CMUQA: Multiple-Choice Question Answering at NTCIR-12 QA Lab-2 Task

Dheeru Dua
Carnegie Mellon University
ddua@cs.cmu.edu

Bhawna Juneja
Carnegie Mellon University
bjuneja@cs.cmu.edu

Sanchit Agarwal
Carnegie Mellon University
sagarwa1@andrew.cmu.edu

Kotaro Sakamoto
Yokohama National University
National Institute of Informatics
Carnegie Mellon University
sakamoto@forest.eis.ynu.ac.jp

Di Wang
Carnegie Mellon University
diwang@cs.cmu.edu

Teruko Mitamura
Carnegie Mellon University
teruko@cs.cmu.edu

ABSTRACT

The first version of the UIMA-based modular automatic question answering (QA) system was developed for NTCIR-11 QA Lab task[5]. The system answers multiple-choice English questions for the Japanese university entrance examinations on the subject of world history. We made improvements in the current system by adding components focused towards Source Expansion and better Semantic Understanding of the question in terms of events and their time-lines.

Team Name

CMUQA

Subtasks

National Center Test, Formal Run (English), Phase-1 and -3

Keywords

Question Answering, Source Expansion, Machine Reading, World History, University Entrance Examination

1. INTRODUCTION

For the National Center Test task of NTCIR QA Lab task[4], participants need to design an automatic Question-Answering system that can answer World-History questions asked in National Center Test for University Admissions in Japan.

There are different types of questions that are asked in the National Center Test. The question types include “choose the right combinations of incorrect and correct answers”, “fill in the blanks”, “chronologically arrange the events” and “image based questions”. The gold set of questions is provided in an XML format that entails a comprehension with underlined phrases and question based on these phrases. The question is followed by a set of answer choices to select from. The system can be developed for questions in Japanese or English language. The participant in the Japanese task provided with Japanese text books on world history. The English task participants can use any external knowledge source like Wikipedia and any other websites which could be crawled except the National Center Test archives which contain the actual answer keys.

The last year’s team [7] developed a generic Assertion-Based System. The system is implemented in the form of a UIMA pipeline. It consists of annotators for preparing a UIMA-CAS object for all the question, answer choices and other meta-information. The Answer to different scorers implementing various algorithms to rank these assertions. The scores from all the scorers are combined using Voted Majority and the top assertion is selected as an answer.
The version 1.0 of the system is focused more towards IR based approaches and lacks a semantics aspect to question and document understanding. In the version 1.1, we tried to augment a semantic side to data collection and scorer implementation to the previous system.
In version 1.1, we experiment with machine reading for expanding data collection to crawl more sources not limited to Wikipedia. We also experiment with a different approach to handling timeline based questions from the past attempts. In the next few sections we will talk about the new components introduced in version 1.1.

2. KNOWLEDGE SOURCE

1. Wikipedia
2. Gutenberg Collection
3. Japanese Textbook Four world history textbooks translated into English from Japanese by Google Translate [1]
4. SourceExpansion: This knowledge source contains documents pertaining to history crawled from sources other than Wikipedia on web.

3. ERROR ANALYSIS

As per the error analysis done on a sample of the results from last year’s system, the statistics for categories where the system performed poorly are given below

[1] https://translate.google.com

Year	Image-based questions	IR based questions	Event-Timeline questions	Miscellaneous
1997	5	4	3	0
2001	1	5	0	3
2005	0	1	2	1
2009	4	3	1	2

The above numbers describe the number of questions that were answered incorrectly in the sample of the results we obtained from the existing system. The analysis shows that we can improve the system by increasing the source collection with documents on world history and develop a better strategy for handling timeline questions.

4. EVIDENCING MODULES

4.1 Source Expansion

4.1.1 Extract Entities using DBPedia as Annotator

Extract Key Entities.

We first filter out all the context question data from the input xml file and then annotate it using DBpedia annotator to get the surface form and its corresponding DBpedia URL.

Extract the facets.

For every given entity DBpedia URL, we extract the properties and ontology tag information which serves as facets for each given entity. This information is usually contained in the InfoBox of Wikipedia page. We store all these unique facets along with the corresponding entities in a file. For example:

Facets Along with Entities
is Part Of Military Conflict Middle Ages
cultures Middle Ages
style Middle Ages
architectural Type Middle Ages
structural System Middle Ages

4.1.2 Extract entities using NER Tagger

NER Tagger gives us entities which are not present in DBPedia.Thus, we also run our question context data through NER Tagger to get various entities like PERSON,LOCATION, ORGANISATION etc. which were not present in Wikipedia and save them as queries in the same file. For example:

NER Tagged Entities
history China Taizong
history Hongwu Emperor
history Sui East
history Kangxi Emperor

4.1.3 Extract entities from Wikipedia History Titles

Wikipedia contains a lot of information about sundry historical events, but not all the events are covered in detail.Thus we decided to crawl all the historical titles as queries from Wikipedia pertaining to World History and then use them as query to Bing API to get deeper context from sources other than Wikipedia.

Wikipedia History Titles
18th century in china
wars involving the soviet union
dynasties in chinese history
8th-century architecture

Once all these queries were compiled together, it was fed as input to Bing API and the relevant text documents were indexed in Solr.

4.2 New Improvements

4.2.1 Query Normalization

There were cases in some questions where we had two assertions and the correct assertion was shorter in length.Thus, the longer assertion got the overall highest score. For example:
Assertion 1: Manila was a Spanish trading post
Assertion 2: In Thailand, the Sukhothai kingdom prospered through trade.
Here, option 1 was correct but option 2 was selected due to lack of normalization.Thus, adding query normalization helped improve accuracy of the system for some questions.

4.2.2 Adding More Context to Assertion

For some questions, the assertions lacked context or there was not enough background information. Due to this, the correct assertion got less overall score value. To overcome this, we added context information to the assertions using the question context information and the underlined text in question. To add more context to assertion, we added five words from both left and right side of underlined text since question is focused on some event about the underlined text.Next, we identified key entities in above extracted context using DBPedia annotator and then appended these entities to the assertion text.For example:
In terms of characteristics of the First World War, firstly, <uText id="U1"><label>(1)</label>many new weapons appeared</uText>, which increased the scale of the human and physical damage
The context comes out to be : **World War I**. Thus, now our assertion would have the context information that the underlined text is in reference to World War I. This helped the system select the correct assertion for given question.

4.3 Semantic Understanding of the question in terms of events and their time-lines

The main idea here is to assign a score for various answer options in chronology based questions. We use a time ordered knowledge base to look up the timeline of such events and assign a score to their order based on these timelines. The complete system is described in Section 6.

4.4 Graph Model based Passage Ranking

The CMUQA system of NTCIR 11 QA Lab has a passage evidencer which retrieves documents from Wikipedia, retrieves passages from the retrieved documents and rank the retrieved passages. We improved the passage ranking using a graph model based algorithm in Phase-3. We hypothesize that the multi-layer graph model based ranking for query-biased summarization[3], which relies on three layers composed of passage layer, sentence layer and word layer, Solr score of passage, basic elements overlap similarity between two sentences, semantic relatedness between two words, is effective on question answering task. We use BEwTE[6][2] as basic elements. We calculate semantic relatedness between two words by WUP[8][3] which relies on the depths of two synsets in WordNet. This module was not included in the official submissions because it took unexpectedly too much time to output the results by the submission deadline of Phase-3 formal run. Table 1 shows our results at the formal runs. Table 2 shows the results after replacing the passage ranker to this graph model based passage ranker. The scores and accuracies of Priority-1 and -3 systems are increased by the graph model based passage ranker, and the score and accuracy of the Priority-2 system are decreased by it.

Table 1: Results at the formal runs

	Priority	Total Score	# of Correct	# of Incorrect	Accuracy
Phase-1	1	29	13	28	0.32
	2	30	13	28	0.32
	3	32	14	27	0.34
	Baseline	32	14	27	0.34
Phase-3	1	25	9	27	0.25
	2	23	8	28	0.22
	3	24	9	27	0.25
	Baseline	27	9	27	0.25

Table 2: Results using the graph model passage ranker

	Priority	Total Score	# of Correct	# of Incorrect	Accuracy
Phase-3	1	29	10	26	0.28
	2	20	7	29	0.19
	3	34	12	24	0.33

Furthermore, we improved the Priority-1 system with the graph model based passage ranker. The Priority-1 system uses coordinate ascent algorithm for learning to rank and the training data are 189 questions in the five years of 1997, 2001, 2005, 2007 and 2009. The result of the leave-one-out cross validation for the training data was 49.5%. When we replaced the coordinate ascent algorithm to the random forest algorithm, the result of the leave-one-out cross validation was increased to 65.1%. Hence, we evaluated the end-to-end result of the Priority-1 system with the graph model based passage ranker and the random forest algorithm. The Table 3 shows that both the total score and the number of correct answers in the end-to-end evaluation result were increased.

Table 3: Results using the graph model passage ranker and the random forest algorithm

	Priority	Total Score	# of Correct	# of Incorrect	Accuracy
Phase-3	1	33	12	24	0.33

4.5 Japanese Corpus & Query Overlap Scorer

From an error analysis of the current system, we observed that Semi-Phrasal scorer on Gutenberg history books collection performs almost as good (and sometimes better) than Simple Solr Scorer and Semi-Phrasal Solr on subset of Wikipedia history.One possible explanation for this behavior is that Gutenberg corpus contains more relevant information and lesser noise than Wikipedia.

This observation motivated us towards using Japanese text books provided by the NTCIR QA Lab Task organizers as one of knowledge source for the system. Hence, we translated the Japanese text books to English using google translator. Once translated, we indexed the xml japanese text books in Solr.

This scheme is similar to Wikipedia scorer. On running few experiments, we found that it did not provide any improvement over the Wikipedia evidencer. To improve the scoring scheme, we added another component to the final evidence score i.e. Query Overlap score. Query Overlap is the percentage of query terms that appear in the document or a section of the document. If the score is high, it is likely that the document is relevant to the query.

To compute query overlap score, we utilized Solr's highlighting feature. Solr Highlighting returns snippets of text from the document that match the user's query. The default size of the fragment was experimentally selected as 100 characters. To compute the overlap score, we considered only the topmost document and the first snippet returned from that document.The score of the document was then combined with query overlap score by taking their weighted average. Weights were determined experimentally by multiple runs on different training data sets. The final scoring equation is:

$$\text{finalScore} = 0.72 * \text{maxScore} + 0.28 * \text{percentageOverlap}$$

This provided considerably better performance than either score used in isolation. The results are tabulated below.Also, the overall accuracy is better as compared to Gutenberg knowledge source.

Accuracy	Solr OR query max-score	Query-Overlap score	Weighted combination
1999	0.257	0.257	0.314
2003	0.324	0.270	0.297
2007	0.455	0.273	0.515
97-01-05-09	0.314	0.259	0.322

5. MACHINE READING

Wikipedia as a knowledge source has a lot of noise and it doesn't cover all the historical events in greater depth.Thus, a new knowledge source based on the past question context was built called SourceExpansion. The key steps were as follows:

1. Extract key historical entities from question context. 2. Next, obtain various facets(which describes them) of the above extracted entities 3. Construct a query which consist of entity along with its facet and ping them using Bing API to get relevant documents for indexing in Solr.
Following are the results obtained after adding Source Expansion and new Improvements to the current code.

[2] https://github.com/igorbrigadir/bewte

[3] https://code.google.com/archive/p/ws4j/

S.No	Year	Accuracy	Percentage Accuracy
1	1997	24/40	0.60
2	2001	22/41	0.56
3	2003	15/41	0.34
4	2005	23/36	0.64
5	2007	21/36	0.58
6	2009	20/36	0.57
7	97-01-05-07	76/151	0.49
8	97-01-05-07-09	93/189	0.49

Following are the performances of the knowledge sources in our system.

S.No	Source	Accuracy on 97-01-05-07	% Accuracy on 97-01-05-07
1	Wikipedia	61/151	0.41
2	External Source	48/151	0.317
3	Japanese Text	51/151	0.337
4	Gutenberg	43/151	0.285

The results above suggest that even just using External Source as a knowledge source works as comparable to Gutenberg history books as a source.In addition, when we give weights to Wikipedia Source, External Source and Japanese text source, it improves the accuracy of the system by a huge margin(61/151 to 76/151).

6. SEMANTIC REASONING

NTCIR QA Lab task is centered around history questions, which makes time spans and events very vital to the task. Building a time-centric knowledge base can be very influential in making the correct choices. This is why a time-based knowledge base seems like a promising approach. However, the conventional knowledge bases are brittle towards noisy real-world data. Probabilistic Knowledge Bases are a better choice in this setting which is why we chose Markov Logic Networks which synergize well with imposing real world constraints on probabilistic graphical models.

6.1 Markov Logic Networks

Markov Logic Networks[2] are great for representing first order logic rules in a probabilistic setting. We approach the event-sequencing problem as a link prediction task leveraging dependency properties in natural language. As shown in figure 1, we can model the relations like before, after, during and simultaneous among event and time spans across sentences. The relations are captured by predicate rules based on various features like conjunctions and prepositions, tense of events, lexical markers like after, before, during etc and meta-data obtained by parsing time expressions with CCG parser.

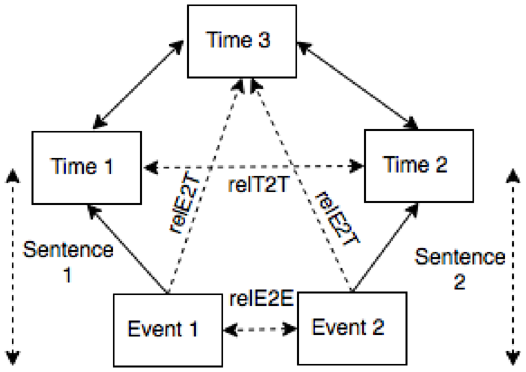

Figure 1: Event and Time interactions.

The type of questions targeted with this component were chronology-based question like below

```
Choose the correct chronological sequence of events
relating to the Cold War.
1.) Warsaw Treaty Organization formed - Berlin
    blockade - Cuban missile crisis - Japan-US
    Security Treaty signed (1951)
2.) Berlin blockade - Japan-US Security Treaty
    signed (1951) - Warsaw Treaty Organization
    formed - Cuban missile crisis
3.) Japan-US Security Treaty signed (1951) -
    Cuban missile crisis - Berlin blockade -
    Warsaw Treaty Organization formed
4.) Berlin blockade - Warsaw Treaty Organiz-
    ation formed - Japan-US Security Treaty
    signed (1951) - Cuban missile crisis
```

6.2 System Architecture

The design adopted for inducing event-time graph as a graphical Markov network is illustrated in Figure 2. First, the entire dataset was annotated for time and event spans. The event spans were extracted from a event-mention detection system built for RichERE event types. It was trained using a Conditional Random Field based on various language features like POS tags, history words, brown clusters etc. The time spans were resolved using the CCG based parser [1]. After the documents are annotated with events and time spans, various features are collected based on tense of events, time resolutions and lexical markers. These features are used to create a database of triples containing event and time ids with these property features. Then, the weights are learned for the first order predicates based on the available data making the rule imposition soft. Then these learned weights can be used to predict time-based relationship among events to populated a time-ordered database.

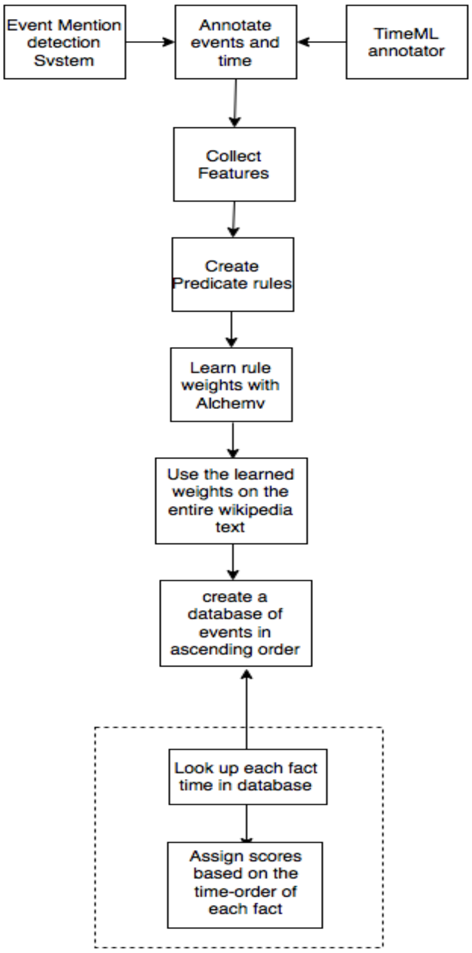
Event Mention detection Svstem
Annotate events and time
TimeML annotator
Collect Features
Create Predicate rules
Learn rule weights with Alchemv
Use the learned weights on the entire wikipedia text
create a database of events in ascending order
Look up each fact time in database
Assign scores based on the time-order of each fact

Figure 2: Event and Time interactions.

6.3 Experimental Setup

We used 20 Wikipedia annotated articles and few randomly selected TimeML documents to train the Markov Network for link prediction. We used alchemy[4] toolkit to train the Markov Logic Network. After training the network on the above documents the weights learned per predicate are used to predict the relationship among annotated events on history wiki corpus.

System	Accuracy
Baseline	61/151
After adding temporal module	64/151

6.4 Future Scope

Once the database is created it can be utilized for many different tasks. Document summarization is one of the examples, where we can entails all the events occurrences in a chronological order for instance in News articles. We can also use this to extract an ontology of events which can be useful in building language generation models based on context.

7. CONCLUSIONS

We tried a lot of different evidencers to understand which dimension we should focus on in making the system better. We realized that we need to concentrate on building semantically smarter system to understand and make inferences from complex Wikipedia text. The Wikipedia text which is the most important background corpus for English task is very complicated to interpret. We need to add evidencers which have better lexical understanding of the Wikipedia text and the questions.

8. REFERENCES

[1] K. Lee, Y. Artzi, J. Dodge, and L. Zettlemoyer. Context-dependent semantic parsing for time expressions. In *Proceedings of ACL*, 2014.

[2] R. Matthew and P. Domingos. Markov logic networks. In *Machine learning 62.1-2*, pages 107–136, 2006.

[3] K. Sakamoto, H. Shibuki, T. Mori, and N. Kando. Fusion of heterogeneous information in graph-based ranking for query-biased summarization. In *Proceedings of the First International Workshop on Graph Search and Beyond, GSB 2015, co-located with The 38th Annual SIGIR Conference (SIGIR 2015)*, pages 19–22, 2015.

[4] H. Shibuki, K. Sakamoto, M. Ishioroshi, A. Fujita, Y. Kano, T. Mitamura, T. Mori, and N. Kando. Overview of the ntcir-12 qa lab-2 task. In *Proceedings of the 12th NTCIR Conference*, 2016.

[5] H. Shibuki, K. Sakamoto, Y. Kano, T. Mitamura, M. Ishioroshi, K. Y. Itakura, D. Wang, T. Mori, and N. Kando. Overview of the ntcir-11 qa-lab task. In *Proceedings of the 11th NTCIR Conference*, 2014.

[6] S. Tratz and E. H. Hovy. Summarization evaluation using transformed basic elements. In *Proceedings of TAC-2008*, 2008.

[7] D. Wang, L. Boytsov, J. Araki, A. Patel, J. Gee, Z. Liu, E. Nyberg, and T. Mitamura. Cmu multiple-choice question answering system at ntcir-11 qa-lab. In *Proceedings of the 11th NTCIR Conference*, 2014.

[8] Z. Wu and M. Palmer. Verb semantics and lexical selection. In *32nd Annual Meeting of the Association for Computational Linguistics*, page 133âĂŞ138, 1994.

[4] http://alchemy.cs.washington.edu/