Machine Reading for Question Answering

Kavya, Srivaths, Yepeng, Qiang, Xiaoqiu, Liping

Problem Motivation

- Using all the available data as source for a QA system has the following problems:
  - More data means more noise
  - Irrelevant data might bias answers away from truth.
  - QA systems might perform better on filtered and enhanced data specific to the domain.
  - Difficult to use in memory constrained situations

All documents from the web

QA system

?

Domain specific Questions and Answers

Machine Reading

Domain specific documents

QA system

answers

Problem Definition

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

Goals for this semester

- Design a workflow for the Machine Reading process.
- Implement the basic framework based on the workflow.
- Full end to end run on Watson as proof of concept.

What is Machine Reading?

- Machine Reading is an automatic, unsupervised text understanding.
- Automatically analyze the questions and answers to “read” documents from the data sources to build a relevant corpus that caters specifically to the domain:
  - Retrieve relevant documents from search engines
  - Build a summary corpus using the retrieved documents
  - Get sets of important concepts and update the corpus.
  - Do this as iteratively to until we have ‘enough’ information about the domain

Related Work

- Perform source expansion by extracting text nuggets from text corpus with one iteration(Schlaefer, Nico, et al.)

  Inspiration: the basic framework of our project
- Extract a list of NE to produce another list of answers and combine with original(Wang, Richard C., et al.)

  Inspiration: after getting the pseudo document, we can extract the entity and do next expansion.

Related Work

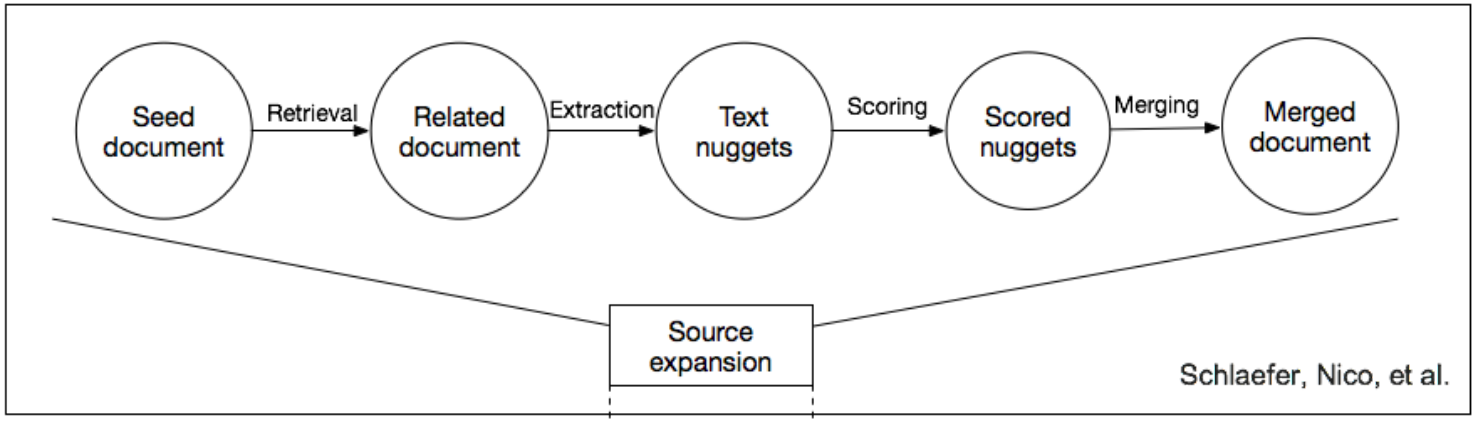
- Summarize and extract information from single / multiple documents, using standard ML approaches(Das, Dipanjan, et al.)

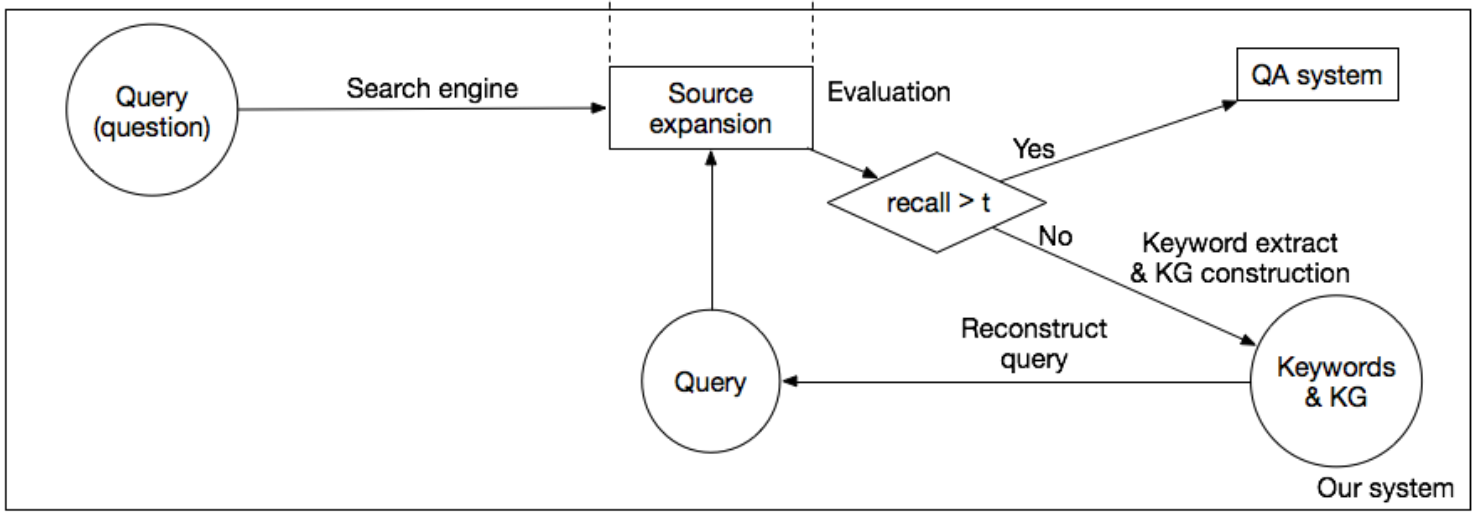
  Inspiration: how to score the sentence in the pseudo document with machine learning algorithm, such as decision tree

- Analyze question first to answer with high confidence(Lally, Adam, et al.)

  Inspiration: analyze the question before doing the source expansion

Related Work





Motivation for Knowledge Graph

- Schlaefer, Nico, et al. performs source expansion by expanding a set of documents to get a larger corpus (1 iteration!)
- We build on top of it, perform multiple iterations to iteratively enhance the corpus.
- Data needs to be stored at every stage of the MR process in a structure.
- An enhanced corpus needs to understand the inherent relationships in data (entities).
- Above mentioned requirements lead to the structure of knowledge graph to store data internally.


Mona Lisa
Da Vinci
Italy
Michelangelo
Date of birth: April 15, 1452
Date of death: May 2, 1519
(age 67 years)

Vision

- Develop algorithms for machine reading.
- Feed more relevant and concise content to the QA system to help improve performance on QA tasks.
- Improve the structure and quality of the corpus by constructing a Knowledge Graph internally that represents data.
- Explore application of MR algorithm for IR tasks requiring high recall [patent/legal document search]

Scope of prototype

- Input: A set of questions and corresponding answers
- Output: A domain specific corpus based on the input.
- Measurement: Recall of the results from Watson QA system using Machine Reading Vs Recall achieved using just the Bing/Wikipedia top results.
- Size: Explored corpus size should be bounded by some function of number of questions
- Time: No bound on computation time
- Data sources : Bing, DBpedia

Ideas from Manual Simulation

1. Process questions together rather than one by one (domain knowledge).
2. Reformulate long queries.
3. Use related search.
4. Select the top document (preferably Wiki page).
5. Choose answer sentence:
    a. (Percentage of overlap, NE and head words, synonyms of relations).
6. If none, expand using sentences that have one of the NE.
7. Extract NE and embedded hyperlink entities from these as queries for expansion.
8. Ensure recall (keep a window of context around answer sentence).

Approach

Components of Framework

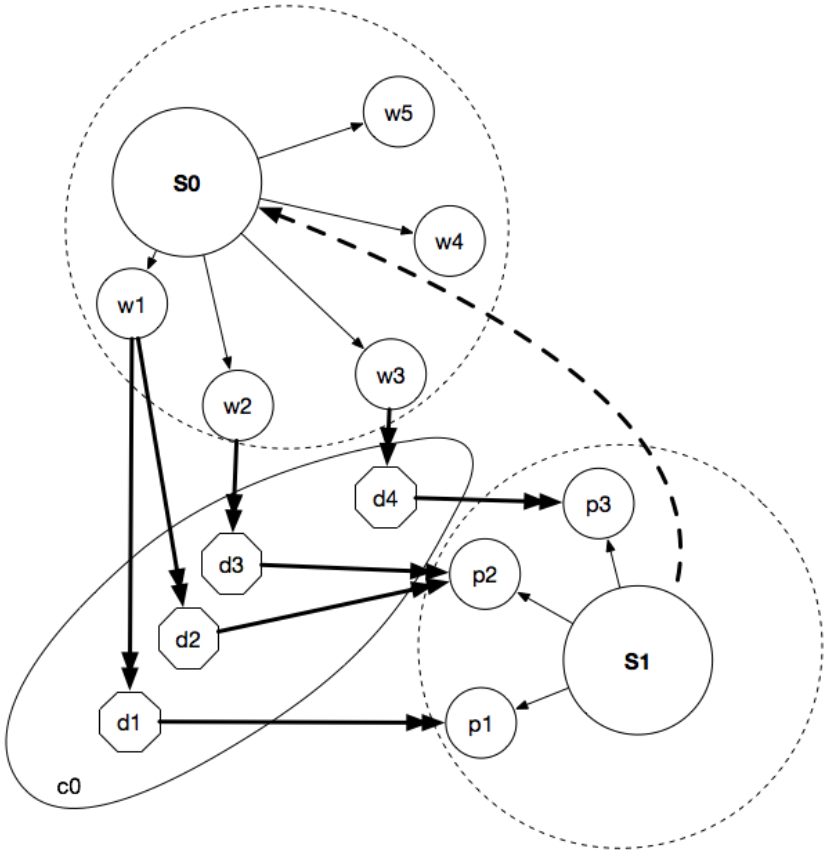
- Machine reading process is modeled as states and transitions
- Each state is defined by the set of seeds, a corpus and a snapshot of the knowledge graph.
- **Seed -** a phrase with a priority weight and features:
  - Counts in question set, corpus; similarity of the article page for this seed/entity with the question+answer set, set of neighbor seeds, etc.
- Weight is a priority score that is used while searching the seeds.
  - Weight is [currently] a heuristic function of the features mentioned above

Approach

Actions of Framework

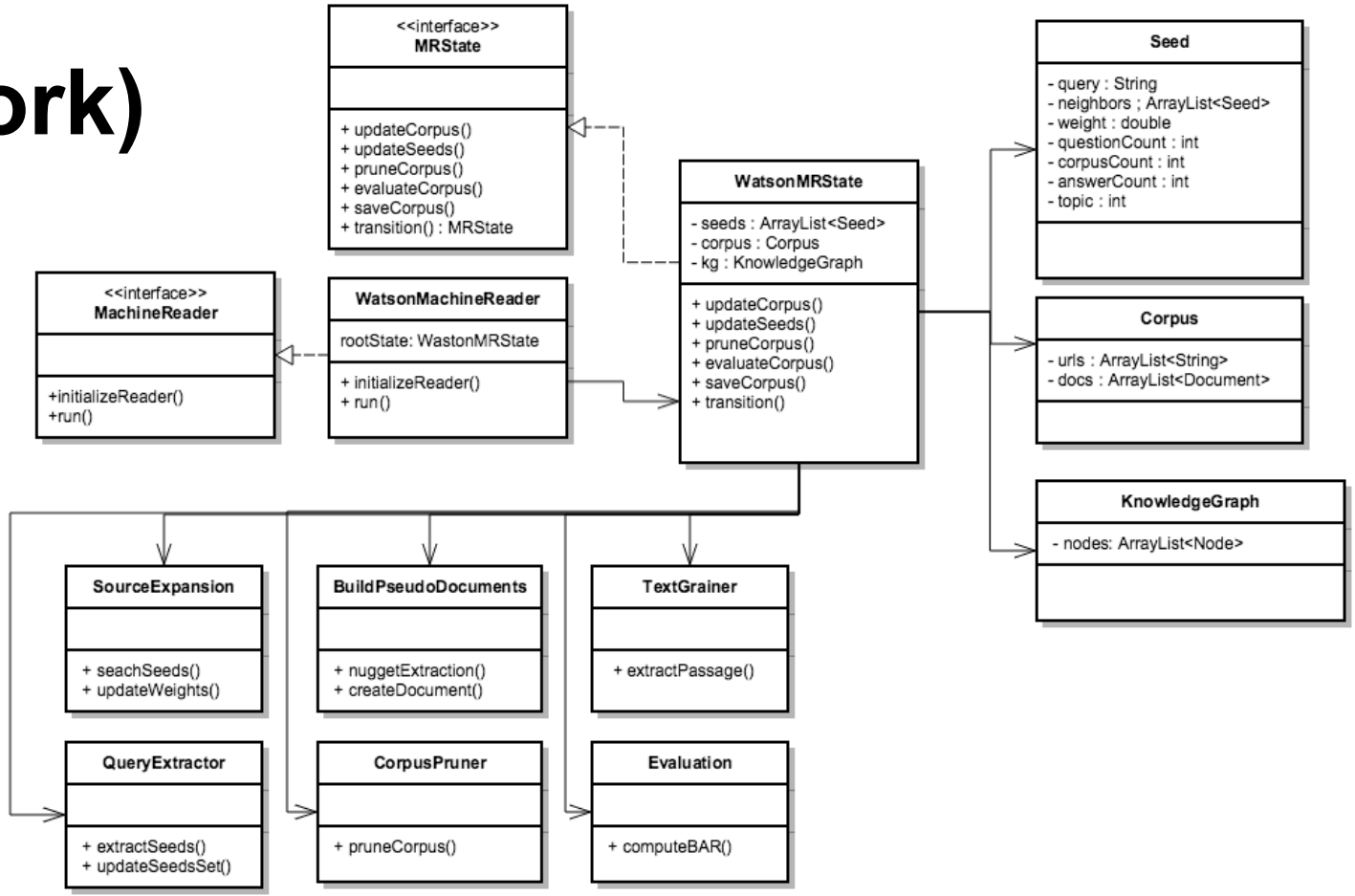
- Transitions are based on actions
  - Update the corpus: by searching from seeds and adding documents/snippets to the corpus
  - Update the set of seeds: By extracting seeds from the corpus
  - Prune the corpus
- During one action of updating the corpus:
  - Based on weight, search using a seed and retrieve next unretrieved document.
  - No exploration of that seed if weight below threshold.
  - At the end, update weights for all the seeds.

State and transitions



s0, s1	State
w1-w5, p1-p3	Seeds [phrases]
d1-d4	Document
c0	Corpus

Design (Framework)



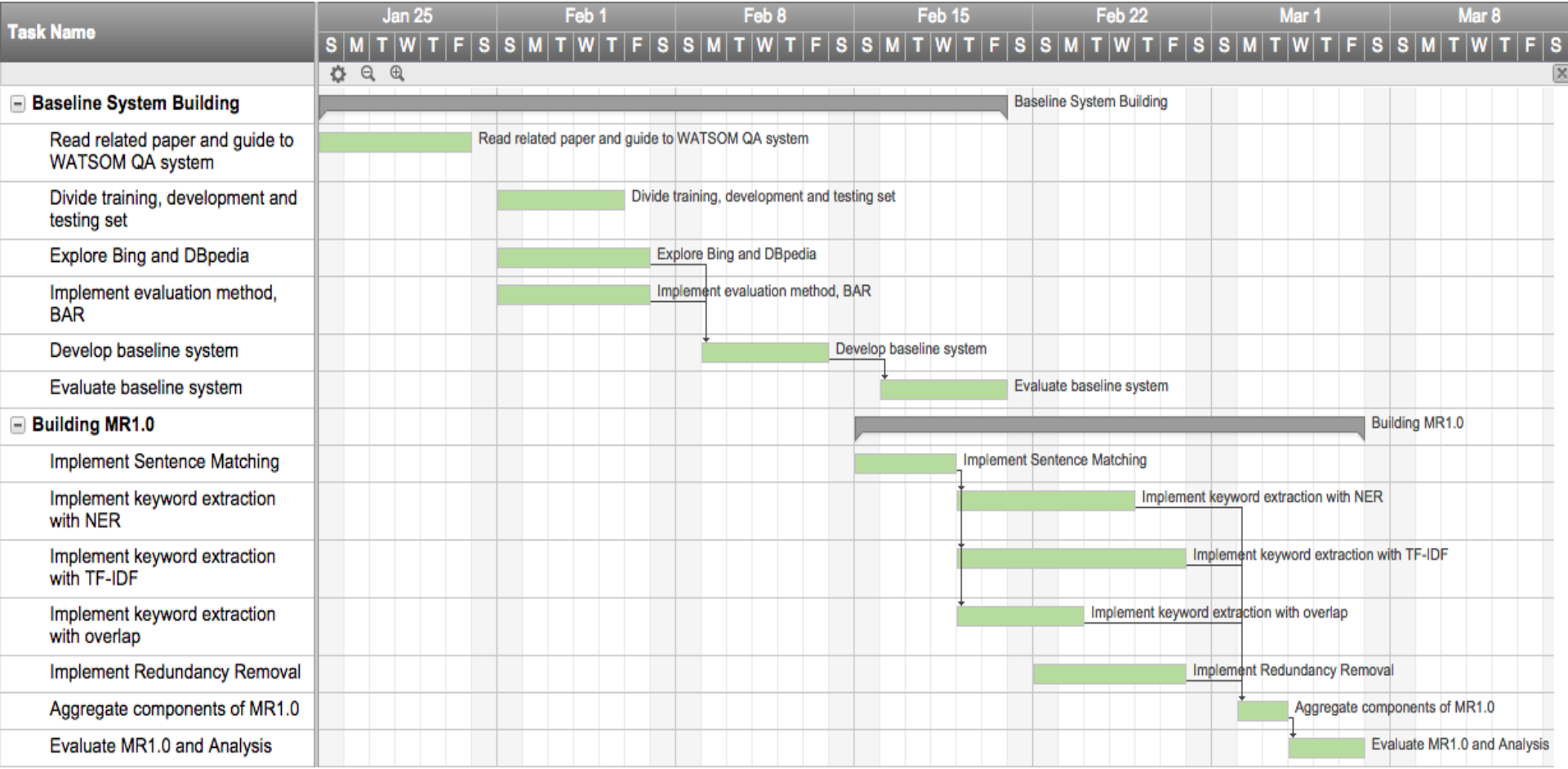
Plan

1. Implement different algorithms for each of the components on the framework we have developed and evaluate them on accuracy and memory of corpus.
2. Test on easier-to-use QA system, since evaluation on Watson cannot be automated.
3. Evaluate our system using text collection from the TREC Total Recall task.

Qiang

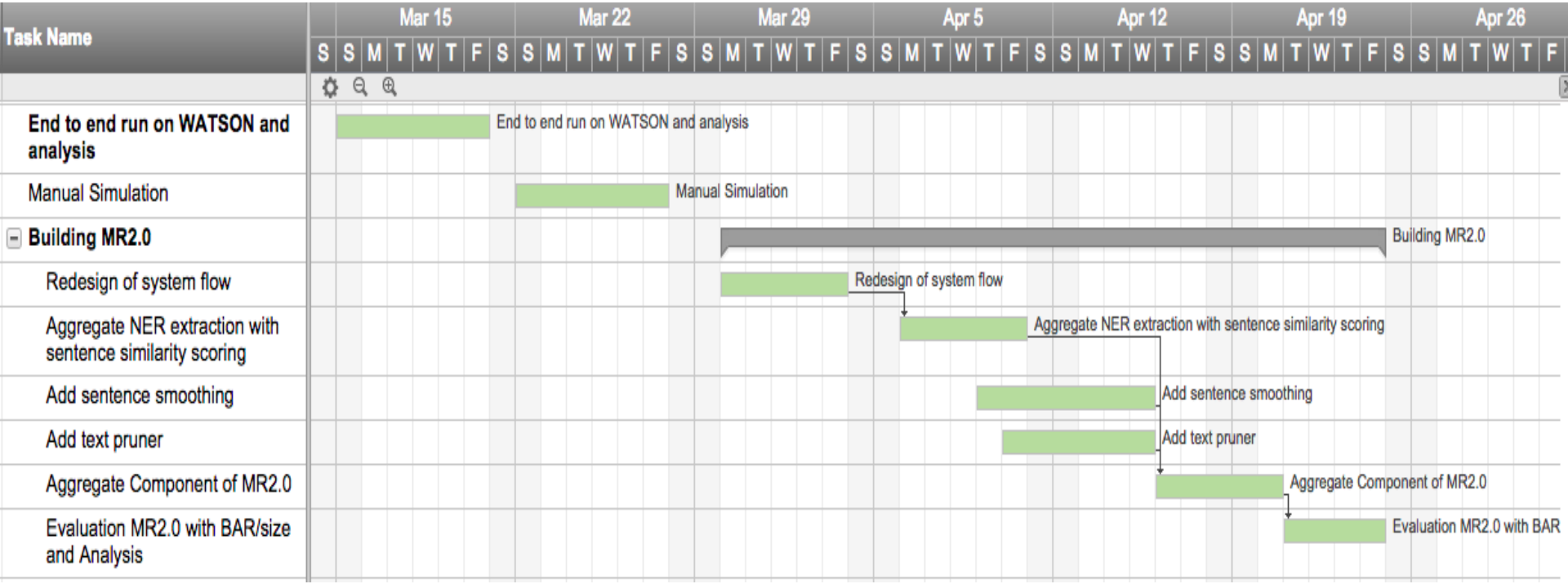
Trec Total Recall

- TREC - Text retrieval conference
- Trec total Recall: with as little human effort as possible, as nearly ***all*** of the relevant documents as possible.
- Useful in cases like search for publications or search in legal domains where we don’t want to miss any relevant documents.
- The corpus generated by the Machine Reading process attempts to have the same properties as above


Task Name
Jan 25
Feb 1
Feb 8
Feb 15
Feb 22
Mar 1
Mar 8
S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S
Baseline System Building
Read related paper and guide to WATSOM QA system
Divide training, development and testing set
Explore Bing and DBpedia
Implement evaluation method, BAR
Develop baseline system
Evaluate baseline system
Building MR1.0
Implement Sentence Matching
Implement keyword extraction with NER
Implement keyword extraction with TF-IDF
Implement keyword extraction with overlap
Implement Redundancy Removal
Aggregate components of MR1.0
Evaluate MR1.0 and Analysis

Qiang

Plan


Task Name
Mar 15
Mar 22
Mar 29
Apr 5
Apr 12
Apr 19
Apr 26
S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F
End to end run on WATSON and analysis
End to end run on WATSON and analysis
Manual Simulation
Manual Simulation
Building MR2.0
Building MR2.0
Redesign of system flow
Redesign of system flow
Aggregate NER extraction with sentence similarity scoring
Aggregate NER extraction with sentence similarity scoring
Add sentence smoothing
Add sentence smoothing
Add text pruner
Add text pruner
Aggregate Component of MR2.0
Aggregate Component of MR2.0
Evaluation MR2.0 with BAR/size and Analysis
Evaluation MR2.0 with BAR

Qiang

Dataset & Data sources

- Dataset
  - Terrorism dataset.
  - Total 96 questions and gold standard answers to these.
  - Training set - 62 (questions and gold standard answers)
  - Development set - 19
  - Test set -15
  - All question types divided evenly
- Datasources
  - Bing
  - DBPedia

Evaluation metric

We evaluated the result based on **Binary Answer Recall** (BAR) and Fixed corpus size.

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

where

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

N is total # of questions

Preliminary Results

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

Method			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 (+1.72%)	0.8421 (+6.68%)	0.8000 (+6.67%)
Phrases	One-time	Snippet + Content	Bing	0.9677 (+ 3.44%)	0.8947 (+13.34%)	0.8000 (+6.67%)

Configuration:

Corpus size = 10 documents / query

Total Corpus size in MB = 48.7MB

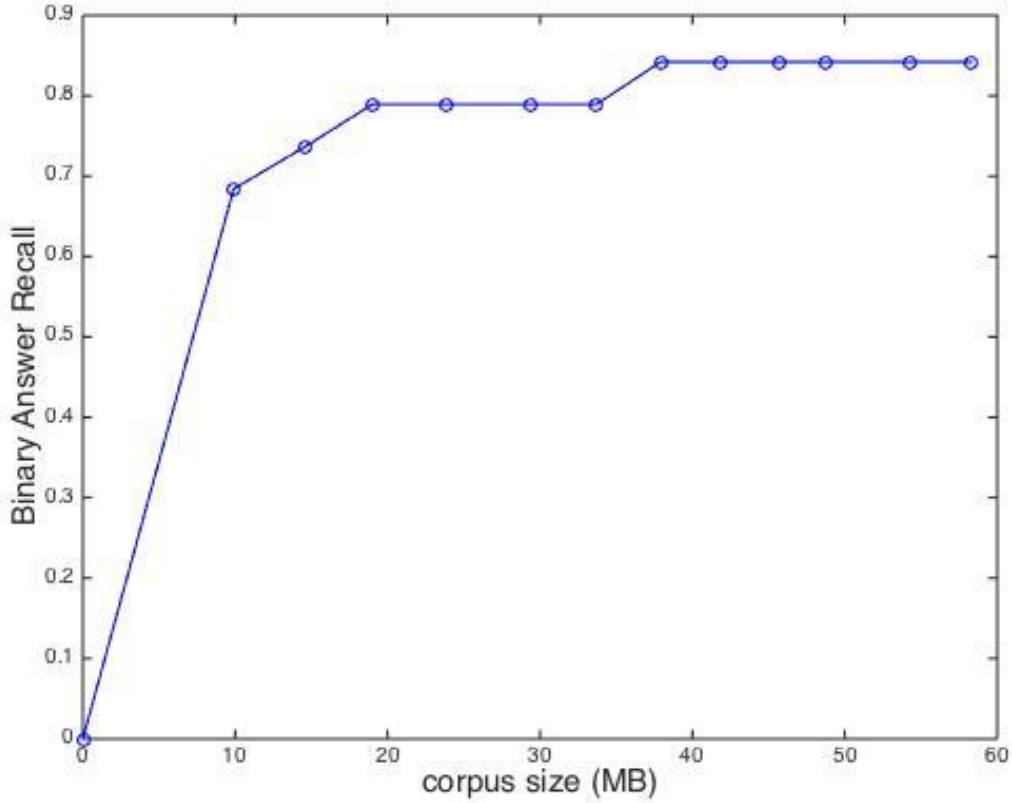
Snippet: Brief description of a document

Content: Web page of a document

Preliminary Results Cont'd

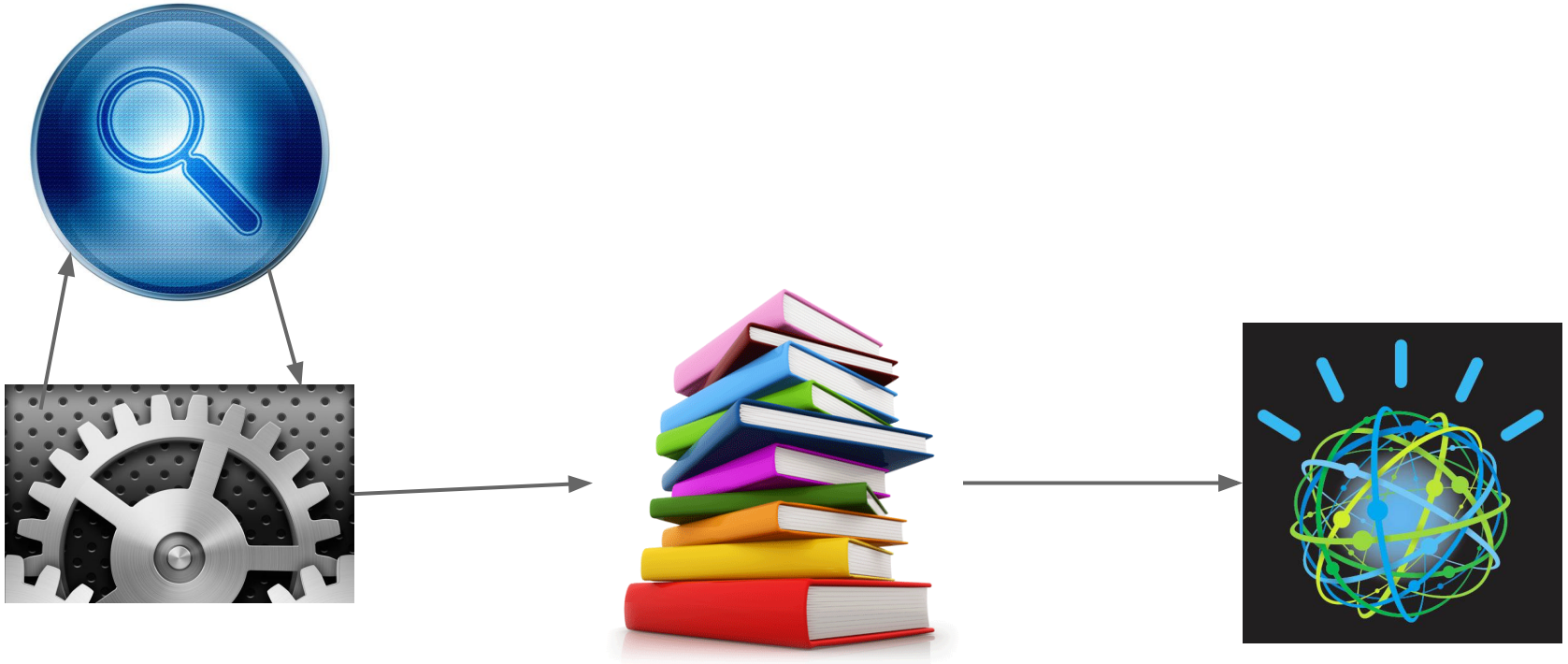
Method			Search Engine	Binary Answer Recall		
Matching	Expansion	Corpus		Training	Development	Test
Phrases	Tf-idf Iteration	Snippet +Content	Bing	0.9677	0.8947	0.8000
Phrases	Named Entity Iteration	Snippet +Content	Bing	0.9677	0.8947	0.8000
Phrases	DBPedia Annotator Iteration	Snippet +Content	Bing DBPedia	0.9677	0.8947	0.8000

Memory vs BAR



WATSON in Jeopardy

Experiment with Watson



Configure and Train Watson

Manage Corpus

Identify and upload documents for Watson to use.

GO

Learn more ›

Train Watson

Teach Watson how to understand your documents.

GO

Learn more ›

Test

Check that your corpus is deployed.

GO

Experiment with Watson

Using our approach we did an end to end run with Watson:

Use a set of questions and answers to get one corpus using our MR system

Feed the corpus to watson, are represented the corpus as HTML file, each document is a section of the html.

watson,we some evidence lists. Use these evidence lists to test whether our corpus contains the answers.

Got the following results based on evident lists:

- Development Set(19) BAR score: 0.7368
- Test Set(15) BAR score: 0.7333

~~Goal~~ Conclusion

- Design a workflow for the Machine Reading process.
- Implement the basic framework based on the workflow. Repo: https://github.com/oaqa/machine-reading-for-watson
- Full end to end run on Watson as proof of concept. We achieved a BAR score of 0.7
- We are optimistic that our approach has the potential to improve the performance of any QA system on a domain.

Next Steps

- Design logic for transitioning between states and regulating the whole MR process.
- Work on persistence approaches for the Knowledge Graph from the MR process
- Explore more on strategies on extracting keywords from relevant documents.
- Take the corpus size constraint into consideration.
- Question Analysis.
- Evaluate our approach’s robustness to work on any generic domain.

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.