Report on Question Answering

General Introduction:

The First Paper tells us the specific learning surface text pattern for a whole question answering system. The Second paper is a kind of a survey for all kind of Question Answering Systems on Semantic web. Basically the difference between the First and the Second paper is First paper describes specifically about one aspect of Question Answering System but the Second paper gives us the survey result done for various Question Answering System on full Semantic Web.

Paper 1: Learning Surface text patterns for a Question Answering system

Introduction

This paper explores the strength of surface text patterns for open domain question answering systems. The popularity and effectiveness of the surface text patterns was shown at TREC-10 QA evaluation where the winning system used a list of extensive surface text patterns only. After reading the paper I explored that there are specific patterns of answers to certain questions. For example, lets take the question "When Bill Gates is graduating?", then the expected answer is "Bill Gates will graduate in 2015 or Bill Gates(2013-2015)". Now if the word "Bill Gates" is tagged as <NAME> and "2015" is tagged as <GRADUATIONYEAR>, then we get the various types of regular expression pattern to find the correct answer. Now we will learn those methods which automatically using machine learning to evaluate the patterns.

Explanation

The process of learning surface patterns can be roughly divided into 4 phases, which takes places as per the steps given below

a) The search result will be downloaded
b) Once the result will be downloaded then the extraction and standardization of patterns has to be done.
c) After extracting the patterns the precision calculation has to be performed.
d) Now the application of patterns has to be implemented on new questions.

Now we see the broader view of the above listed items

a) Downloading the Search Result:

Through this process we gathers the sentences which required to construct the corpus of tagged regular expression for each question which will going to be used to find answer in the later stage. To accomplish the downloading search result task

First the question type has to be chosen, lets say our question type is "GRADUATIONYEAR".
Second Now we have to decide on the question for question type "GRADUATIONYEAR", Lets say for "GRADUATIONYEAR" question is "BILL GATES" and Answer is "2015"
Third In this step whatever question is decided that will be given to any search engine like Bing.
Fourth In this step download the top 1000 documents and tokenizing of the document contents has to be done on the basis of the sentence.
Fifth Now in the last step we eliminate all the unnecessary tags like html tags and just consider the sentences, which contains both question and answer.

b) Standardization and Extraction of Patterns

After collecting all the appropriate sentences from the above steps, the next following steps will be used to create the tagged corpus.

First: The sentences, which we got after processing the download search results, will be passed to a suffix tree constructor.
Second: In this step the task is to find the longest matching substrings. Here the substring refers to the common part, which is present in both question and answer sentence. For example, in "Bill Gates (2013-2015) is a very innovative student", "The winner for the best journal paper presentation in the field of semantic web goes to Bill Gates (2013–2015), So here we can see that the matching substring is "Bill Gates (2013–2015)
Third: Now the next task is to filtering the longest matching substring in the suffix tree containing the question and answer and replaces them with <Name> <Answer>
Fourth Now the above two processes are repeated for different pairs of <NAME> and <ANSWER>

c) The steps performed in Precision Calculations

First: The question term has to be queried for the same search engine. List out the top 1000 documents and download them.
Second: The documents, which contain the question terms, will be split into the sentences.
Third: Now whatever pattern we got from the step Extraction and standardization will be used to check the presence of each in the sentence got from Second step.
Fourth: The placeholder for answer may match any word or the correct answer term.
Fifth: In this step the precision is calculated, the formulae for calculating precision is dividing total number of patterns with the answer term present by total number of patterns present with answer term replaced by any word.

d) Procedure for Application patterns on new question:

In this step the new type of the question type is to be determined. The question terms will be identified after analyzing the existing system. The query should be framed from the question terms and Information retrieval process will be performed. Again the documents will be segmented into the sentences after removing the unnecessary html tags. After that question term will be removed by question tag. Now the use of pattern table generated, search for patterns with the question tag. Select words matching <ANSWER>. At last sorting has to be performed on these results by their precision scores.

Shortcomings

First: There is not external knowledge has been added to these patterns. In the answer sentence only one question term can be handled.
Second: The system is not standardizing into lowercase or uppercase. Thus "Bill Gates" and "bill gates" won't match.
Third: Definition question pose a problem. Even though the system's patterns matches but they are very general.

Extension

Canonicalization of words should be done in the system so that instead of enlisting all the possibilities of tagger, one could be used to cluster all the variations and tag them with the same term.

Conclusions

First: Web result easily outperforms the TREC result. This suggests that there is a need to integrate the output of the web and the TREC corpus.
Second: Many tools required by the sophisticated QA systems are language specific and required significant effort to adopt a new language. Though the answer patterns used in this method are learned using only a small number of manual training terms, one can rapidly learn patterns for new languages, assuming that the web search engine is appropriately switched.

Paper2: Is Question answering fit for the Semantic Web (SW)? : A survey

Introduction

Through this paper we focused on the QA system, which basically exploits the opportunity offered by the structured semantic information on the web. Through the user-friendly QA system, end users can easily query and explore this novel and diverse structured information space, which makes the vision of the SW a reality. Firstly we studied about the history of the QA systems developed by the various artificial and Database communities. Secondly we analyzed the potential of this technology to go beyond the current state of the art to support end-users in reusing and querying the SW content.

Explanation

Survey on the Goal and Dimension of Question Answering system.

Goal of QA is to allow user to ask question in Natural Language Processing using their own terminology and receive a concise answer. Through this section author throws the light on the multiple dimensions in the QA process. Now the dimensions in which QA system is classified for searching and querying SW content are as follows:

1) **Input** or type of question it is able to accept. (Keywords, Factoids, Understanding and casual reasoning, Temporal and spatial reasoning, Facts from different sources, common sense reasoning)
2) **Source** from which it can derived the answers (Structured (NLIDB), Semi-Structured (documents), Textual (web, TREC), Semantics (Ontologies & KBs)
3) The **Scope** (domain specific (Close), domain independent (Open), Proprietary KBs (private))
4) How it **copes** with traditional intrinsic problem that the search environment imposes in any non-trivial search system. (Large Scale, Heterogeneity (Mapping, Disambiguation), Openness (fusion, ranking))

Survey on Question Answering (QA) System Targeting Different Sources

Structured Database | Unstructured Free Text | Precompiled Semantic KBs |Semantics Ontologies

1) **Structured Database QA**: Early structured database NLIDB (Natural Language Interface Database) was unreliable because of the domain specific grammars, hard-wired knowledge or hand written mapping rules which could not be easily modified with different database and were difficult to port to different application domains. The new generation NLDIB's use the intermediate representation language, which expressed the meaning of the

user's question in terms of high-level concepts, which is independent of the database structure.

2) **Unstructured Free Text QA:** In this basically two kinds of system developed named as Document based and Web based Question answering system. Document based Question answering system typically involves two steps 1) identify the semantic type of the entity sought by the question (2) determining the additional constraint on the answer entity. In Web based Question answering system extract answers to factual questions by consulting a repository of documents 1) NL queries are translated into the IR queries (2) search engine over the Web, instead of a IR engine searching the document (3) the answer extraction module that extracts answers from the retrieved documents.
3) **Precompiles knowledge semantics based QA:** This is the latest development on the structured open question answering in which the semantics is introduce to search for the web pages based on the meaning of the words in the query, rather than just matching keywords and ranking pages by popularity.
4) **Semantic Ontology based QA:** The Semantic based Ontology system take queries expressed in NL and a given ontology as input, and return answers drawn from one or more KBs that subscribe to the ontology. Therefore, they do not require the user to learn the vocabulary or structure of the ontology to be queried. Ontology based QA system vary on two main aspects: (1) the degree of domain customization they require, which correlates with their retrieval performance, and (2) the subset of NL they are able to understand in order to reduce both complexity and the habitability problem, pointed out as the main issue that hampers the successful use of Natural Language Interface (NLI). There are certain limitations of domain specific QA approaches on the large Semantic Web. Such domain restriction may be identified by the use of just one, or a set of, ontologies covering one specific domain at a time. The user still need to tell the system which ontology is going to be used. To overcome the domain specific limitation of the previous approaches like AquaLog, PANTO, and GINSENG etc. the SW and NLP communities introduces Open QA over the Semantic Web.

The main clear advantage of ontology based QA systems over Structured based, Unstructured based, Proprietary based QA is the use of the NL query tool which in result prove a easy interaction for non-expert users. As SW is gaining momentum, it provides the basis of QA applications to exploit and reuse the structural knowledge available on SW.

Shortcomings:

1. Performance and Scalability issues are there. Balancing the complexity of the querying process in an open domain scenario and the amount of semantic data is still an open problem.
2. Most of the large datasets published in Linked Data are light-weight because the factual QA in large scale semantic system, in which intelligence becomes a side effect

of a system's ability to operate with large amounts of data from heterogeneous sources in an meaningful way rather that being primarily defined by their reasoning ability to carry out complex tasks.

Extension

To scaling up the SW in its entirety to reach the full potential of the SW, the more work can be done to bridge the gap between the semantic data and unstructured textual information available on the web. I think as the number of annotated sites increases, the answer to a question extracted in the form of lists of entities from the SW can be used as a valuable resource for discovering Web content that is related to the answers given as ontological entities. This will enhance the presentation and performance of traditional search engines with semantic information.

Conclusion

In the above paper we can conclude that QA over semantic data distributed across multiple sources has been introduced as a new paradigm, which integrates ideas of traditional QA research into scalable SW tools. In my view, there is a great potential for open QA approaches in the SW. We have seen that semantic open QA has tackled more problems than other methods for many of the analyzed criteria. To overcome from the limitation of search approaches, that restricts the scope to homogenous or domain-specific content, current QA systems have developed syntactic, semantic and contextual information processing mechanism that allow a deep exploitation of the semantic information space.