Milestone 1

Team 08

M1 Enhancements

Annotation Stage:

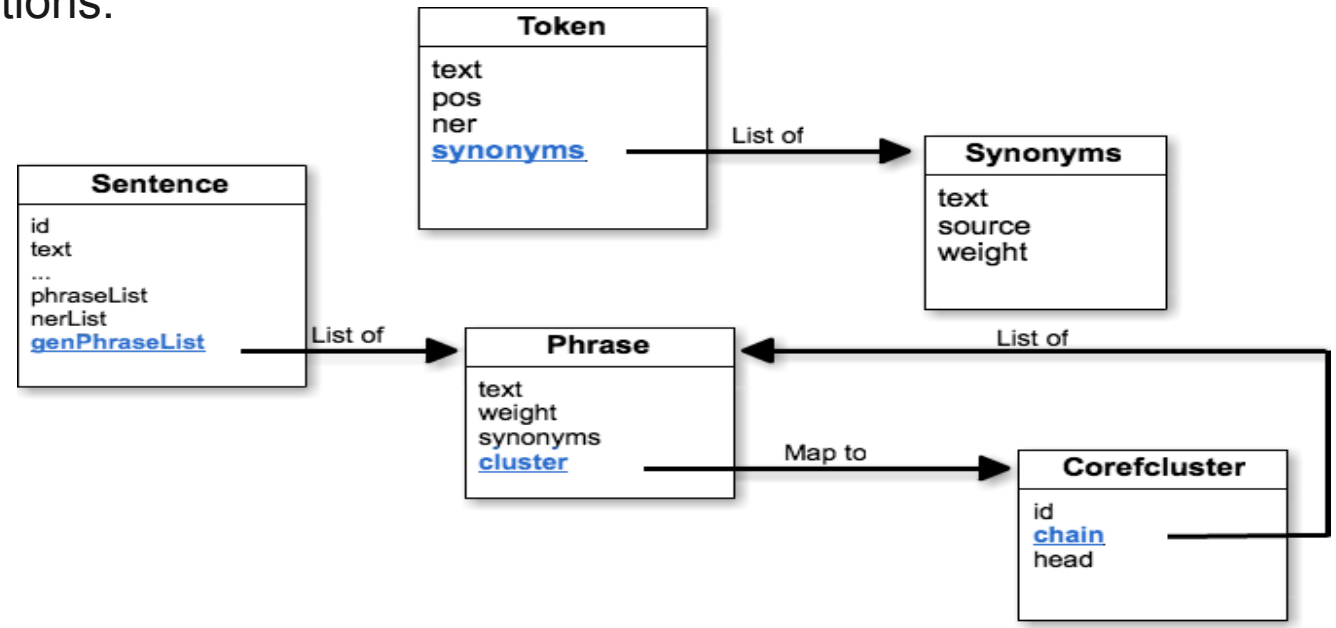
- Coreference Annotator: adds chains of coreference annotations.
- Symbol Expansion: expands symbols to their English counterparts (ex: “Aß” expands to “Abeta”).
- Acronym Expander: expands acronyms to their full text (ex: “AD” expands to “Alzheimer’s Disease”).

Candidate Sentence Searching Stage:

- Expanded Queries:
  - Baseline implementation queried off of text, NER, NP tags.
  - Enhanced version queries off of text, NER, NP, coreferences, symbol expansions, acronym expansions.

Type System

Our type system is based on the baseline system and there are several modifications:



M1 Results

	doc1	doc2	doc3	doc4
PMI	0.22	0.77	0.33	0.2
Co-ref+PMI	0.4	0.5	0.4	0.2

- 11/40 give different choice: 4 Incorrect->Correct; 3 Correct->Incorrect; 4 remain incorrect.
- Our approach can find at least 1 candidate sentence for each question, so we answer all the questions. (3 questions are not answered in baseline system, because it doesn't find any candidate sentences.)

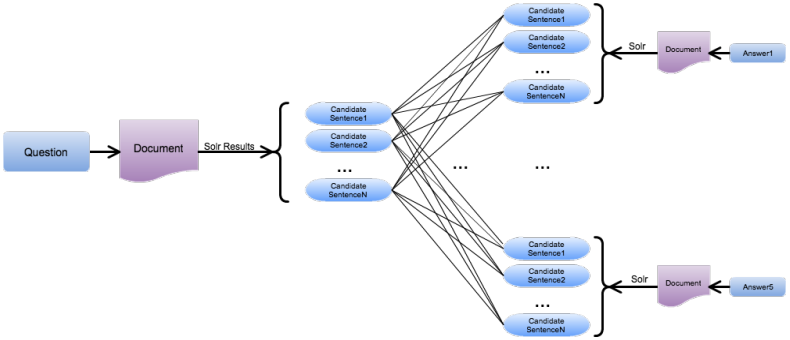
Error Analysis - Takeaways

- Incorrect->Correct:
  - Enhanced search leads to more relevant documents. ‘null’ candidate sentences too are populated (e.g. D3Q3).
- Correct->Incorrect:
  - More noise is added due to query expansion and potentially incorrect coreference. Correct answers lose out only slightly in most of the cases.
- Incorrect->Incorrect:
  - Need an algorithm to eliminate obviously wrong answers for certain question types (“How many...”, “When...”). NP-based PMI (for answer scoring) does not cover answers which are adjective-based (e.g. “longer”, “shorter”).

Next Steps: Milestone 2

Pre-Processing Stage:

- Co-Occurrence Vectors



Candidate Sentence Searching Stage:

- “Answer Query”: pulls back sentences based upon keywords in each answer; we will then perform distance calculations between sentences returned by the “Question Query” and the “Answer Queries”.
- “Question+Answer Query”: for each possible answer, we will combine the question terms with the answer terms to create a question+answer query.

Next Steps: Milestone 2 (cont'd)

Answer Choice Scoring

- Expansion on the Answers: Find candidate sentences for each answers
- Design our own scoring function to take the co-occurrence, sentence distance and synonyms overlap into consideration.

Answer Choice Selection:

- Heuristics to eliminate candidate answers (based on question type, answer type)

Structural:

- TBD: add parallel pipelines based on question type (i.e. a factoid pipeline, a "causal" question pipeline, etc). This will take place after other activities, and will require further error analysis.
- TBD: use CSE and/or ML to optimize parameter settings of different stages.