cognitive approach is discussed for a question answering system and an architecture is proposed implementing the problem-solving techniques as stated by Polya. The architecture is also being attempted to be realized in a question answering system and a prototypical example of the system is also discussed in the paper.

In the future, more and more insights of human problem solving can be discovered and the architecture can be improved upon these principles to accurately imitate and even outperform human beings in answering complex questions where large repositories of data have to be researched in limited time constraints. There is also space to integrate a speech recognition system to ease the interaction with the system.

Acknowledgments

The authors would like to acknowledge the contributions of Mathew Honnibal, Explosion AI - a digital studio specializing in Artificial Intelligence and Natural Language Processing, in authoring spaCy, the leading open source library for industrial-strength NLP. Most of the examples presented in this paper were generated using spaCy.

The authors would also like to thank Ms. Swati Bhonde for her valuable suggestions and comments that have greatly contributed in shaping the final version of this paper.

References

Benjamin C. M. Fung, Ke Wang, Martin Ester. 2003. Hierarchical Document Clustering Using Frequent Itemsets.

Boris Katz, Gary Borchardt, Sue Felshin. 2006. Natural Language Annotations for Question Answering.

D. A. Ferrucci. 2012. Introduction to "This is Watson", IBM Journal of Research and Development, Vol. 56.

D. C. Gondek, A. Lally, A. Kalyanpur, J. W. Murdock, P. A. Duboue, L. Zhang, Y. Pan, Z. M. Qiu, C. Welty. 2012. A framework for merging and ranking of answers in DeepQA, IBM Journal of Research and Development, Vol. 56.

Dipanjan Das, André F. T. Martins. 2007. A Survey on Automatic Text Summarization.

E. M. Voorhees. 1999. The TREC-8 Question Answering Track Report.

Eric Brill, Susan Dumais and Michele Banko. 2002. An Analysis of the AskMSR Question-Answering System.

Green R. F., Wolf A. K., Chomsky, K. Laughery. 1961. BASEBALL: An automatic question answerer, Proceedings of Western Computing Conference, vol.19.

George Polya. 1957. How To Solve It, 2nd ed, Princeton University Press.

Gideon Zenz, Xuan Zhou, Enrico Minack, Wolf Siberski, Wolfgang Nejdl, 2009. From Keywords to Semantic Queries - Incremental Query Construction on the Semantic Web, Web Semantics: Science, Services and Agents on the World Wide Web, Vol. 7, Issue 3.

J. M. Prager, J. Chu-Carroll, E. W. Brown, and K. Czuba. 2006. Question answering by predictive annotation, Advances in Open-Domain Question-Answering.

J. M. Prager, J. Chu-Carroll, and K. Czuba. 2004. A multi-strategy, multi-question approach to question answering, New Directions in Question-Answering, AAAI Press, 2004.

Jinho D. Choi, Martha Palmer. 2012. Guidelines for the Clear Style Constituent to Dependency Conversion.

Mary Elaine Califf, Raymond J. Mooney. 1999. Relational learning of pattern-match rules for information extraction.

Mitchell Marcus, Beatrice Santorini, Mary Ann Marcinkiewicz, Ann Taylor. 2016. Penn Treebank, Linguistic Data Consortium, University of Pennsylvania.

Ulrich Furbach, Ingo Glockner, Hermann Helbig, and Bjorn Pelzer. 2008. LogAnswer - A Deduction-Based Question Answering System.

V. V. Raghavan, S. K. M. Wong. 1986. A Critical Analysis of Vector Space Model for Information Retrieval.

Woods W. A., Kaplan R. A, Nash-Webber. B. 1972. The lunar sciences natural language information

system , Technical report, Bolt Beranek and Newman Inc., Cambridge, MA.

Zhiheng Huang, Marcus Thint, Zengchang Qin. 2008. Question Classification using Head Words and their Hypernyms.