181345
Q4

Question answering systems focus on factoid question answering which are question that can be answered with simple facts, often returned in short texts. When building such systems, there are two major standards which could be implemented:

- IR-based question answering
- Knowledge-based question answering

This essay will discuss the two different paradigms, evaluate which one would be best suitable when building a system to answer questions about historical events given the access to a database, and discuss the challenges and strategies the system implemented would have.

An **IR-based question answering** uses vast quantities of textual information from the web or collection, given a user question to find relevant documents and passages using information retrieval techniques. Then, the systems (feature-based, neural, or both) will use reading comprehension algorithms to evaluate and read these retrieved documents/passages to provide an answer directly from a range of text.

On the contrary, a **Knowledge-based question answering** system will build a semantic representation of the query such as time,dates,locations,entities,numeric quantities. For instance for the given question "What is the capital of France?", the system will provide a logical representation of the queries: $\lambda x.capital(x) \wedge what(x,France)$. These meanings are then used to query database of facts.

Given access to a database consisting of English-language newspaper and magazine articles for the last 50 years, it would be suitable to build a question-answering system based on a knowledge-based question answering approach. As we know that the system will be answering question pertaining to historical events, the information retrieval based method will be able to acquire the answers from the collection of documents. The system is comprised of three phases represented in ***figure 23.2***[1]:

- Question Processing
- Document and Passage Retrieval
- Answer Extraction

[1] "Question Answering - Stanford University." 23 Sep. 2018, https://web.stanford.edu/~jurafsky/slp3/23.pdf. Accessed 14 Jan. 2019.

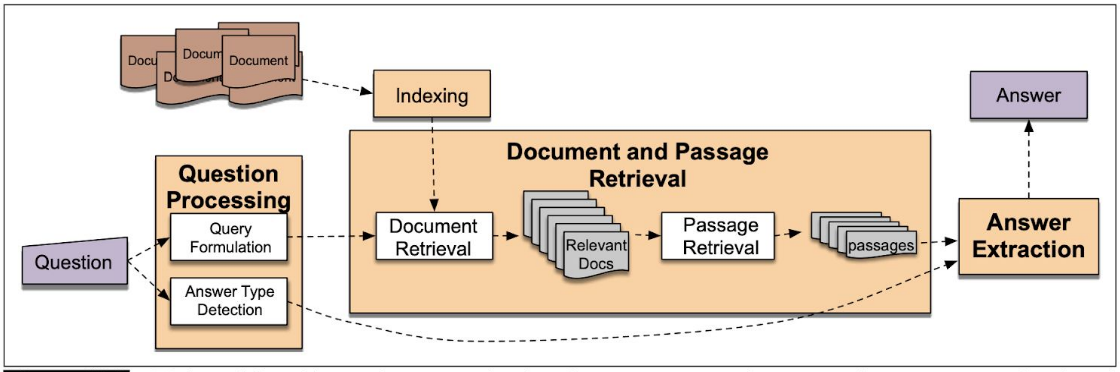


Figure 23.2 IR-based factoid question answering has three stages: question processing, passage retrieval, and answer processing.

Question Processing

The main goal of question processing is to understand what the question asked is by extracting the query. There are many as 5 elements that are usually extracted from the question depending on its nature:

1. Answer Type Detection
   a. This represents a named entity type of the answer (person,place,date) which defines what is looked for in the factoid question
2. Query Formulation
   a. This represents the set of words which will be sent to the IR engine to determine what to look for and passages are most likely containing the answer.
3. Question Type classification
   a. The definition and nature of the question could be extracted. For instance, if a question pertains to a definition, the answer will most likely exists reside within a dictionary or a text comprised of word definitions
4. Focus Detection
   a. This finds the question words that are replaced by the answer
5. Relation Extraction
   a. Find relations between entities in the question.

For instance, for this given sentence, "Where was the football World Cup held in 2002 ?" The system will extract the following:

- Answer Type: Place
- Query: football, World Cup, 2002
- Focus: World Cup held in 2002
- Relations: World Cup(where,?x,2002)

Within the Relations portion, the 'x' represents the unknown answer which is in the World Cup relation pertaining to specifically a location (where) and a specific year (2002). Hence, magazines or newspapers within the databases expressing sports events specifically the World Cup in 2002, would be searched.
In addition, the answer to a question might appear in many different forms, one of which might satisfy a match within a smaller document set. To account for this, different morphological variants of the content words could be added in hopes of increasing the probability of the answer appearing. This is one common query expansion method used.

Furthermore, when implementing a query formulation approach for questioning the web, query reformulation rules are applied. The rule rephrases the question to a different variant substring of possible declarative answers. For instance, the question *"When did princess Diana die"* might be reformulated as *"princess Diana died"*.

There are three main methods of approaching **Answer Type Detection** existing in modern systems:

- Hand-written rules
- Machine learning
- Hybrids

For the first part of Answer Type Detection, some systems may use question classification which is responsible for finding the answer type. Thus, named-entities the entities are considered: A question like *"<u>Who</u> was the leader of the National Union of Miners during the Miners strikes of the 1980's ?"* would expect an answer type PERSON. Once we know the answer type of the question, we are able to focus on sentences which correspond to the specific named-entity, instead of examining every sentence in the document collection.
Furthermore, while using named-entities could provide a direction as to what the answer type of the question is, we can also use larger hierarchical set of answer types called answer type taxonomy to improve this. Such taxonomies could be hand-built oe automatically generated from resources liek WordNet. One commonly used Answer Type

Taxonomy is the 6 coarse classes **Li and Roth (2005)**[2] which is comprised of (Abbreviation, Entity, Description, Human, Location, Numeric). In addition, within this tagset, 50 finer classes could be labeled with each 6 coarse classes. For instance a Location can be associated with a city,country and mountain, the former being one of the six coarse classes and the latter being the fine gained tag.

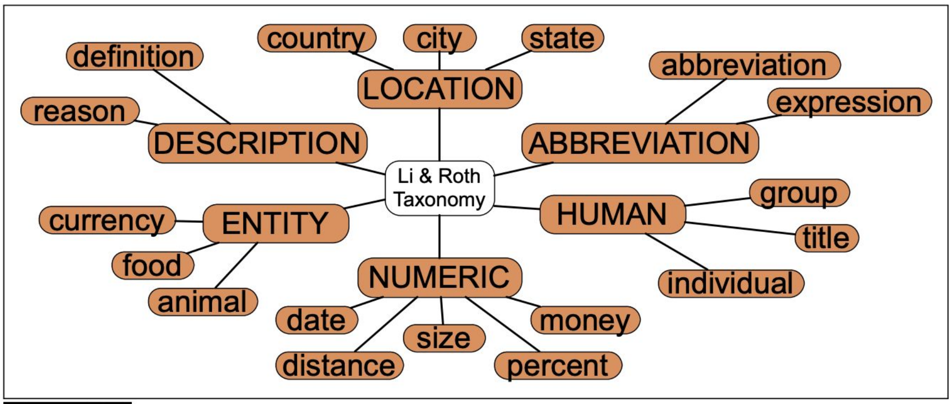


Figure 23.3 A subset of the Li and Roth (2005) answer types.

The figure above subset of the Li and Roth (2005) answer types. [3]

However, most question classifiers are implemented with supervised learning, trained on databases of questions that have been hand-labeled with an answer type. (Li and Roth, 2002). Either feature-based or neural methods can be used.

The feature based method relies on words in the questions and their embeddings, the PoS of each word and the named entities in the questions. It is often that a single word in the question provide additional information about the correct answer type. This word is often referred to as the question head word and maybe defined a the headword of the first NP (noun phrase) after the question's wh-word; headwords are indicated in boldface in the following examples:

- Who was the **leader** of the National Union of Miners during the miners' strikes of the 1980s?
- Where was the football **World Cup** held in 2002?
- When did **Princess Diana** die?

In general, question classification accuracies are relatively high on easy question types like PERSON, LOCATION, and TIME questions; detecting REASON and DESCRIPTION questions can be much harder.

---

[2] "Question Answering - Stanford University." 23 Sep. 2018, https://web.stanford.edu/~jurafsky/slp3/23.pdf. Accessed 14 Jan. 2019.
[3] "Question Answering Using Deep Learning - CS224d." https://cs224d.stanford.edu/reports/StrohMathur.pdf. Accessed 14 Jan. 2019.

Once the system has passed the **Answer Type Detection** it will proceed to **Query Formulation** which has the goal to create a query that would be sent to an information retrieval engine to retrieve documents that might contain the correct strings. If we consider how the system would be built to provide correct answers for the given question, we might consider the **Keyword selection algorithm** from *Moldovan et Al. (1999)*[4] which utilizes a number of heuristics each of which would indicate which keywords in the question might be important to insert in the query. The Sections of the algorithm could be devised into these following steps:[5]

1. Select all non-stop words in quotations
2. Select all NNP words in recognized named entities
3. Select all complex nominals with their adjectival modifiers
4. Select all other complex nominals
5. Select all nouns with their adjectival modifiers
6. Select all other nouns
7. Select all verbs
8. Select all adverbs
9. Select the QFW word (skipped in all previous steps)
10. Select all other words

The ranking of importance is in descending order. For instance, when choosing the keywords from the query given the question

"Where was the football World Cup held in 2002?"

We might throw out the stop words (highlighted in a lighter colour):

"Where was the football World Cup held in 2002?"

Now, *football,World* and *Cup* can be ranked [2] as nominals are crucial. *2002* could be ranked as [2] as it is a named entity providing a specific YEAR.

"Where was the football[2] World[2] Cup[2] held in 2002[2]?"

[4] "Question Answering Using Deep Learning - CS224d." https://cs224d.stanford.edu/reports/StrohMathur.pdf. Accessed 14 Jan. 2019.
[5] "Question Answering Using Deep Learning - CS224d." https://cs224d.stanford.edu/reports/StrohMathur.pdf. Accessed 14 Jan. 2019.

Furthemore, we would give *held* a rank[7] as it is a verb.

"Where was the football[2] World[2] Cup[2] held[7] in 2002[2]?"

Once we have went through the algorithm, we might provide the IR engine to extract from this that:

football/2 World/2 Cup/2 2002/2 held/7

There are two 4 terms that are most likely to be in the query and an element that is less likely to appear. Furthermore, we would be able to send a query in various ways:
We might send couple of elements with rank 2 and that does not return the correct query, we might send the entire array of elements until the system has returned the correct one.

Document and Passage Retrieval

Once the question has been processed and figured out which queries to send to the IR engine, Document retrieval is the next step to the system that is being implemented.

Passage retrieval can be divided into 3 steps:

1. The information retrieval engine acquires documents using the query terms.
    a. Once the IR query produced from the question processing stage is sent to an IR engine, it would result a set of document that are ranked by their relevance to the given query.
2. The documents are segmented into shorter units such as paragraphs.
    a. The most straightforward method of passage retrieval is to pass every document and passages to the answer extraction stage.
    b. However, a more smooth variant of the method mentioned in (a) is to apply a named entity or answer type classification filtering on the retrieved passages. Hence, the passages that don't contain the answer type that did not apply to the question are discarded.
    c. An alternative would be also to apply a supervised learning method to fully rank the remaining pass using features such as:
        i. The number of **named entities** of the right type in the passage.
        ii. The number of **question keywords** in the passage.

iii. The longest exact sequence of question keywords that occurs in the passage.
iv. The rank of the document from which the passage was extracted.
v. The **proximity** of the document from which the passage was extracted.(Pasca 2003, Monz 2004).
vi. The number of ***n*-grams** that **overlap** between passage and the question.(Brill et al.,2002).

3. The passages obtained are now ranked using answer type, which will be discussed further.

Either in rule-based classifiers or with supervised machine learning, these following features are used for Passage Ranking: [6]

- Number of **Named Entities** of the right type in passage
    - We might ask how many named-entities of the answer type occur in the passage. For instance, if the question pertains to a type PERSON and DATE but the passage does not contain these types then the passage is ranked as unlikely.
- Number of **query** words in passage
    - We might also consider how many of the query words exist in the passage. For instance, we might know the query words have occured in the document but we would also want to know how many times it has occured in a specific passage.
- Number of question N-grams also in passage
    - Instead of words, we might consider entire N-grams.
- Proximity of query keywords to each other in passage.
    - We might also consider how close these query keywords occur to each other in the passage. For instance, if there are three keywords which occur next to each other, we might want to consider the possibility that they pertain to the same meaning.
- Longest sequence of question words
    - We could also consider the length of the sequence of words.
- Rank of the document containing passage
    - The rank of the document itself could be a useful feature.

Now once we have retrieved ranked a number of passages, we can proceed to the last step of the Q/A system which is **Answer Processing.**

[6] "Question Answering Using Deep Learning - CS224d." https://cs224d.stanford.edu/reports/StrohMathur.pdf. Accessed 14 Jan. 2019.

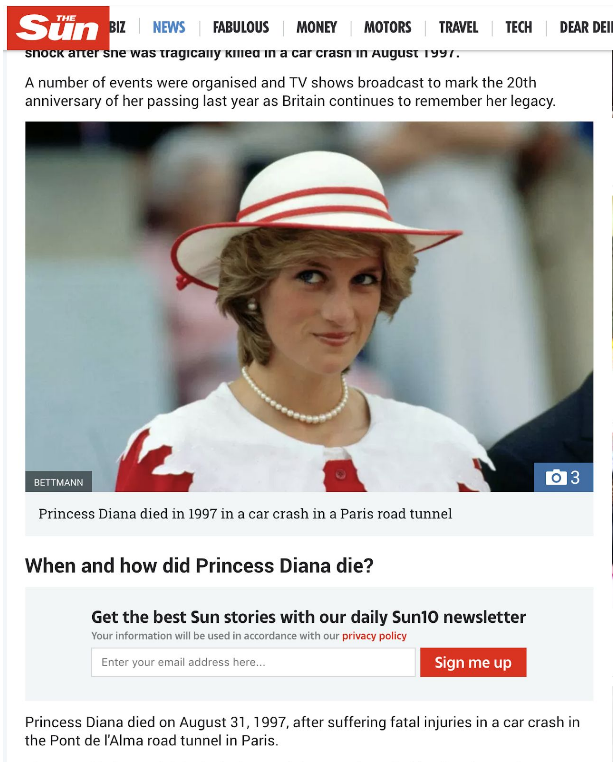
Answer Processing

The final stage of question answering is to correctly extract a specific answer from the passage. To do so, we will first run an answer-type named entity tagger on the passages. Each answer type requires a named-entity tagger that detects that specific answer type, for instance, if we know that the answer type is PERSON, we must have a tagger that would tag to PERSON also. This could be executed with a full named-entity tagger, simple regular expressions, or hybrid methods.
Furthermore, this would return the string with right type:

When did Princess Diana die? (PERSON-DATE)

For instance, given the sentence above being a PERSON-DATE type question and a news article below, we would want to know that **Princess Diana** is a person and **August 31, 1997** is the date which would likely be the answer.

"**Princess Diana** died on **August 31, 1997,** after suffering fatal injuries in a car crash in the Pont de l'Alma road tunnel in Paris."[7]



---

[7] "When did Princess Diana die, where did the Paris car crash ... - The Sun." 12 Sep. 2018, https://www.thesun.co.uk/news/3678292/princess-diana-car-crash-age-death-paris-prince-philip-letters/. Accessed 14 Jan. 2019.

Challenges and Solutions

Challenges

Now the issue that might be present is when a passage contains multiple **candidate answers** of the correct entity type. For instance for the question:

Who was the leader of the National Union of Miners during the miners' strikes of the 1980s?

And passage from Wikipedia:[8]

Article Talk Read Edit View history Search Wik

UK miners' strike (1984–85)

From Wikipedia, the free encyclopedia

The **miners' strike of 1984–85** was a major industrial action to shut down the British coal industry in an attempt to prevent colliery closures. It was led by Arthur Scargill of the National Union of Mineworkers (NUM) against the National Coal Board (NCB), a government agency. Opposition to the strike was led by the Conservative government of Prime Minister Margaret Thatcher, who called the union leaders as the "the enemy within", strikers and organisers misinterpreted the quote to suggest that Thatcher was referring to all miners.[1]

The **miners' strike of 1984–85** was a major industrial action to shut down the British coal industry in an attempt to prevent colliery closures. It was led by **Arthur Scargill** of the National Union of Mineworkers (NUM) against the National Coal Board (NCB), a government agency. Opposition to the strike was led by the Conservative government of Prime Minister **Margaret Thatcher**, who called the union leaders as the "the enemy within", strikers and organisers misinterpreted the quote to suggest that Thatcher was referring to all miners.[1]

The question asks who was the leader of the National Union of Miners, hence we know we are looking for a person. However, the passage above contains multiple PERSON type entities which may cause issues when extracting the answer from the passage.

The answer to the question is Arthur Scargill as it is stated that the National Union of Mineworkers was "was led by" him. However, to decide which of the named-entities is the correct answer is one of the machine learning problems that requires multiple features to extract the correct one.

[8] "UK miners' strike (1984–85) - Wikipedia." https://en.wikipedia.org/wiki/UK_miners%27_strike_(1984%E2%80%9385). Accessed 14 Jan. 2019.

Solutions

In order to minimize the challenge of machine learning, we would apply different features for ranking candidate answers:

Answer type match: True if the candidate answer contains a phrase with the correct answer type.
Pattern match: The identity of a pattern that matches the candidate answer.
Number of matched question keywords: How many question keywords are contained in the candidate answer.
Keyword distance: The distance between the candidate answer and query keywords.
Novelty factor: True if at least one word in the candidate answer is novel, that is, not in the query.
Apposition features: True if the candidate answer is an appositive to a phrase containing many question terms. Can be approximated by the number of question terms separated from the candidate answer through at most three words and one comma *(Pasca, 2003).*
Punctuation location: True if the candidate answer is immediately followed by a comma, period, quotation marks, semicolon, or exclamation mark.
Sequences of question terms: The length of the longest sequence of question terms that occurs in the candidate answer.

Evaluation

Furthermore, once we have picked an answer we must evaluate how accurate it is. If the system is returning one answer, we may calculate the **accuracy** to consider if the answer matches a gold-labeled answer for that question. However, with the case of returning multiple answers, we would use the **mean reciprocal rank**. For each query, we will return a ranked list of M candidate answers. The score of this query is 1/Rank of the first correct answer and 0 if there are no correct answers. For instance, if M is 5 giving 5 answers and only the third one is the first one that is correct then the score of that specific query is ⅓. Furthermore, we will take the mean of those ranks over all N queries. This is represented by the MRR equation below:

$$MRR = \frac{\sum_{n}^{i=1} \frac{1}{rank_i}}{N}$$