A Survey on Question Answering System

By

Biplab Ch. Das
Roll No: 123050068

Under the guidance of

Prof. Pushpak Bhattacharyya

Department of Computer Science & Engineering,

Indian Institute of Technology, Bombay

Question Answering System

Abstract

Question Answering (QA) is a classic application of Natural Language Processing. It has practical applications in various domains like education, health care and personal assistance. In this report we survey history of question answering system and then discuss **IBM Watson**, a QA System built at IBM Research Labs. Question classification is surveyed, conferring to **Bloom's Taxonomy** in Learning Domains.

Table of Contents

Chapter 1:

Introduction to Question Answering System

Computer science has always helped man in making his/her life easier. New era in human history is Information Era. With assistance of web search engines, we can get any information at our finger tips. We are just a click away from accessing a page at remote corner of the world. In addition to the brilliant research, faster computer processors and cheaper memory supported these great advancements.

We have always wanted computers to act intelligent. To accomplish this task the field of Artificial Intelligence came into existence. One of the key barriers in making computers intelligent is understanding of Natural Language. Natural language processing which deals with understanding of languages is sub division of Artificial Intelligence. Question Answering is a classic NLP application. Task that a question answering system realizes is given a question and collection of documents, finds the exact answer for the question. It has two complementary goals: first understand the various issues in natural language understanding and representation and the second to develop natural language interface to computers.

1. Motivation for Question Answering System

We are always in a quest of Information. However there is difference in information and knowledge. Information Retrieval or web search is mature and we can get relevant information at our finger tips. Question Answering is a specialized form of Information Retrieval which seeks knowledge. We are not only interested in getting the relevant pages but we are interested in getting specific answer to queries. Question Answering is in itself intersection of Natural Language Processing, Information Retrieval, Machine Learning, Knowledge Representation, Logic and Inference, Sematic Search. It provides a nice platform to delve into "almost" all of AI. If a statement is made that "Question Answering is the ultimate AI", the statement will be univocally accepted. Question answering system in its being is an art, at the same time it has science in its essence. Question Answering Systems are needed everywhere, be it medical science, learning systems for students, personal assistants. It is necessity in every aspect where we need some assistance from computers. It goes without saying that it is worth exploring the exiting field of question answering.

2. History of Question Answering Systems

Question Answering systems have transformed much in past four decades at par with the whole of natural language processing. It was in the year 1978 when the first classic QA book was published. It was based on Lehnert's thesis where she proposed a question answering system based on semantics and reasoning. We must not be mistaken, question answering systems were at place much before the publication of the book, the story dating back to sixties (Tomek Strzalkowski, Sanda Harabagiu, 2010).

Last four decades witnessed the creation of hundreds of question answering systems. The earliest of such systems, even before 1978 were natural language

peripherals to database access. The supported natural language was restricted and the question answering system was constrained to a particular domain. These systems worked by translating the restricted natural language query to database queries.

Example 1.1:

Question: "List the cricketers who have written books about cricket"
Would generate the following structured query:

SELECT firstname, lastname
FROM sportsman, books
WHERE sportsman.id =books.authors_id
AND
sportsman.game="CRICKET"
AND
Books.type="CRICKET"

Two notable examples of earliest question answering systems were LUNAR and BASEBALL. LUNAR answered questions about rocks that were analyzed during the Apollo Lunar missions. BASEBALL on the other hand answered questions about baseball league over a period of one year. LUNAR and BASEBALL were good at their respective domains. Both of the systems used the techniques used in chatter bot systems.

Terry Winograd at MIT Artificial Intelligence Laboratory (1968-71) developed SHRDLU[1] a question answering system that could perform dialogues based on blocks world. A short dialogue is provided below for the system.

---

[1] SHRDLU is not a short form. Non qwerty typewriters had SHRDLU as the second line of key board buttons. Sometimes we would randomly type asdf as some random characters when we want to fill some garbage information, shdlu is analogous to that.

Example 1.2

Person: Pick up a big red block.
Computer: OK.
Person: Grasp the pyramid.
Computer: I don't understand which pyramid you mean.
Person (changing their mind): Find a block which is taller than the one you are holding and put it into the box.
Computer: By "it", I assume you mean the block which is taller than the one I am holding.
Computer: OK.
Person: What does the box contain?
Computer: The blue pyramid and the blue block.
Person: What is the pyramid supported by?
Computer: The box.

In seventies and eighties natural language processing advanced, resulting in ambitious projects in domain of question answering. One of such project was creation of Unix Consultant or the UC which was developed by Robert Wilensky at U.C. Berkley in later part of 1980s.

1990s saw advancements in the development of question answering systems. We would like to take a look at START, the question answering system developed by Boris Katz. It was first web based question answering system which is continuously running since December 1993.

Israel

Population:
7,707,042 (July 2013 est.)
country comparison to the world: 97

Figure 1.1: A snapshot from question answering system START at CSAIL at MIT.

Late 1990s was era of search engines in incubation. By the mid 2000 search became mature and piggy backing on web search various question answering systems were developed. A notable mention is Open Ephyra, an open source framework for building a question answering system. Open Ephyra is a web based question answering system that can be downloaded from sourceforge[2] It requires just a normal desktop to run the system, however provides performance compared to other state of the art systems.

Nearing year 2010 apple's SIRI was there in apple app store. SIRI is a personal assistant which started as a project funded by DARPA back in 2003. Siri International (SI) was setup up in 2008 after DARPA ended the funding. In 2010 SIRI was launched as a free app, but after four months apple acquired it and integrated it with iOS 4.

Meanwhile the researchers at IBM were working on "Watson" where they were challenged to compete with the champions at Jeopardy game show. The project started in 2007 and was finished in 2011, where it made its public appearance. It was

---

[2] http://sourceforge.net/projects/openephyra/

able to beat two Jeopardy Champions. It was a breakthrough in open domain Question Answering System.

As we proceed through the report we will be presenting details of the working of Watson at the same time mentioning what is known of SIRI.

3. Agenda of Question Answering

As early as 2002 a group of researchers[3] wrote a roadmap of the research in the field of question answering. They also identified the issues related to question answering. The following discussion is based on the issues[4] they identified during their research.

1. Question classes
2. Question processing
3. Context and QA
4. Data sources for QA
5. Answer extraction
6. Answer formulation
7. Real time question answering
8. Multilingual (or cross-lingual) question answering
9. Interactive QA
10. Advanced reasoning for QA
11. Information clustering for QA
12. User profiling for QA

---

[3] Burger, J., Cardie, C., Chaudhri, V., Gaizauskas, R., Harabagiu, S., Israel, D., Jacquemin, C., Lin, C-Y., Maiorano, S., Miller, G., Moldovan, D., Ogden, B., Prager, J., Riloff, E., Singhal, A., Shrihari, R., Strzalkowski, T., Voorhees, E., Weishedel, R. Issues, Tasks and Program Structures to Roadmap Research in Question Answering (QA)

[4] The issues have been listed as given in wikipedia.

3.1 Question Classes

A question may belong to different category and depending on its category we require different strategies to answer the question. We may have a line of attack for a category of factoid questions, which on the other hand will not work for questions that require deeper understanding of facts. We need a profound understanding of what category a question belongs.

Example 1.3:

Recall Question or Factoid Questions seek for Fact:

Question: Who is also known as "chacha", and was born on 14th November?

However questions like:

Question: Why is sky blue in color?

Requires understanding of not only facts but we must have knowledge of Scattering of Light.

As we will discuss later, questions can be classified based on its form in hierarchy. We discuss such classifications in Chapter dedicated to bloom's taxonomy.

3.2 Question Processing

Same question may be asked in different forms. We may ask it in interrogative way or the assertive way. We need to understand the semantics of the question. We need to recognize what the question is asking for before proceeding to answer the question itself. The practice to understand the question is termed as Question Processing.

Example 1.4:

We are seeking same information using different forms of the question:

Interrogative:

Question: What is the capital city of India?

Assertive:

Question: Tell me the name of city which is capital of India?

3.3   Context and QA

Questions are always asked in a context. Questions are rarely asked in universal context. We are required to have a knowledge of context before proceeding to answer a question.

Example 1.5:

Question: Where is Taj?

For a person in America:

He is interested in finding the location of Taj Mahal in Agra.

For a person who just reached Mumbai:

He is interested in finding Taj Hotel.

3.4   Data Sources and QA

Before we can answer questions we require sources which are relevant and exhaustive. We require a data source which will act as the base for all the information required for answering the questions. It may be collection documents. It can be the whole web which we can search for. It can be a database from where we can get the answers for structured queries.

3.5   Answer Extraction

Depending on the question we may want to extract specific type of information from the data sources. Reliant on the complexity of the question we would like to know the expectation of user.

Example 1.6:

Extraction of a Name:
Question: Who is eldest brother among Pandavas?

Extraction of time or date:
Question: When was first battle of panipat fought?

Extraction of Place:
Question: Where was Gandhi born?

3.6   Answer Formulation

Simple extraction may be enough for certain questions. We may want the partial answers to be extracted from various sources and combine them. At the same time, we want the results of the QA System to be as natural as possible. For generating answers we require to generate answers which is termed as answer formulation.

3.7   Real time question answering

We need to answer questions, even the complex question must answered in matter of seconds. People would not like to wait for hours in front of computer to get

answers to questions. Watson for instance when played Jeopardy was able to answer to in average of 3 seconds. We need to develop architecture such that the end product is a real time system.

3.8   Cross lingual or Multilingual QA

Cross lingual QA or Multilingual QA is based on seeking answers in sources other than the language the question was posed in. There are many data sources for English Question Answering system to look for. But other languages like Hindi have a lack of such resources. So we translate a given query to more resourceful language and get the answer which is translated back to original language.

Example 1.7:

Question: इंडोनेशिया के राष्ट्रपति कौन है?

We assume that we do not have answer to this question in our Hindi sources. We translate the question to:
Question translation: Who is the president of Indonesia?

Then we will extract "Susilo Yudhoyono" in English and return the answer "सुसीलो युधोयोनो" after translating it to Hindi.

3.9   Interactive QA

We do not want a boring question answering system that just answers questions that we pose to it. We will like the QA system to be interactive and clear doubt in case it finds the question ambiguous.

Example 1.8:

Person: Please tell me some place where I can eat?

Computer: *Veg or Non Veg*

Person: *Veg will be fine.*

3.10   Advanced reasoning For QA

We want the QA system to not only reproduce what is there in the text collection. We want it to do more. We want it learn facts and apply reason to generate new facts which will be helpful in answering the posed question.

Example 1.9:

These are statements in the corpus:

Statement 1: Ravi is Mohan's brother.

Statement 2: Brother of father is uncle.

Statement 3: Ravi is father of Saini.

Question: Who is uncle of Saini?

Can be answered by reasoning.

3.11   Information Clustering for QA

We want to categorize information according to its type so that the answer search becomes efficient. After we know the category of question we will only search for the document categories which are relevant for our question.

3.12   User Profiling for QA

We would like to know intention of the user by means of analyzing his/her previous queries. For accomplishing the task we need to build a profile of user. We would like to get the answers based upon the taste of the user.

Example 1.10:

Question: What are stuffs to do in Spain?

For a Football fanatic:

Answer: "Real Madrid and Barcelona Match".

For adventurist:

Answer: "Bull Run".

4. Roadmap

 In this chapter we started by introduction to history and agendas of question answering system. In chapter Bloom's Taxonomy we look at the Blooms Taxonomy for classifying questions. We discuss IBM Watson and its architecture in chapter IBM Watson.

Chapter 2:

Blooms Taxonomy

Blooms Taxonomy was developed in 1956 under the leadership of Dr. Benjamin Bloom to promote higher level of thinking in educational domain. The idea was to promote in student the higher mode of thinking such as analyzing and evaluating rather than just recalling facts.

1. Bloom's Classification of Learning Domain

Researchers under bloom identified that educators must improve learning of the students to promote higher level of thinking. In the year 1956, a framework was created with three domains of learning:

1. The cognitive - knowledge based domain.
2. The affective - attitudinal based domain.
3. The psychomotor - skills based domain.

Sometimes, we categorize cognitive as head/knowing, affective as feeling/heart and psychomotor as hands/doing. Blooms taxonomy is considered as the foundational and indispensable constituent in educational technology. In the later sections keeping in mind the goal of building question answering system, we would like to categorize the questions according to difficulty of answering them. Before going to

categorizing questing we would briefly describe the levels in other domains (Benjamin S. Bloom, 1984).

2. Bloom's Taxonomy for Affective Domain:

This section briefly identifies the levels in affective domain of learning. Learning in affective domain is to acquire feelings, attitude towards other fellow human being. The five levels in affective domain of learning are:

1. Receiving
2. Responding
3. Valuing
4. Organizing
5. Characterizing

Receiving is the lowest level in the hierarchy and in this level student pays passive attention. No learning happens, receiving is about students' memory and recognition. In the responding level, the student not only listens but responds to the stimulus. At the level of valuing he attaches value to phenomenon, object and the pieces of information. He attaches values to information he acquired, he doesn't treat object as only object at this level. In organizing level he learns to organize the values he learnt in a schema. He compares and relates them expounding on what he has learnt. On the top most level the student makes a value as a belief, making it as the characteristic of him.

3. Taxonomy for Psychomotor Domain

Bloom and his colleagues never created sub categories in the psychomotor domains, but later in 1972 Simpson created taxonomy for psychomotor domain. Psychomotor

domain of learning is about acquiring skill to perform physical actions which are organized into 7 levels:

1. Perception
2. Set
3. Guided Response
4. Mechanism
5. Complex overt response
6. Adaptation
7. Origination

Perception is the skill of using sensory units to guide motor activities, like estimating the position of a moving cricket ball in air to catch it with hands. Set is level of readiness to act, it includes not being ready physically, but mentally and emotionally. Sometimes it can be seen as related to the respond level in affective domain. Guided response level is level at which the student learns about a complex skill by imitation, trial and error. Learning to solve a complex mathematical equation using the help of the instructor is considered a skill at Guided Response level. Mechanism is level of habitual responding. It is learning a skill at an intermediate level, which may include performing operations on personal computer or closing a leaking tap. Skillful performance is dependent on accuracy, efficiency and coordinated movement patterns. Level at which such skills are operated is Complex overt response level which includes parking a car in tight parallel parking slot. Adaptation is level at which he can perform actions which are unexpected. Origination is the final level which involves the creation of new movement patterns for performing some action. An example may be create a form of dance, or we can think of trademark helicopter shot of Mahendra Singh Dhoni[5].

---

[5] Indian cricketer, captaining at the time of writing of report.

4. Question Classification under Bloom's Taxonomy

In this section we will learn about classifying question with respect to Bloom's taxonomy on basis of cognitive difficulty. The questions can be classified into six levels at cognitive levels:

1. Knowledge
2. Comprehension
3. Application
4. Analysis
5. Synthesis
6. Evaluation

Knowledge questions concern with remembering previously learned information. Comprehension is demonstration of understanding of facts. Application concerns with applying the knowledge in actual situations. Analysis is about breaking the questions into simpler parts and find patterns for generalization. Synthesis questions test the ability of a student to compile the existing knowledge to generate new ideas or propose alternative solutions. Evaluation questions are questions that expect the students to make and defend judgments on external or internal criteria. In each of the following sub sections we look at these questions in details.

4.1   Knowledge questions

Knowledge questions[6] demonstrate skills of observation and recall of information. These questions test the knowledge of dates, events and places. In addition, it also includes the question which require knowledge of major ideas and mastery of subject matter.

---

[6] The questions and the skills they test have been adapted from
http://www.coun.uvic.ca/learning/exams/blooms-taxonomy.html

The following action verbs[7] are often used while framing the recall or knowledge questions:

Arrange, Define, Describe, Duplicate, Identify, Label, List, Match, Memorize, Name, Order, Outline, Recognize, Relate, Recall, Repeat, Reproduce, Select, State.

Example 2.1:

1. *What is Melanesia?*

2. *Define hypervolemia?*

3. *Arrange "Gold, Platinum, Sodium" based on reactivity?*

4. *Define the term "Photosynthesis"?*

5. *Describe the process of "Digestion"? (Recall/comprehension)*

6. *Find the Duplicate entries "A, S, D, A, F"?*

7. *Identify the odd one out: "chess, cricket, football".*

8. *Label the place with highest rainfall in the map.*

9. *List the metro cities in India?*

10. *Match the country to its capital:*

    Pakistan: Jakarta

    India: Islamabad

    Indonesia: New Delhi

11. *Memorize the table of 9.*

12. *Name the prime minister of India?*

13. *Order the following according to the dictionary order:*

    Bat, Mat, Cat

14. *Outline the process of making an omelet?*

15. *Recognize the face given in the picture.*

16. *Relate the person to event.*

17. *Recall the difference between NFA and DFA?*

18. *Repeat the following lines ten times.*

18.1 *"Mary had a little lamb"*

19. *Reproduce the similarity between a HMM and a CRF?*

20. *State the "First law of motion".*

4.2   Comprehension questions

Comprehension questions are posed to test the user standing of the concept, grasp meaning, translate knowledge into new context, interpret facts, compare, contrast order, group, infer causes and predict consequences. Action verbs associated with comprehension questions are:

Classify, Convert, Defend, Describe, Discuss, Distinguish, Estimate, Explain, Express, Extend, Generalized, Give example(s), Identify, Indicate, Infer, Locate, Paraphrase, Predict, Recognize, Rewrite, Review, Select, Summarize, Translate.

Example 2.2:

1. *Classify the following based on whether they are metal or not: Oxygen, Carbon ,Gold*

2. *Convert 10 km to miles.*

3. *Defend whether Mount Everest is taller than K2?*

4. *Describe the process of "Embryo Formation"?*

5. *Discuss the similarities and difference between stars and planets.*

6. *Distinguish between ATP and ADP?*

7. *Estimate the time taken by light to reach earth from sun?*

8. *Explain the reasons why light is particle*

9. *Express in your own words the Framing of Indian Constitution*

10. *Extend the tree kernel to incorporate dynamic programming.*

11. *Give example(s) of Electrical Items.*

12. *Identify the truth of statement "This statement is false".*

13. *Indicate the points of maximum and minima in graph of x^2*

14. *Infer if Babar was more popular than Asoka.*

15. *Predict the next outcome in the sequence 1,2,3,5,8,---*

16. *Recognize the famous pattern in the above sequence*

17. *Rewrite the sequence in such that this time you take the difference of consecutive terms.*

18. *Review the "Von Newman" architecture of Computers.*

19. *Select the one with lowest training time for POS tagging HHM,CRF,MEM*

20. *Summarize what happened in economics during 2nd world wars.*

21. *Translate "What is your name" to Japanese?*

4.3   Application questions

Application questions demonstrate use information use methods, concepts, theories in new situations, solve problems using required skills or knowledge. The action verbs associated with application questions are:

Apply, Change, Choose, Compute, Demonstrate, Discover, Dramatize, Employ, Illustrate, Interpret, Manipulate, Modify, Operate, Practice, Predict, Prepare, Produce, Relate, Schedule, Show, Sketch, Solve, Use, Write.

Example 2.3:

1. *Apply the separating hyper plane theorem to two points.*

2. *Change the co-ordinate system and find the new co-ordinates.*

3. Choose the best alternatives viable from the electricity or gas based stove.

4. Compute the area of the circle with radius 3.14?

5. Demonstrate the question answering system created by Biplab das, after you log in to 10.105.19.19

6. Discover any hidden python files that are required to run the QA System.

7. Dramatize the Life of alexander in 100 pages

8. Employ the techniques developed for dynamic programming to derive Viterbi algorithm

9. Illustrate the application of Dijkstra's Shortest path algorithm

10. Interpret the cipher text that we intercepted between USA and Russia.

11. Manipulate the algorithm of single variable to incorporate multivariable case.

12. Modify the Standard CYK Algorithm to get the parse tree also.

13. Operate the machine in R mode.

14. Practice the problems given in Cormen for better understanding?

15. Predict whether it will be a rainy day, day after tomorrow?

16. Prepare the flow chart for hello world program.

17. Produce the steps required to get the solution for $x^2 - 2 = 0$?

18. Relate the concepts of Urn with balls and POS Tagging?

19. Schedule the timetable for the classes going on at IITB

20. Show that two lines can intersect at three point only if they coincide

21. Every integer number has a integer number greater than it. Sketch the outline of you proof using mathematical induction.

22. Solve for x: $x - e^{(x^2 - x\tan(x))} x = 2x \left(\frac{1}{e^x x} \right)$

23. Use the results from string theory to provide existence of gravitation.

24. Write the functions which have saddle points.

4.4   Analysis questions

Analysis questions are designed to demonstrate the skills of seeing patterns, organization of parts, recognition of hidden meanings, and identification of components. The action verbs associated with analysis questions are:

Analyze, Appraise, Breakdown, Calculate, Categorize, Compare, Contrast, Criticize, Diagram, Differentiate, Discriminate, Distinguish, Examine, Experiment, Identify, Illustrate, Infer, Model, Outline, Point out, Question, Relate, Select, Separate, Subdivide, Test.

Example 2.4:

1. *Analyze the following matrix to check if it is PSD.*
2. *Appraise the mathematical tool, HMM for its applicability in POS Tagging*
3. *Breakdown the Question Answering System into smaller parts.*
4. *Calculate the integral of x^4 where x is in [2,4]*
5. *Categorize the questions according to blooms hierarchy.*
6. *Compare the different models of Cognition*

7. *Contrast the different approaches for the fight for Indian Independence*
8. *Criticize the role of Role of Shah Rukh Khan in movie "Baadsah"*
9. *Draw a Diagram for the cycles of Heart Beat.*
10. *Differentiate between the form 0/0 and 1/0.*
11. *Distinguish between intra lingual and inter lingual translation*
12. *Examine the hardness of Stainless Steel.*
13. *Experiment with various Steels to find the exact carbon content.*
14. *Identify the best method to create money with zero investment.*
15. *Illustrate the above method in real world.*

16. *Infer whether the method worked or no*

17. *Model the money making method using probabilistic models.*

18. *Outline the difficulties faced in making an actual QA System*

19. *Point out the points of singularity in the expression 1/x.*

20. *Question if singular points actually occur in universe?*

21. *Relate the mathematical concept of singularity with singularity in physics.*

22. *Separate the mixture of gold and bronze*

23. *Subdivide the process of Election in sub processes.*

24. *Test if White powder present is Sodium Bicarbonate.*

4.5 Synthesis questions

Synthesis questions are the next class of question which require synthesis of newer ideas to answer the question. It demonstrates the skills such as use old ideas to create new ones, generalize from given facts, relate knowledge from several areas, predict, and draw conclusions. The action verbs associated with Synthesis questions are:

Arrange, Assemble, Categorize, Collect, Combine, Comply, Compose, Construct, Create, Design, Develop, Devise, Explain, Formulate, Generate, Plan, Prepare, Rearrange, Reconstruct, Relate, Reorganize, Revise, Rewrite, Set up, Summarize, Synthesize, Tell, Write.

Example 2.5:

1. *Arrange the parts to create a new bicycle.*

2. *Assemble various classifiers to make a fast and scalable Answer type classifier.*

3. *Collect information regarding QA System and then create something new.*

4. *Combine Statistical frameworks and Logic to get MLN.*

5. *Compose a lyrics for the tune "sa-re-sa-ga-ni-sa ....*

6. *Construct a linear classifier for checking if a mail is spam.*

7. *Create a kernel based approach to check if the mail is spam.*

8. *Design an algorithm to count number of spanning trees for a graph*

9. *Develop an algorithm to rank the web pages the web graph.*

10. *Devise a method to put pumpkin inside an earthen pot.*

11. *Explain how you can put a pumpkin in earthen pot*

12. *Formulate web graph ranking as a convex optimization problem.*

13. *Generate different cases where the Stanford parser fails.*

14. *Plan the building construction according to government norms.*

15. *Prepare a draft for the book that you are writing.*

16. *Rearrange the steps in prolog program to check if time taken changes.*

17. *Reconstruct the recursion program as an iterative program.*

18. *Relate the iterative and recursive versions of the Fibonacci search.*

19. *Reorganize the recursive version of the algorithm to make it as a tail recursive program.*

20. *Revise the techniques presented for summarization.*

21. *Synthesize a protein molecule that can cure cancer.*

4.6   Evaluation questions

Evaluation questions are posed to test the evaluation skills of the student. With respect to the skills it requires to answer such questions are compare and discriminate between ideas, assess value of theories, presentations make choices based on reasoned argument, verify value of evidence and recognize subjectivity. The action verbs associated with evaluation questions are:

Appraise, Argue, Assess, Attach, Choose, Compare, Conclude, Contrast, Defend, Describe, Discriminate, Estimate, Evaluate, Explain, Judge, Justify, Interpret, Relate, Predict, Rate, Select, Summarize, Support, Value.

Example 2.6:

1. *Evaluate whether binary search on a sorted list is faster than linear search.*

2. *Appraise the performance of the new Employee based on the project he was working*

3. *Argue whether we are going to fire him or not.*

4. *Choose the reasons to be and not be an atheist.*

5. *Compare the communist and the extremist ideas.*

6. *Conclude the discussion on the meaning of life?*

4.7   Summary

We began the chapter by introduction to Bloom's Taxonomy. We looked at the Learning, Affective and Psychomotor domain in Bloom's Taxonomy. In the learning domain we looked at the different types of questions under Bloom's Taxonomy. We discussed about knowledge questions, comprehensive questions, applications questions, synthesis questions and evaluation question

Chapter 3:

IBM Watson

In this chapter we will review the series of 17 papers by IBM research on Watson. The introductory section "This is Watson" covers the brief history of Watson and the pipeline of IBM Watson Architecture. Then we move on to Question Analysis module of Watson covering "Question analysis: How Watson reads a clue" which covers the first stage in IBM Watson. The question analysis module uses the Parsing capabilities of Watson which will be addressed in the section "Deep parsing in Watson". One of the issues identified is the introduction chapter in Question Answering System was data sources for QA System, with respect to Watson we will cover "Textual resource acquisition and engineering". Automatically extracting relation is an important step in question answering system, so in the section "Automatic knowledge extraction from documents" we review the methods Watson employs to extract relations from a textual corpora. A key to development of robust question answering system is Hypothesis generation for a question, we discuss "Finding needles in the haystack: Search and candidate generation" which covers issues concerning the strategies to search and get correct answers to the posed question. A 5% increase in capability of Watson comes from understanding the type of answer required. In section "Typing candidate answers using type coercion" resolves issues regarding the typing of answers. Supporting Evidence Retrieval was a novel method devised by researchers at IBM Watson which will be looked at in the section "Textual evidence gathering and analysis". We come back to relation extraction again, we discuss "Relation extraction and scoring in DeepQA" which in addition talks about the scoring mechanism in IBM Watson. Although most of the resources in Watson comes from unstructured text corpus, there are components

which deal with organized information. "Structured data and inference in DeepQA" shades light on this area. Jeopardy game show has special questions which is addressed in "Special Questions and techniques". To answer natural language questions we need to understand the hidden associations which is discussed in "Identifying implicit relationships". The question contains more than what we can see. "Fact-based question decomposition in DeepQA" deals with understating the fine grained analysis of complex questions. The final stage deals with ranking answers which is looked in section "A framework for merging and ranking of answers in DeepQA". With such complex computation going inside Watson, it hard to think of Watson being real time. "Making Watson fast" deals with ideas and research to make Watson fast enough to compete with the Jeopardy contestants. We end with discussing the interface design of Watson in section "In the game: The interface between Watson and Jeopardy". We conclude the discussion by presenting the report "Watson: Beyond jeopardy".

1. Overview of Watson

In 2011, an open domain question answering system was able to beat two of the best Jeopardy players. The paper "This is Watson" is about the events that lead to genesis of Watson, at the same time it covers the architecture and pipeline processing of IBM Watson. It is a narrative of the algorithms working inside Watson. The discussion is concluded with the researchers providing their future plans on Watson.

Considering what Watson has achieved the technical challenges to build Watson is covered with a brief history of Watson (Ferrucci, D. A., 2012).

On January 14, 2011 something different happened on a television quiz show that was in air in US for over more than 25 years. A computer was playing the game, real time against the best of the human contestants. The historic match was televised on February 14-16 by Jeopardy productions. But the story dates back much before 2011.

Most of the resources for information is unstructured for humans understanding the unstructured information is relatively easy. But it's very difficult for computers to automatically understand the natural language. With large amount of electronic data available, we would like to use the information present in the data to make decision. Question Answering is a way to explore this field where we want to infer the answers processing unstructured information.

Natural language Processing Techniques which is also know by the names computational linguistics or text analytics, tend to explore the meaning of the textual data by analyzing the syntax, semantics and context of the phrases in the data. After many years of research nice techniques have been developed for handling these issues at different levels. The techniques may range from part of speech tagging by hidden markov model to parsing, anaphora resolution, text entailment and much more. Much of research has also gone in Question Answering System, but building an end to end question answering system is difficult. In the time period 2001-06 the researchers at IBM built Unstructured Information Management Architecture (UIMA) a framework to integrate various computer software, each different and independent to work together. UIMA provided an architecture for feasibility of building a real-time and scalable question answering system.

In May 1997, as one of the ambitious project by IBM, Deep Blue computer beat Gary Kasparov, which brought IBM Research to lime light. One might ask a question whether it is hard to win a chess game or build a question answering system? It may seem chess requires much of intellect, but ever it has a defined set of rules, something computers are very good at. Question Answering in open domain is however very difficult for computers.

In 2006 the head of IBM Watson challenged the researchers to build a question answering system that will be able to compete at Jeopardy. A notable mention is PIQUANT question answering system which was developed by IBM dating back in the year 1999. However the system was not able to compete with contestant at Jeopardy.

1.1 What it takes to win at Jeopardy

To compete with the contestants at Jeopardy the system has be self-contained and must not be able to connect to web search. It has to analyze the questions, search for the answers in the resources it has and then should be able to answer the questions. All the steps has to be done in matter of seconds. It has to do all the analysis relate the questions to answers and then must buzz before the contestants. It must be confident enough to answer the question and if not confident must pass to the next contestant.

To analyze how good humans are at Jeopardy, prior data of 2000 games were collected. It was found that on an average a player attempts 40% to 50% of the questions, and gets between 85% and 95% of the answers correct. On basis of this a performance goal was set, which aimed at Watson to buzz for 70% of the time and get 85% of them correct. A term was coined for the performance: 85% precision at 70% answered or 85% Precision @70. To win at Jeopardy, the system must get such accuracy and must be able to do that in matter of seconds.

1.2 Inside DeepQA: The architecture underlying Watson

Watson employs hundreds of Algorithms in pipeline to get answers to the question posed to the system. In 2207 baseline version Watson was made over PIQUAINT which was able to deliver 16% Precision@70. Hence an architecture was developed to improve the system.
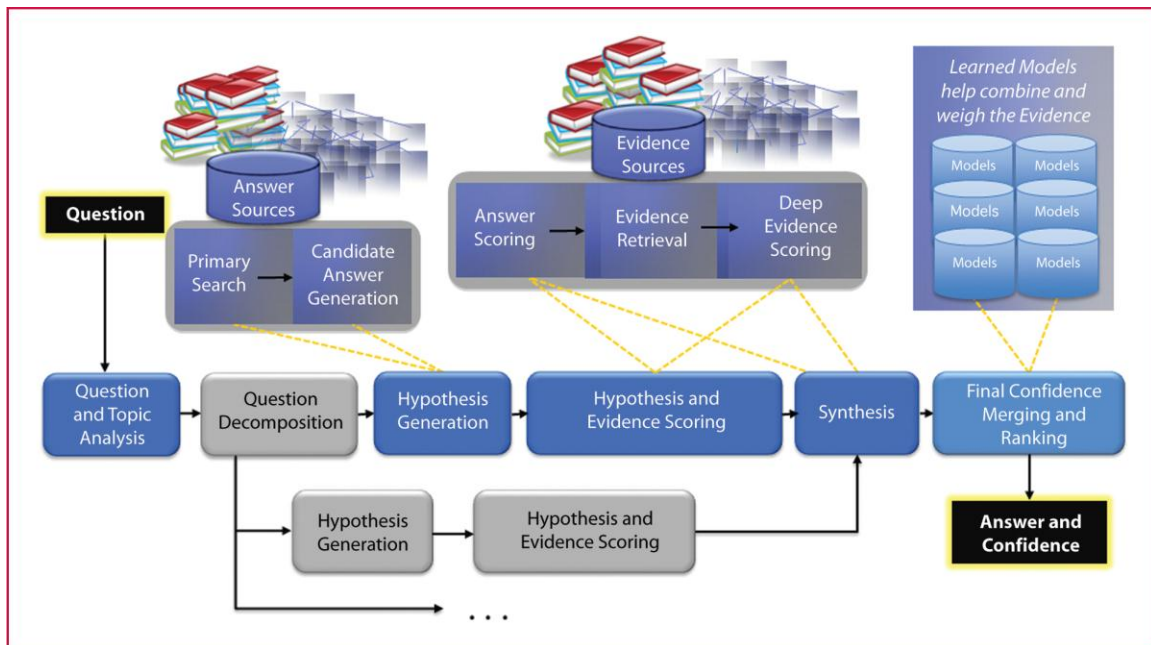
Figure 3.1: Architecture of Watson[8].

The pipeline has the following components:

1. Question and Topic Analysis.
2. Question Decomposition
3. Hypothesis generation
4. Hypothesis and evidence scoring
5. Synthesis
6. Final Ranking and Merging

---

[8] Image Reference: (Ferrucci, D. A., 2012)

It is to be mentioned that each of the steps has many alternatives which will be looking at details throughout the chapter.

2. Question analysis: How Watson reads a clue

A number of rules and classifiers are applied to the question to analyze critical elements of the questions (A. Lally, J. M. Prager, M. C. McCord, B. K. Boguraev,S. Patwardhan, J. Fan, P. Fodor, J. Chu-Carroll., 2012):

1. Part of question that is important reference for the getting the answer. We call such terms as "Question focus".
2. Terms in the question that specify what the question is looking for, which is called **Lexical Answer Type** or LAT.
3. A classification of Question in various categories of several types.
4. And elements of question which will require special handling.

Example 3.1[9]:

For the Jeopardy Question:

POETS & POETRY: He was a bank clerk in the Yukon before he published "Songs of a Sourdough" in 1907.

Question Focus: Question Focus is the reference to the answer in this case the question focus will be "he".

Lexical Answer Type: We are looking for a person in this question. The LAT for the question is "clerk", "poet" and "he".

---

[9]Most of the examples in this chapters have been replicated from the Watson papers. However there are certain cases where own examples have been provided without violating the context of discussion.

Question Classification: The question class here is factoid question, however there may be other categories like Definition, Multiple-Choice, Puzzle, Common Bonds, Fill-in-the-Blanks, and Abbreviation.

Special handling: Here the Question Section or QSection case is not prominent however another example when the QSection requires special handling is if question has section like "3 Letter word" or "4 letter Word".

2.1 Foundations of Question Analysis

The question analysis builds upon the parsing and the semantic analysis capabilities of IBM Watson. It is composed of English Slot Grammar Parser, ESG with an associated Predicate Argument Structure (PAS). The parser will be discussed in the later section, however to give what it could possibly do we provide a simple illustration.

Example 3.2:

POETS & POETRY: He was a bank clerk in the Yukon before he published "Songs of a Sourdough" in 1907. Would generate the following predicate argument structure:

publish (e1, he, "Songs of a Sourdough")
in (e2, e1, 1907)
andtemporalLink(publish(…), 1907).

2.2 Focus and LAT Detection

A baseline system was designed to detect the focus. The rules that were used to create the system are directly taken from the paper and presented here and the focus that system has detected has been presented below:

A noun phrase with determiner "this" or "these":

Example 3.3:

*THEATRE: A new play based on this **Sir Arthur Conan Doyle canine classic** opened on the London stage in 2007.*

"This" or "these" as a pronoun:

Example 3.4:

*'88: In April 1988, northwest became the first U.S. air carrier to ban **this** on all domestic flights.*

When the question is a noun phrase, we conventionally label the entire question as the focus:

Example 3.5:

*AMERICAN LIT: **Number of poems Emily Dickinson gave permission to publish during her lifetime.***

One of the pronouns "he/she/his/her/him/hers":

Example 3.6:

*OUT WEST: She joined Buffalo Bill Cody's Wild West Show after meeting **him** at the Cotton Expo in New Orleans.*

One of the pronouns "it/they/them/its/their":

Example 3.7:

*ME "FIRST"! **It** forbids Congress from interfering with a citizen's freedom of religion, speech, assembly, or petition.*

The pronoun 'one':

Example 3.8:

*12-LETTER WORDS: Leavenworth, established in 1895, is a federal **one**.*

When none of the above applies, the question may have no focus:

Example 3.9:

MOVIE TITLE PAIRS: 1999: Jodie Foster & Chow Yun-Fat.

The baseline system was further improved to correctly identify the question focus.

2.3   Extracting LATs from the category

The LAT were detected based on the previous questions seen by Watson. It was able to adjust its decision based on current quest and category.

Example 3.10:

In one of the games that Watson Played, for the category "CELEBRATIONS OF THE MONTH" which contained the following questions, it incorrectly identified the LAT as "day".

"D-Day Anniversary and Magna Carta Day"

"National Philanthropy Day and All Souls' Day"

"National Teacher Day and Kentucky Derby Day"

"Administrative Professionals Day and National CPAs Goof-Off Day"

"National Magic Day and Nevada Admission Day"

But later after getting to Know correct answer to first question:

"D-Day Anniversary and Magna Carta Day" as "June". It was able to adjust itself to "Month" as the Lexical Answer Type.

2.4   Question Classification and QSection issues

The question were classified into QClasses so that the later stages become easier for Watson. As we have discussed in Introduction Chapter that different classes of question require different type of handling. Watson analyzes the questions and finds the QClasses, and depending on the QClass it tries to answer the questions. Various categories were identified.

1. Definition

   These are the questions which contain definition of the answer.

2. Fill in the blanks

   These are the question in Jeopardy in which we are required to fill up the blanks.

3. Abbreviation

   The answer is expansion of the abbreviation in the question.

4. Category relation

   Answer has semantic relation to the question, where the category will specify the relation between them.

5. Puzzle

   Answers that requires inference, derivation and reasoning.

6. Verb

   Questions ask for a verb.

7. Translation

   A question which requires the translation of a word or phrase from other language.

8. Number

   Questions where the answer is a number.

9. Bond

   The question is to identify the bonds between the categories or words.

10. Multiple Choice

    We choose one answer out of multiple answers provided in the question.

11. Date

    Questions that ask for a date or time as the answer.

The detection in Watson is done mostly rule based, which includes regular expressions patterns to detect the QClass. However logistic classifier is employed to get the best possible class. To this classifier the binary features corresponding to firing of rules are fed and trained against previously annotated QClass as the label.

A special mention is the QSection, which is important in identifying the special handling of questions.

Example 3.11:

Question: Name a three letter word that is a color and ends with 'D'

It has a section "three letter word" which applies a constraint in answering the questions.

3. Deep parsing in Watson

The essential part of linguistic capabilities come from two components: English Slot Grammar Parser, ESG and the Predicate Argument Structure Builder, PAS Builder. ESG is a deep parser, which not only shows the shallow structure of the English Sentence but also the deeper logical analysis. To have an idea of where it is used we provide the following examples (M. C. McCord, J. W. Murdock, B. K. Boguraev., 2012):

1. Relation Extraction, where semantic relationship between the entities are extracted.
2. Analysis of Question requires the parser to get the indication of what the question is in quest of.
3. Component responsible for keyword extraction, which extracts the important keywords having stronger semantic relation to the question, uses the parsing capabilities of Watson.

4. Passage scoring which aligns the evidence in the passage with is dependent on Deep Parsing components of Watson.

5. Type Coercion system uses PAS to compare the demanded type to those found in the natural language text.

6. Relations extracted by PAS builder from large text corpora are accumulated in PRISMATIC knowledge base.

4. Textual resource acquisition and engineering

A little emphasis has been given to type of corpora required for hypothesizing answers to the questions. The text corpora needs to be of high quality and it must be highly relevant to the expected type of questions. Researchers at IBM developed three procedures to accomplish the task which have been listed below:

1. Source Acquisition,
2. Source Transformation
3. Source Expansion

Source Acquisition is a way of iteratively acquiring textual corpora that enhances the already existing resources (J. Chu-Carroll, J. Fan, N. Schlaefer, W. Zadrozny, 2012).

Source Transformation is procedure of representing acquired textual corpora into the form that can be easily used by the Watson System.

Source Expansion is final stage in resource acquisition and engineering, which deals with building coverage of Watson, incorporating the lexical and syntactic variations.

5. Automatic knowledge extraction from documents

Having more knowledge is always a critical theme in open domain question answering system. Knowledge represented formally is easy to reason with. However the extraction of structured knowledge from unstructured text data is difficult.

Information extracted is done in stages:

1. Shallow knowledge Extraction
2. Additional semantic inference

Example 3.12:

Shallow knowledge extraction

"Einstein, who has published more than 300 scientific papers, won the Nobel

Prize for Physics in 1921"

A Named entity recognizer can be used with a dependency parser to extract following patterns:

"Scientists publish papers"

 "Scientists win Nobel prizes"

 "Nobel prizes are associated with subjects and years"

Such patterns are extracted from the unstructured text. The point is that we can make a knowledge base, which can directly be used for answering questions.

6. Finding needles in the haystack: Search and candidate generation

One of the key steps in IBM Watson is hypothesis generation. The Hypothesis generation step is dependent on two steps:

1. Searching the available textual corpora
2. Candidate Generation

When a question is posed to Watson, the search component searches what is available to it. It fetches all the relevant content that will be required to answer the question.

Candidate generation component uses the above step and does further analysis on the retrieved text. Using various strategies, which include using the metadata of the retrieved content, like document title, anchor text for the hyperlink it is able to hypothesize the answers.

It was found that using the strategies Watson system was able to bring candidate answers, to be exact correct answer pool to 87.17% of the blind questions posed to the system (J. Chu-Carroll, J. Fan, B. K. Boguraev, D. Carmel, D. Sheinwald, C. Welty., 2012).

7. Typing candidate answers using type coercion

Many popular methods in question answering systems use the fact that answer type can be extracted from the question directly. And they tend to retrieve those answers which are of the expected type. However if the approach is applied to the Jeopardy Games, such approach would not work. IBM Watson doesn't make and hard answer type hypothesis just looking at the question.

To add to the complexity the predefined types cover only a small pool of answers. The alternative approach of type coercion, generates answers which are of different type, and a scoring is performed to get the best type of the answer.

Example 3.13:

Suppose a question wants the type as "musician" and the answer it expects is "AR Rahman".

The lexical answer type is "musician".

It will look for the sources like:

Dbpedia: It will try to find what dbpedia entry "AR Rahman" refers to and it will find that it is indeed musician.

It also looks for evidences from WorldNet and other ontologies.

It was found that using type coercion component in Watson increases the accuracy of Watson by 5% (J. W. Murdock, A. Kalyanpur, C. Welty, J. Fan, D. A. Ferrucci, D. C. Gondek, L. Zhang, H. Kanayama., 2012).

8. Textual evidence gathering and analysis

We not only want the answers to the questions but we want the system to provide confident answers. Once hypotheses has been generated by the system. The textual evidence gathering component starts executing in parallel. The textual evidence gathering system, uses a novel method called "Supporting evidence retrieval". The passages retrieved by the SER component will try to find close semantic relatedness to the passages. The confidence of the answers will be based on the amount of evidence gathered for each of the answers.

To perform the match the following scoring approaches are used:

1. Passage term match.
2. Skip bigram match.
3. Textual alignment.
4. Logical form Answer candidate scoring.

Example 3.14:

Question: Who is author of Mahabharata?

Evidence: The great saint Ved Vyas, is the author of epic Mahabharata.

*Let the **hypothesis** be: Ved Vyas*

The passage matching terms are:

"Mahabharata", "is", "author", "of".

*The skip bigram match may be: **Author** of **Mahabharata***

Textual alignment: "is author of Mahabharata" can be aligned to respective words in passage.

Logical form candidate scoring: The focus "who" matches with "Ved Vyas" the candidate answer.

The evidence gathering system has a big impact on the question answering system as it is now able to give confidence to the answers, which is useful in deciding whether to buzz or not during the Jeopardy games (J. W. Murdock, J. Fan, A. Lally, H. Shima, B. K. Boguraev., 2012).

9. Relation extraction and scoring in DeepQA

Detection of relations is not only an important area in question answering but it has an important mark in all of natural language processing and information retrieval.

For a question answering system detecting relations plays a central role in not only extracting answers but also in supporting evidences for the passages. There are two approaches by which relation are extracted in the Watson system:

1. Manual pattern specification
2. Statistical Pattern elicitation

Manual pattern specification is the rule based method of specifying patterns and is able to cover about small number of relations, approximately 30.

Statistical methods are complementary to the rule based approach and covers the larger set of approximately 7000 extracted relations. Even though the accuracy of

the statistical method is less accurate compared to the manual pattern specification, it has a huge impact in overall accuracy of the Watson system (C. Wang, A. Kalyanpur, J. Fan, B. K. Boguraev, D. C. Gondek., 2012).

10. Structured data and inference in DeepQA

As mentioned before most of the evidences in Watson system comes from unstructured text. But the Watson also has several components that deal with structured data. The structured data processor is not monolithic, instead it is used with various other components in Watson to perform reasoning.

Structured data complements the unstructured data in the Watson, however advantage of having structured data enables us to represent the data in formal specifications and perform reasoning and inference on it (A. Kalyanpur, B. K. Boguraev, S. Patwardhan, J. W. Murdock, A. Lally, C. Welty, J. M. Prager, B. Coppola, A. Fokoue-Nkoutche, L. Zhang, Y. Pan, Z. M. Qiu., 2012).

Areas where the structured data has most impact are:

1. Geospatial and temporal constraints.
2. Presidents and countries.

Example 3.15:

Question: *Who was president of India in 2011?*

Structured information:

Country: India

President: Pratibha Patil

Time frame: 2007 -2012

Can be used to answer the above question. We find that the temporal constraint 2011 is satisfied by 2007-2012. Which means the correct answer for the question is:

Answer: Pratibha Patil

11. Special Questions and techniques

This section is mostly Jeopardy specific. Standard questions in Jeopardy are providing or two assertions about an unnamed entity and our task is to identify the described entity (J. M. Prager, E. W. Brown, J. Chu-Carroll., 2012).

There are however small section of special questions, where it is required to derive answers from the assentation(s), or derive assentation from the provided assertion(s).Although the special category questions are not significant in broader domain question answering systems. But they form an important class to triumph at Jeopardy.

Sample special type questions include:

1. Puzzle
2. Multiple choice
3. Common Bond
4. Fill in the blanks

Example 3.16:

We describe what issues with answering multiple choice questions are.

In most of the question answering system we give a huge penalty to answers extracted that is present in the question. This will work fine for question like:

Question: What is capital of India?

We may extract answers from the text saying "India". Which in turn will be heavily penalized.

But if the question is:

Question: Who was the first prime minister of India, Jawaharlal Nehru or Lal Bahadur Shastri?

We cannot penalize the answer extracted answer Jawaharlal Nehru for being the correct answer.

Solution to the above problem is a multiple choice question classifier. However the question patterns are manual specified. The solution provided is essentially rule based.

12. Identifying implicit relationships

Normal questions tend to seek for an entity which is an answer to the question. However there are cases where an explicit question is posed to find relation between entities. To identify the answer to the explicit question and to find the implicit relation between the entities, we have to discover the hidden concepts in the question (J. Chu-Carroll, E. W. Brown, A. Lally, J. W. Murdock., 2012).

The section can be termed as Jeopardy specific solution to answer the "common bonds" questions.

A strategy that is applied to answer such questions is concept expansion. We take the entities in the question and expand the concept and try to find if they are related.

Example 3.17:

COMMON BONDS: lingual, your legs, validation

One of the way of expanding the concept is:

1. *Cross lingual*

2. *Cross your legs*

3. *Cross validation*

So we can say the common bond is "cross".

13. Fact-based question decomposition in DeepQA

Factoids questions are what question answering systems are good at. Most of the factoids questions posed to the system are simple. However there are cases where the factoid question may be complex. Besides Watson the other question answering systems typically do not do a sufficient analysis on such complex questions (A. Kalyanpur, S. Patwardhan, B. K. Boguraev, A. Lally, J. Chu-Carroll, 2012).

According to IBM Watson the complex factoid questions can be classified as:

1. Nested Factoid Questions
2. Parallel decomposable questions

Parallel questions contain sub questions that can be analyzed independent of each other.

Nested Questions on the other hand are to be processed in sequence, the inner question is processed first and it is plugged into the outer question.

Example 3.18:

Question: *This person who was born in 1869 is also known as father of the nation.*

This question contains two facts:

1. *This person was born in 1869*

2. *He is known as father of the nation*

The two components can be analyzed independently and we can find that both provide "Mahatma Gandhi" as answer.

So the question answering system will be confident to answer that answer is "Mahatma Gandhi".

14. A framework for merging and ranking of answers in DeepQA

The final stage in IBM Watson pipeline is the merging of answers. The DeepQA uses phase based framework in which machine learning techniques are applied in succession[10]. There are some cases where the candidate answers are equivalent or closely related, so the evidence for one answer will be relevant to the other (D. C. Gondek, A. Lally, A. Kalyanpur, J. W. Murdock, P. A. Duboue, L. Zhang, Y. Pan, Z. M. Qiu,C. Welty., 2012).

We will discuss the **answer merging** in this section. The following example illustrate the working of answer merging.

Example 3.19:

There are multiple ways of referring to "John F Kennedy".

The candidate generation system may hypothesize the following answers:

1. *JFK*
2. *Kennedy*
3. *J F Kennedy*
4. *John F Kennedy*

Handling the entities separately means we are making discriminations in potentially equivalent answers. This will be confusing for the ranking system. However if we can merge the mentioned entities into a canonical form we can get a more robust system.

---

[10] Please refer to A framework for merging and ranking of answers in DeepQA for more details on how machine learning is used in DeepQA

15. Making Watson fast

IBM Watson was designed to showcase the DeepQA techniques in the Jeopardy game show. So the system designed was expected to have low latency. Hence massive parallel system was built which was based on UIMA[11] architecture (E. A. Epstein, M. I. Scho,B. S. Iyer, A. Lally, E. W. Brown, J. Cwiklik., 2012). The important mentioning components in UIMA are:

1. Common analysis structure(CAS)
2. CAS Multiplier
3. Annotator
4. Flow Controller
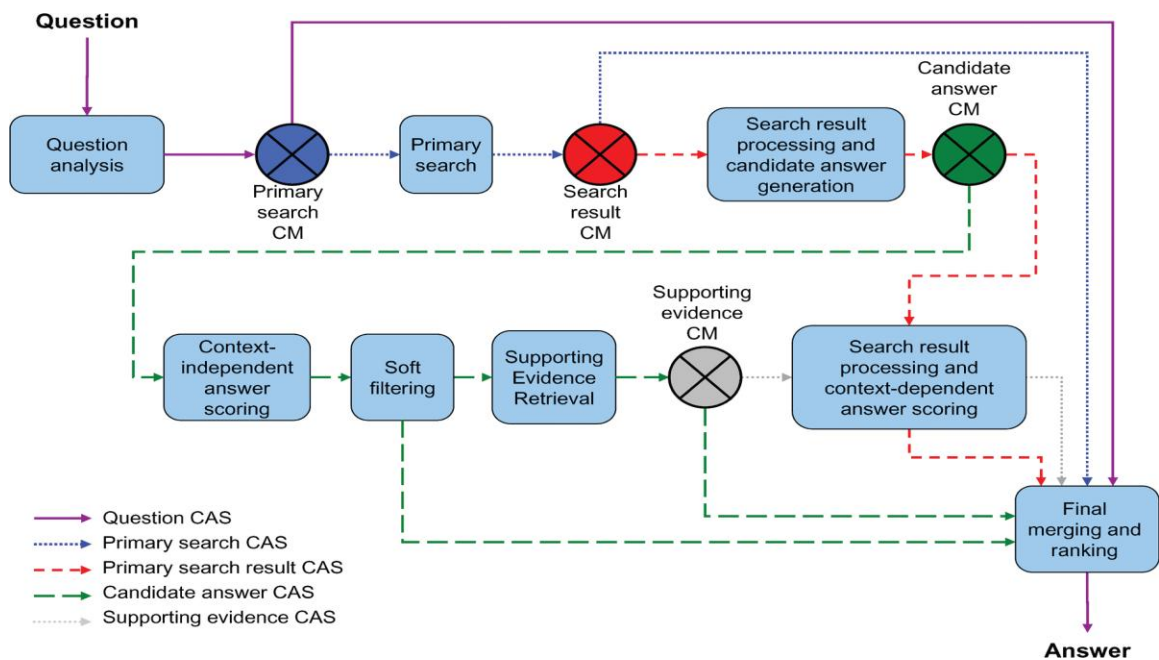
Figure[12] 3.2: The DeepQA in UIMA.

CAS is a unit of work that is transmitted from one component to another. Annotators are the individual processing components. CAS multiplier feeds in a CAS and provides multiple children of parent type. Flow controller determines how CASes move along the UIMA components.

IBM Watson is written in JAVA and C++, which make up the 100 analytical components and about 1 million lines of code. Each of the algorithms implemented were fully optimized to be at par with latency requirements. The data required to answer a question was directly put into RAM owing to glitches in Disk IO.

In addition to software architecture, the hardware was massive. 400 processes were deployed across 72 IBM POWER 750 machines.

16. In the game: The interface between Watson and Jeopardy

The IBM Watson has to play games with the human counterparts. It has to operate the following components:

1. Question Answering
2. Game Strategy
3. Speech
4. Buzzer

Watson is machine and hence cannot see or hear the clues given to the contestants during the Jeopardy show. Hence the clues were sent electronically to the Watson. Watson monitors the buzzer system to find whether it can buzz to give the correct answer (B. L. Lewis, 2012).

When Watson gets the electronic clue, it analyzed the clue to find answers. If Watson is confident about the answer it will activate the solenoid that will press the

buzzer. The speech component will convert the text answer to speech that will be presented to the audience. Since it cannot hear or see the judgment made. It relied on the change in the scores to check whether the answer provided was indeed correct.
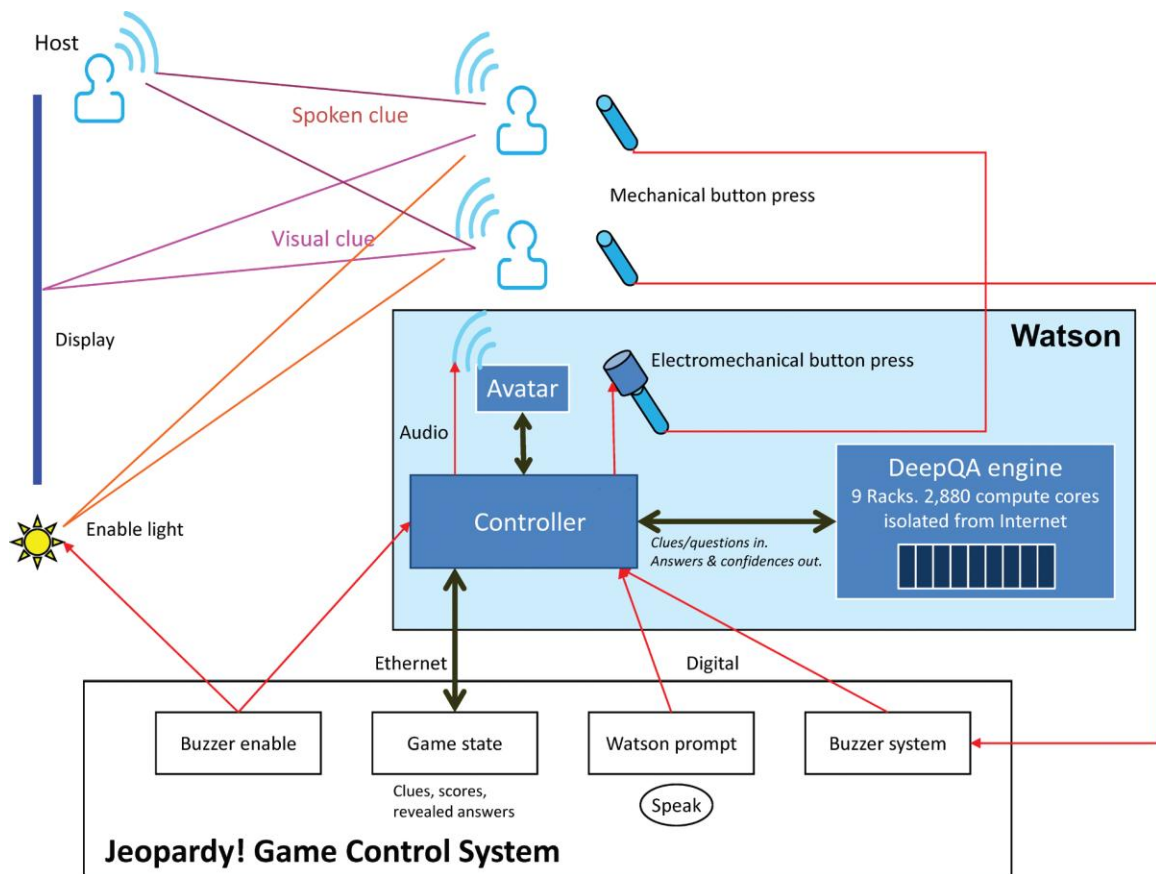

Figure 3.3: Pictorial Representation of IBM Watson interface[13].

17. Watson: Beyond jeopardy

Watson proved its ability in question answering at jeopardy, but the success of DeepQA can be extended to other domains like healthcare and education. The

[13] Image Reference: (B. L. Lewis, 2012)

incorporation of NLP Techniques in DeepQA makes it capable to understand unstructured text. In this section we discuss how Watson can be used in Heath Care domain.

Creating a complete rule based expert system is difficult in the health care domain. It is not only costly to build such a system but also brittle. The rules must match accurately to the input data. DeepQA on the other hand can be directly applied to readily available text in natural language and queries can be used to get relevant information.

In clinical setting it can be used to develop an assistant to the physician, where an input would be provided to the system consisting of the cases and patient's medical condition. DeepQA engine would then do all the reasoning and provide assistance to doctor regarding the appropriate actions to be performed. The questions that is required to be answered is almost jeopardy like for medical domain.

Example 3.20:

Question: The syndrome characterized by joint pain, abdominal pain, palpable purpura, and a nephritic sediment.
Answer: Henoch-Schonlein Purpura.

Question: Familial adenomatous polyposis is caused by mutations of this gene.
Answer: APC Gene.

Question: The syndrome characterized by narrowing of the extra-hepatic bile duct from mechanical compression by a gallstone impacted in the cystic duct.
Answer: Mirizzi's Syndrome.

With few adaptation to the DeepQA engine that worked at Jeopardy we can have a robust system for medical diagnosis (David Ferrucci, Anthony Levas, Suato Bagchi, David Gondek nad Eric Mueller, 2011).

18. Summary

The introductory section covers the brief history of Watson and the pipeline of IBM Watson Architecture. Then we move on to papers published by IBM Watson. We follow the roadmap presented in the beginning of the chapter and end the discussion by Watson beyond Jeopardy, where Watson is used in medical domain.

References

A. Kalyanpur, B. K. Boguraev, S. Patwardhan, J. W. Murdock, A. Lally, C. Welty, J. M. Prager, B. Coppola, A. Fokoue-Nkoutche, L. Zhang, Y. Pan, Z. M. Qiu., 2012. Structured data and inference in DeepQA. IBM J. RES. & DEV. VOL. 56 NO. 3/4, pp. 1-14.

A. Kalyanpur, S. Patwardhan, B. K. Boguraev, A. Lally, J. Chu-Carroll, 2012. Fact-based question decomposition in DeepQA. IBM J. RES. & DEV. VOL. 56 NO. 3/4, pp. 1-11.

A. Lally, J. M. Prager, M. C. McCord, B. K. Boguraev,S. Patwardhan, J. Fan, P. Fodor, J. Chu-Carroll., 2012. Question analysis: How Watson Reads a clue. IBM J. RES. & DEV. VOL. 56 NO. 3/4, pp. 1-14.

B. L. Lewis, 2012. In the game: The interface between Watson and Jeopardy!. IBM J. RES. & DEV. VOL. 56 NO. 3/4, pp. 1-6.

Benjamin S. Bloom, 1984. Taxonomy of educational objectives. s.l., s.n.

C. Wang, A. Kalyanpur, J. Fan, B. K. Boguraev, D. C. Gondek., 2012. Relation extraction and scoring in DeepQA. IBM J. RES. & DEV. VOL. 56 NO. 3/4, pp. 1-12.

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 J. RES. & DEV. VOL. 56 NO. 3/4, pp. 1-12.

David Ferrucci, Anthony Levas, Suato Bagchi, David Gondek nad Eric Mueller, 2011. Watoson: Beyond Jeopardy, s.l.: IBM Research Report.

E. A. Epstein, M. I. Scho,B. S. Iyer, A. Lally, E. W. Brown, J. Cwiklik., 2012. Making Watson fast. IBM J. RES. & DEV. VOL. 56 NO. 3/4, pp. 1-12.

Ferrucci, D. A., 2012. Introduction to "This is Watson". IBM J. RES. & DEV. VOL. 56 NO. 3/4, pp. 1-15.

J. Chu-Carroll, E. W. Brown, A. Lally, J. W. Murdock., 2012. Identifying implicit relationships. IBM J. RES. & DEV. VOL. 56 NO. 3/4, pp. 1-10.

J. Chu-Carroll, J. Fan, B. K. Boguraev, D. Carmel, D. Sheinwald, C. Welty., 2012. Finding needles in the haystack: Search and candidate generation.. IBM J. RES. & DEV. VOL. 56 NO. 3/4, pp. 1-12.

J. Chu-Carroll, J. Fan, N. Schlaefer, W. Zadrozny, 2012. Textual resource acquisition and engineering. IBM J. RES. & DEV. VOL. 56 NO. 3/4, pp. 1-12.

J. Fan, A. Kalyanpur, D. C. Gondek, D. A. Ferrucci., 2012. Automatic knowledge extraction from documents. IBM J. RES. & DEV. VOL. 56 NO. 3/4, pp. 1-10.

J. M. Prager, E. W. Brown, J. Chu-Carroll., 2012. Special Questions and techniques. IBM J. RES. & DEV. VOL. 56 NO. 3/4, pp. 1-13.

J. W. Murdock, A. Kalyanpur, C. Welty, J. Fan, D. A. Ferrucci, D. C. Gondek, L. Zhang, H. Kanayama., 2012. Typing candidate answers using type coercion. IBM J. RES. & DEV. VOL. 56 NO. 3/4, pp. 1-13.

J. W. Murdock, J. Fan, A. Lally, H. Shima, B. K. Boguraev., 2012. Textual evidence gathering and analysis. IBM J. RES. & DEV. VOL. 56 NO. 3/4, pp. 1-14.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning, 2005. Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. s.l., s.n.

M. C. McCord, J. W. Murdock, B. K. Boguraev., 2012. Deep parsing in Watson. IBM J. RES. & DEV. VOL. 56 NO. 3/4, pp. 1-15.

SATOSHI SEKINE AND RALPH GRISHMAN, 2003. Hindi-English Cross-Lingual Question-Answering System. s.l., s.n.

Stefan Wemter and Wendy Lehnert, 1990. A survey of Question Answering in Natural Language. B.V. (North-Holland), s.n.

Tomek Strzalkowski, Sanda Harabagiu, 2010. Advances in Open Domain Question Answering. P.O. Box 17, 3300 AA Dordrecht, The Netherlands: Springer.

Baker, Collin F., Charles J. Fillmore, and John B. Lowe. "The berkeley framenet project." Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 1. Association for Computational Linguistics, 1998.

Bird, Steven. "NLTK: the natural language toolkit." Proceedings of the COLING/ACL on Interactive presentation sessions. Association for Computational Linguistics, 2006.

Cortes, Corinna, and Vladimir Vapnik. "Support vector machine." Machine learning 20.3 (1995): 273-297.

Carlson, Andrew, et al. "Toward an Architecture for Never-Ending Language Learning." AAAI. 2010.

De Marneffe, Marie-Catherine, Bill MacCartney, and Christopher D. Manning. "Generating typed dependency parses from phrase structure parses." Proceedings of LREC. Vol. 6. 2006.

De Smedt, T., & Daelemans W. (2012). Pattern for Python. Journal of Machine Learning Research. 13, 2063-2067.

Harabagiu, Sanda M., Marius A. Paşca, and Steven J. Maiorano. "Experiments with open-domain textual question answering." Proceedings of the 18th conference on Computational linguistics-Volume 1. Association for Computational Linguistics, 2000.

Hovy, E. H., Hermjakob, U., & Lin, C. Y. (2001, November). The Use of External Knowledge of Factoid QA. In TREC (Vol. 2001, pp. 644-52).

Huang, Zhiheng, Marcus Thint, and Zengchang Qin. "Question classification using head words and their hypernyms." Proceedings of the Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2008.

Joachims, Thorsten. "Making large scale SVM learning practical." (1999).

Krishnan, Vijay, Sujatha Das, and Soumen Chakrabarti. "Enhanced answer type inference from questions using sequential models." Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2005.

Levy, Roger, and Galen Andrew. "Tregex and Tsurgeon: tools for querying and manipulating tree data structures." Proceedings of the fifth international conference on Language Resources and Evaluation. 2006.

Li, Xin, and Dan Roth. "Learning question classifiers." Proceedings of the 19th international conference on Computational linguistics-Volume 1. Association for Computational Linguistics, 2002.

Miller, George A. "WordNet: a lexical database for English." Communications of the ACM 38.11 (1995): 39-41.

Moschitti, Alessandro. "Efficient convolution kernels for dependency and constituent syntactic trees." Machine Learning: ECML 2006. Springer Berlin Heidelberg, 2006. 318-329.

Moschitti, Alessandro. "Making Tree Kernels practical for Natural Language Learning."

Nguyen, Dat PT, Yutaka Matsuo, and Mitsuru Ishizuka. "Relation extraction from wikipedia using subtree mining." Proceedings of the National Conference on Artificial Intelligence. Vol. 22. No. 2. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2007.

Schlaefer, Nico, et al. "A pattern learning approach to question answering within the Ephyra framework." Text, Speech and Dialogue. Springer Berlin Heidelberg, 2006.

Vapnik, V. "The Nature of 6tatistical Learning Theory." Data Mining and Knowledge Discovery (6): 1-47.

Zhang, Dell, and Wee Sun Lee. "Question classification using support vector machines." Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval. ACM, 2003.