Report
Topic: Machine Reading for Question Answering

Kavya Srinet, Liping Xiong, Qiang Zhu, Srivaths Ranganathan, Xiaoqiu Huang, Yepeng Yin

1. Motivation:

Traditionally, Question Answering (QA) systems query search engines or have access to a crawled collection of the documents on the web to use as data sources in their search and answering process. In many cases, QA systems work on only some domain and might not have to answer questions on everything. When the questions that the QA system will be working on is restricted to some domain or topics, most of the data on the Web will be irrelevant to the questions that it must answer. So, using the whole Web as the source corpus becomes unnecessary and might even affect the performance of the QA system. Therefore, having a reduced and enhanced corpus specific to the domain can improve the QA system's performance on future unseen questions on the same domain.

2. Problem Definition

Given an initial set of domain specific questions and answers, build a corpus to be fed into a QA system to answer future questions from that domain.

3. Machine Reading

Machine Reading is a process that automatically analyzes the questions and answers to "read" documents from the data sources and builds a relevant corpus that caters specifically to the domain. It includes following steps:

1. Retrieve relevant documents from data sources like search engines.
2. Build a relevant corpus from the set of documents retrieved.
3. Extract a set of important concepts and entities from this corpus and use it to update the corpus.
4. Do the above process iteratively until we have enough information to answer any question about the domain.

4. Related work

One of the major work that was done in this field was by Nico Schlaefer that gave us the motivation to work on this project.

Schlaefer, Nico, et al. [1] performs source expansion by extracting text nuggets from text corpus, scores those nuggets and combines them as a pseudo document. His work did one step of source expansion only, and he used documents as the seeds to further expand on.

Wang, Richard C., et al. [2] extracts a list of Named Entities to produce another list of answers from the system and combines the results with the original list. This paper took a

step further and performed a two step expansion by extracting seeds again from the constructed pseudo document.

Das, Dipanjan et. al. [3] summarizes and extracts information from single and/or multiple documents, using standard ML approaches. They explored some approaches like Decision trees and Graphs for scoring the sentences better.

Lally, Adam et. al. [4] analyzes question first to answer with high confidence. They use the entity type expected by the question to extract the answer with high confidence if the answer type matches the expectations.

We plan on integrating the above mentioned techniques in our approach and leverage the most out of the ensemble of these approaches. As of now, we have integrated the first and second approaches, we plan to implement the third and fourth approaches next.

5. Motivation for Knowledge Graph:

The work by Nico et. al. focused on expanding a set of documents to get a larger corpus which contains more documents. He achieved this by expanding on the information in the source documents. Our work builds on this approach and improves it by focusing on how this process can be done in multiple iterations to iteratively expand and enhance the corpus. This additional complexity involved in iteratively building the corpus requires us to store the data explored at different stages of the MR process with some structure. Moreover, if we need to generate a corpus that is not just a subset of the whole Web but is also in someway enhanced, then the systems needs to have a basic understanding of the relationship between the data in the corpus. This leads naturally to the structure of a knowledge graph to store the data internally.

6. Scope of prototype

- The input to our system is a set of questions and the corresponding gold-standard answers.
- Output: A domain specific corpus based on the input questions and answers.
- Measurement: Recall of the results of Watson QA system using Machine Reading vs Recall achieved using just the bing/wikipedia top results as corpus
- Size: Explored corpus size should be bounded by some function of number of questions
- Time: No bound on computation time (as of now).
- Data sources : Bing, DBpedia

7. Approach & Design:

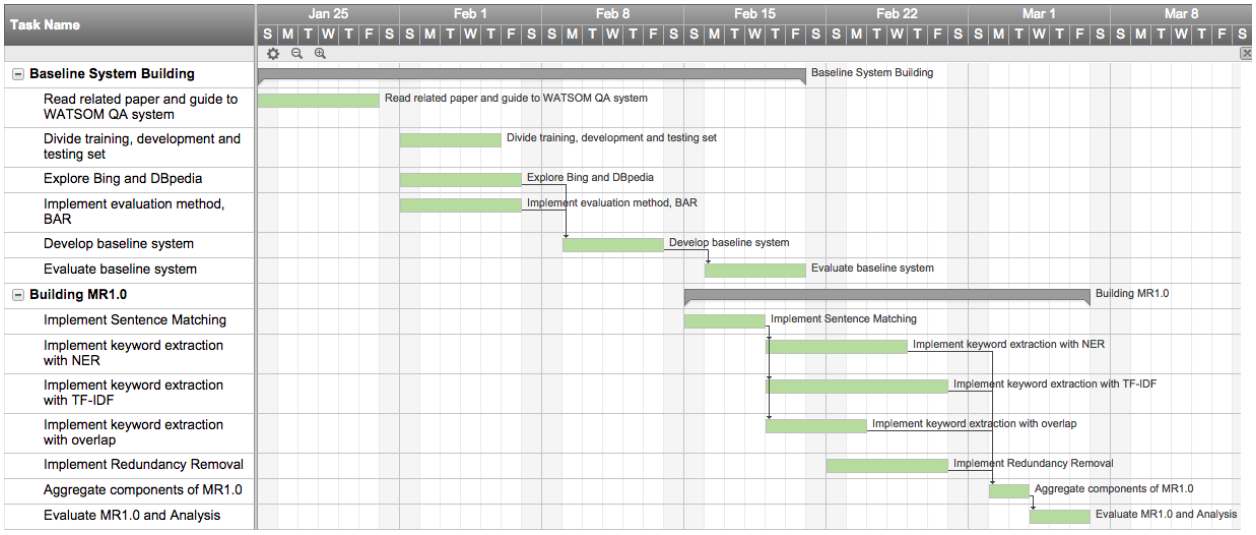
Our system presents an approach that performs the source expansion task iteratively.
We model the machine reading process as states and transitions, where each state is defined by the set of seeds, a corpus and a snapshot of the knowledge graph. The transition will transit from one state to a new state by updating corpus and seeds in the current state or pruning the corpus. During transitions, the expansion is performed iteratively and the

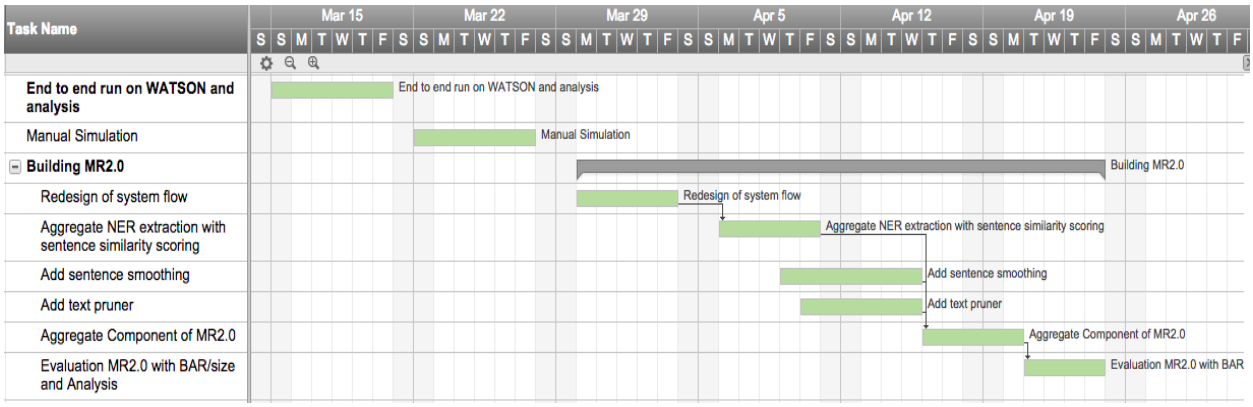
generated corpus is updated. The way we pick which seed to expand on is : Pick seeds with the highest weight and expand on it (by crawling on the documents associated with this seed), decrement the weight with every expansion, we keep doing this process iteratively for all seeds in the state that have a weight higher than threshold. After we have done the required operations on a state, we make a decision of which state to go next to and which operations to perform to make the transition.

8. Plan

We intend to try to implement different algorithms for each component on the framework. Then the corpus generated by our system will be evaluated on accuracy(precision and recall) and memory of corpus. In addition, since evaluation on Watson cannot be automated, an easier-to-use QA system needs to be chosen to test on.

Shown below is our executed plan:





9. Dataset and sources

Our dataset focuses on the Terrorism domain. There are in total 96 questions and gold standard answers to them. In order to build our system and evaluate our results, the datasets are split into three subsets:

a. Training set - 62 (questions and gold standard answers)
b. Development set - 19
c. Test set -15

It is noted that all question types divided evenly into different sets to in order to accurately evaluate our system and avoid overfit. Data sources to which we send query is Bing and DBPedia.

10. Evaluation metric

We evaluated our system performance using the metric : Binary Answer Recall

$$BAR = \frac{M}{N}$$

where

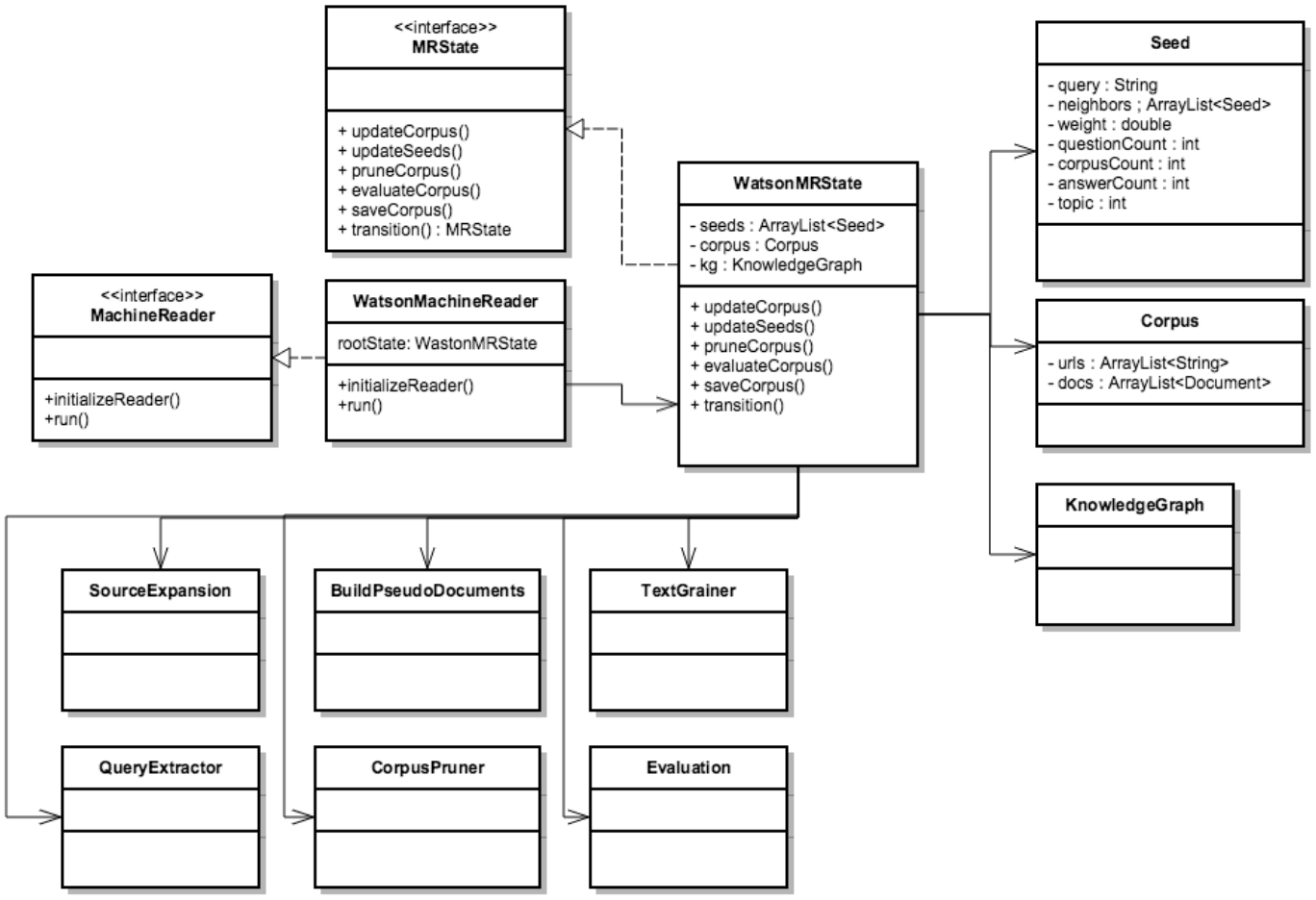
M is the number of questions that found at least one gold-standard answer in the corpus and

N is total number of questions.

We also did some experiments of plotting the BAR scores versus the corpus size, and we could used this metric too for the later part of our experiments.

11. Implementation

Our system contains 5 main parts: getting corpus by querying search engines, try to find suitable



12. Preliminary results

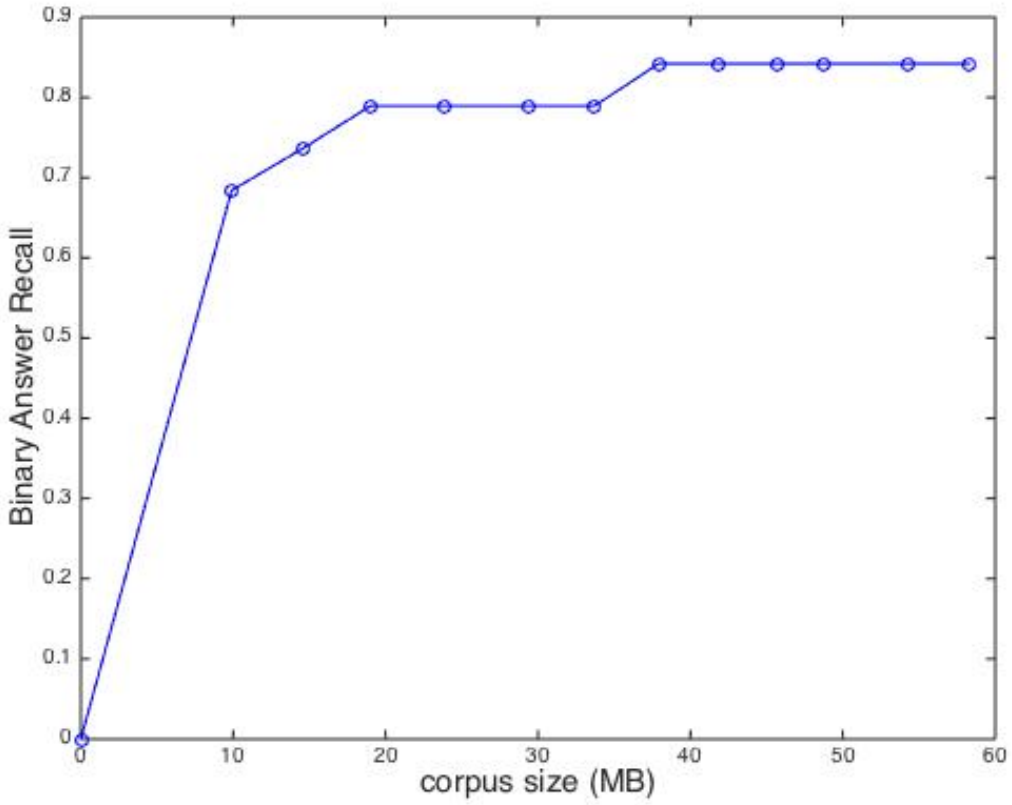
Methods			Search Engine	Binary Answer Recall		
Matching	Expansion	Corpus		Training	Development	Test
Phrases	One-time	Snippet	Bing	0.9355	0.7894	0.7333
Phrases	Tf-idf Iteration	Snippet	Bing	0.9516	0.8421	0.8000
Phrases	One-time	Snippet & Content	Bing	0.9677	0.8947	0.8000
Phrases	Tf-idf	Snippet	Bing	0.9677	0.8947	0.8000

	Iteration	& Content				
Phrases	Name Entity Iteration	Snippet & Content	Bing	0.9677	0.8947	0.8000
Phrases	DBPedia Annotator Iteration	Snippet & Content	Bing DBPedia	0.9677	0.8947	0.8000
Named Entities	Stanford Parser to extract Named Entities, noun and verbs	Snippet & Content	Bing	0.9677	0.8947	0.8000

We conducted 6 experiments by using different expansion methods and query extract algorithms. We use the exact string matching as our matching method (match the exact answer string in the corpus). Our baseline is one-time source expansion while only get the snippet from the search engine. The second experiment shows that iterative source expansion perform better than one-time expansion. The third experiment further shows that the content of a web page contains more information. The rest of the experiments use different query extract algorithms. The reason is that the rest of the answers that can not be found in the corpus are long strings, which are hard to be exact matching. Also, there are some date answers while the date format is different from the format explored from search engine.

13. Memory vs BAR recall

We performed an experiment where we observed the manner in which our BAR scores get affected as we increase the permissible corpus size(The number of top documents retrieved from the search engine after the query search), and below is the graph explaining the same.



As we can clearly see above, our BAR scores do not necessarily increase with the corpus size. They increase to a certain extent and then remain consistent, even if we increase the corpus size, hence we can actually leverage the study of this pattern to decide when to stop expanding.

14. Experiment with Watson

We used our proposed approach to build a relevant corpus and sent it to the Watson CMU instance, to get the evidence lists from Watson. We then used these evidence lists to evaluate our system. We get a BAR score of 0.7368 on the development set and 0.7333 on the test set.

15. Conclusion

We proposed an iterative approach for source expansion (SE) and Machine Reading and implemented an end-to-end system that builds a concise and relevant corpus with related information from large, external text corpora.

The source expansion approach was applied to the question answering (QA) task, and its impact on the performance of Watson QA system.

We have designed and implemented the basic framework for the MR process and also completed full end to end run on Watson as proof of concepts. Our approach achieved a BAR score of 0.7 and we are optimistic that our approach has the potential to improve the performance of any QA system.

The link to our github repository is : https://github.com/oaqa/machine-reading-for-watson

16. Next steps

- First, we can do some analysis of the initial questions, such as extracting topics.
- Explore more on strategies on extracting keywords from relevant documents.

For MR process, we still need to:

- Design logic for transitioning between states and regulating the whole MR process.
- Work on persistence approaches for the Knowledge Graph from the MR process
- Also, the constraint of the corpus size should be taken into consideration. Lastly, We hope to generalize our approach to work on any generic domain.

17. References:

1. Schlaefer, Nico, et al. "Statistical source expansion for question answering."*Proceedings of the 20th ACM international conference on Information and knowledge management*. ACM, 2011.
2. Wang, Richard C., et al. "Automatic set expansion for list question answering."*Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2008.
3. Das, Dipanjan, and André FT Martins. "A survey on automatic text summarization." *Literature Survey for the Language and Statistics II course at CMU* 4 (2007): 192-195.
4. Lally, Adam, et al. "Question analysis: How Watson reads a clue." *IBM Journal of Research and Development* 56.3.4 (2012): 2-1.