ISSN 2395-8618

Building an Information Extraction and Question Answering Model for Text Based on the Human Brain Process

F. A. K. Hemant

***Abstract*—An information extraction and question answering model for text, which is based loosely on the human brain process, is showcased in this paper. The ideology used is based on how humans perceive and interact with text, and the process of storing the text for future reference. Each word of each sentence is cross referenced and linked with all available information and the answer is given based on matching information found. The model is basic, but the future applications and scope of improvement is also shown.**

***Index Terms*—Question Answering, Linguistics, Information Extraction, Text Analysis**

I. Introduction

The majority of data and information in the world is transmitted and stored in the form of text. It could even be said that language and text are the backbones of the human civilization. Without communication of knowledge and ideas, humans could not have progressed to the state they are in now.[5][6]

The first use of language was to communicate, whether it be ideas or information. With this motivation in mind, the goal was to build a simplistic computer model for analyzing and storing information in text, for easy retrieval [7].

This document is divided into four sections. The first section addresses the ideology used to build the model. The second section displays the model built so far. The third section shows the results achieved. The last section talks about the future scope and applications of this project.

II. Ideology

A very simple adaptation of the human brain's process of perceiving information is used. Consider a simple sentence:

"Bob went to Jim's house last weekend."

The first thing that is addressed is the identity of the entity "Bob". The brain first goes about remembering information regarding the entity, and the latest known instances when the entity was referenced. This is followed by the same process regarding the second entity, "Jim".

The information regarding the other entities, i.e "house" and the time period "last weekend" are also recalled. Then, the information is stored, and this information is added to the list of instances recalled.

The process is only one way that a human brain might perceive information, and the existence of other processes is disregarded for now. This process is used because it functions at the most basic level, and is thus easier to implement, while also following the norms established i.e. to emulate the brain in at least the most rudimentary way [6].

III. Implementation

A system is built which tries to emulate the process shown in the last section. Each aspect is explained in detail in the following subsections.

A. Data Used and Depiction of Process

The flowcharts Fig. 1 and Fig. 2 depict the salient features of the process. The data sources are listed below:

1. Pang & Lee Data Set: A collection of movie reviews [1]
2. Newspaper Data: Local newspaper data was collected and used. About 400 English news articles were collected.
3. Wikipedia Data: Around 200 Wikipedia pages were used.

B. Text Parsing Module

Stop words in the input text are removed and the remaining words are considered to be the entities in the text.

Input text is parsed using the Stanford Parser. The dependency parsing module of the Stanford parser is most useful and efficient in this present endeavor, as we can obtain the relations between entities in data, which is most useful for storing data for easy retrieval. For example, for the sample sentence used in the last section, "Bob went to Jim's house last weekend.", the dependencies output is:

Manuscript received on November 18, 2016, accepted on October 12, 2017, published on June 30, 2018.

F. A. K. Hemant is with the International Institute of Information Technology, India (+919494868838; e-mail: kancharla.hemant@gmail.com).

ISSN 2395-8618

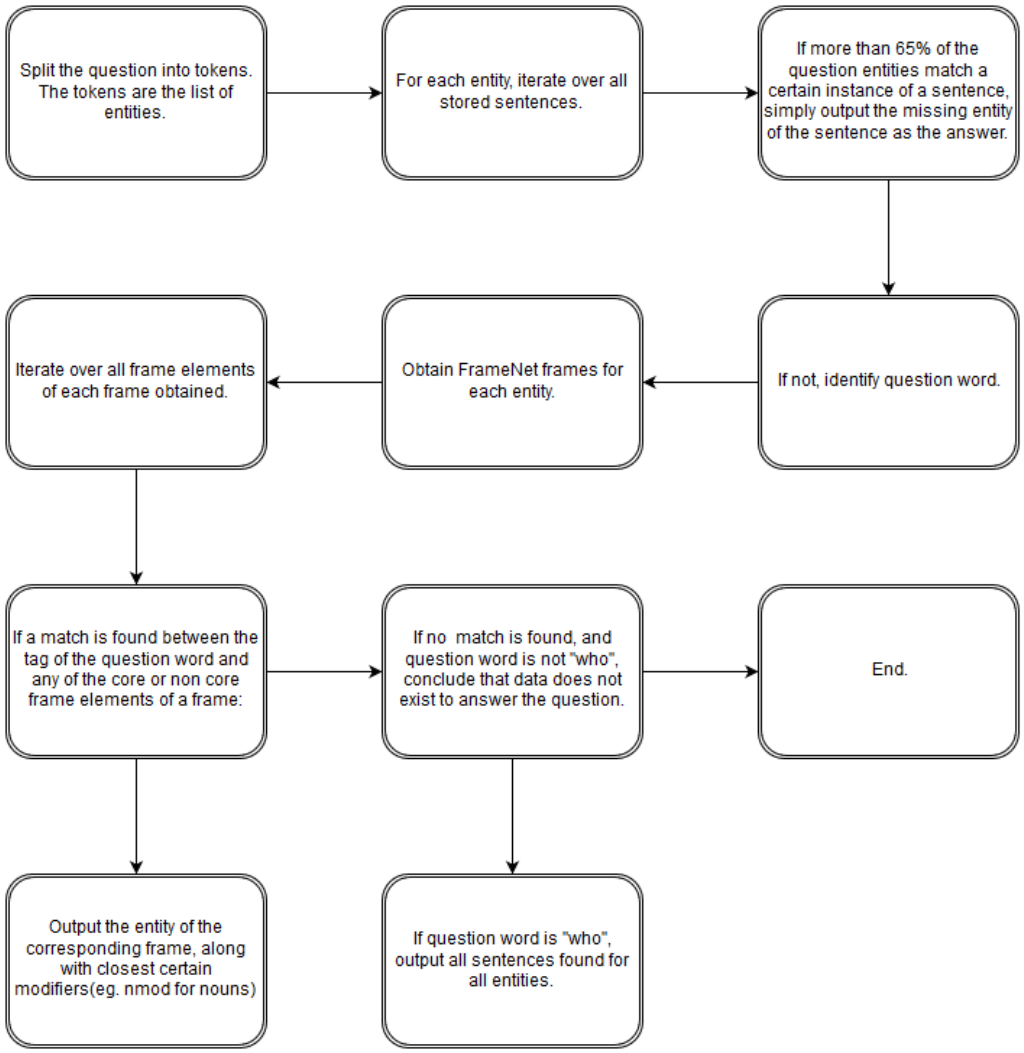

Figure 1. General flowchart of the process.

nsubj(went-2, Bob-1)
root(ROOT-0, went-2)
case(house-6, to-3)
nmod:poss(house-6, Jim-4)
case(Jim-4, 's-5)
nmod(went-2, house-6)
amod(weekend-8, last-7)
dobj(went-2, weekend-8)

Thus, we have a set of the dependencies between the words.

C. Text Storage Module

Text is stored in the form of a dictionary of lists in python. All the dependency tags associated are also stored. For the entity Bob, the resulting information stored would be:

Bob : {1{nsubj(went-2, Bob-1),
root(ROOT-0, went-2),
case(house-6, to-3),
nmod:poss(house-6, Jim-4),
case(Jim-4, 's-5),
nmod(went-2, house-6),
amod(weekend-8, last-7),
dobj(went-2, weekend-8)}}

The same is done for all the entities in the sentence, including "house". For decreasing space taken, the string of tags is stored once, and subsequently referenced in each entity.

This is done because as each entity is referenced, for effective question answering, it is relevant to have the information of all occasions the reference was instanced [8][9].

D. Question Answering Module

Question answering is the main test through which the system can be assessed and tested. A sample question:

"To whose house did Bob go to last weekend?"

ISSN 2395-8618

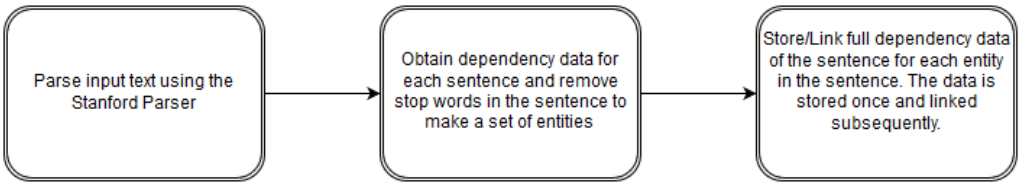

Figure 2. Flowchart of one step of the process.

The steps in which simple questions are answered are:

1. Identify entities (in this case, tokens) using a tokenizer i.e. house, Bob, last weekend
2. Iterate over all entries of each entity of the question sentence in the database.
3. Output the instance entry with the maximum matching (at least above 65 percent, this figure established by manual testing)
4. Output the missing entity in the instance as the answer

This is a simple and naive approach for basic questions. In the case that this fails, the case is either that the question is complex, or that no data exists to answer the question. In the case of a complex question like:

Where is Bob?

In this case, the matching approach wouldn't work. Thus, a different approach is used:

1. All question words are hard coded to tags, for instance, "where" to the "spatial location" tag and "when" to the "temporal collocation" tag (except the "who" tag).
2. FrameNet is used. The FrameNet project is building a lexical database of English that is both human- and machine- readable, based on annotating examples of how words are used in actual texts. It is a dictionary of more than 10,000 word senses, most of them with annotated examples that show the meaning and usage.

Frame elements are frame-specific defined semantic roles that are the basic units of a frame.

In the case of "where", which has a spatial tag, all entities in instances of the entity in the question are searched in framenet. If a "spatial" tag is located in the core or non-core frame elements of the frame of the entity searched, then the entity, and its closest modifiers (from the stored Stanford dependency tags) are outputted.

For example, "Where" has the "spatial" tag. Each word in the sentence is searched in framenet. For the entity "house", which is in the frame "buildings", there is a "spatial" tag in the non-core frame element. Thus, the entity, and its closest modifier, which is "Jim's"(only certain tags are considered, like nmod) is chosen as the answer.

The tags for each question word are:

- When: Temporal_Collocation
- What: Entity
- Why: Reason
- Which: Entity
- Where: Spatial_Co-location
- How: Means

The approaches combined give nominal results for all question words except "who". For "who", all the instances themselves are outputted. In the following section, the types of questions used and observations seen are displayed [2][3][4].

E. Results & Observations

This approach was taken after first manually checking the viability of using such an approach. About 100 sentences from newspaper data were taken and checked manually. As an accuracy of more than 60 percent was obtained, the work was continued. Accuracy in this case is simply meant to be whether any frame element matched with the tag of the question word. The tags of the question word were also decided upon after tweaking with other alternatives. The best results were obtained when using these tags, which are the tags that the question words have themselves in framenet.

The results obtained for each data set are displayed in Table 1.

TABLE 1.
Obtained results.

Dataset	Accuracy
Pang & Lee	67%
Newspaper	69%
Wikipedia	53%

Accuracy was checked manually for 200 sentences from each data set. Questions were 50% simple questions, and the rest complex.

It is to be noted that in the case of newspaper data, the highest accuracy was achieved. This can be attributed to the style and general format of sentences in the data. As the majority of the sentences are used to state facts, it is easier to answer questions. In the cases where answers spanned a phrase, the first approach gives answers accurately. It was

ISSN 2395-8618

also noted that most of the questions of such a variety did fall under the category of the first approach. This could be because in the cases where specific answers are needed, the questions also need to be specific. E.g., for the specific date of a certain event, some other information like the location of the event must also be given the question. And the presence of such information enables the first approach to work.

The lowest accuracy was in the case of the Wikipedia dataset. This is because of the high presence of data which might not be directly related to the topic of the text itself, at least in a way that can be identified by the present system. Answers which had too many sentences were also regarded as false, as such the accuracy is lower.

The performance was uniform on the Pang & Lee data sets, as the data itself was fairly uniform. Not many abnormalities were noticed, but the abundance of data for the "who" question was noted, as a very naive approach was used in that case.

IV. Conclusion

The future development of this model lies in using also the context of the text to answer questions regarding a sentence. Question answering has a wide variety of uses. Although for now the model cannot contend either in the scope of accuracy or complexity, it can compete in the area of varying domains. Because of the general implementation, which is not restricted to a certain domain, the model can be used in any domain for reasonable results.

A time-based model can also be built based on this model, which can record the states in the text. For example, if there are two sentences-"Bob went to Jim's house last weekend", and, "Bob is in Harry's house now", and the question regarding the location of Bob is asked, the system, which keeps a track of the temporal implications of sentences, should be able to give the correct answer, i.e. Harry's house. This is only one state, and other states can also be recorded and used to answer questions. This would increase accuracy by quite a fair bit.

Another improvement would be adding the capability of taking social media and chat data as the input. This would further increase the number of areas in which the model can be used.

Overall, this was only the first step in building a model which can even be referred to as being based on the human brain process. Future work will revolve around first increasing the viability and accuracy of the system, before again focusing on replicating the human brain process, to any extent.

Acknowledgment

I would like to thank Dr. Radhika Mamidi of the International Institute of Information Technology, Hyderabad for her steadfast guidance and belief in my idea.

References

[1] Bo Pang, Lilian Lee: “Opinion Mining and Sentiment Analysis”- Foundations and Trends in Information Retrieval archive, Volume 2 Issue 1-2 (January 2008), Pages 1-135

[2] Deepak Ravichandran, Eduard Hovy: “Learning surface text patterns for a Question Answering system”- ACL '02 Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Pages 41-47

[3] David L. Waltz: “An English language question answering system for a large relational database” – Magazine Communications of the ACM, Volume 21 Issue 7(July 1978), Pages 526-539

[4] Dan Moldovan, Sanda Harabagiu , Marius Pasca , Rada Mihalcea , Roxana Girju , Richard Goodrum, Vasile Rus: “The structure and performance of an open-domain question answering system”- Proceeding ACL '00 Proceedings of the 38th Annual Meeting on Association for Computational Linguistics,Pages 563-570

[5] Hecht-Nielsen, R.: “Neurocomputing: Picking the human brain”- IEEE Spectrum (United States) Volume: 25:3(1988-03-01)

[6] John C. Mazziotta, Arthur W. Toga, Alan Evans, Peter Fox, Jack Lancaster: “A Probabilistic Atlas of the Human Brain: Theory and Rationale for Its Development”- The International Consortium for Brain Mapping (ICBM), Volume 2, Issue 2, Part A (June 1995), Pages 89-101

[7] Steven Pinker: “How the Mind Works”- Annals of the New York Academy of Sciences, Volume 882, Great Issues For Medicine In The Twenty-First Century: Ethical And Social Issues Arising Out Of Advances In The Biomedical Sciences (June 1999), Pages 119–127

[8] Stephen Soderland: “Learning Information Extraction Rules for Semi-Structured and Free Text”- Soderland, S. Machine Learning (1999) 34: 233

[9] Ellen Riloff , Wendy Lehnert: “Information extraction as a basis for high-precision text classification”- ACM Transactions on Information Systems (TOIS), Volume 12 Issue 3(July 1994), Pages 296-333