Passage Retrieval and Re-ranking

Ling573
NLP Systems and Applications
May 3, 2011

Upcoming Talks

- Edith Law
  - Friday: 3:30; CSE 303
  - **Human Computation: Core Research Questions and Opportunities**
    - Games with a purpose, MTurk , Captcha verification, etc

- Benjamin Grosof: Vulcan Inc., Seattle, WA, USA
  - Weds 4pm; LIL group, AI lab
  - **SILK's Expressive Semantic Web Rules and Challenges in Natural Language Processing**

Roadmap

- Passage retrieval and re-ranking
  - Quantitative analysis of heuristic methods
    - Tellex et al 2003
  - Approaches, evaluation, issues
  - Shallow processing learning approach
    - Ramakrishnan et al 2004
  - Syntactic structure and answer types
    - Aktolga et al 2011
  - QA dependency alignment, answer type filtering

Passage Ranking

- Goal: Select passages most likely to contain answer
- Factors in reranking:
  - Document rank
  - Want answers!
    - Answer type matching
      - Restricted Named Entity Recognition
  - Question match:
    - Question term overlap
    - **Span** overlap: N-gram, longest common sub-span
    - Query term **density:** short spans w/more qterms

Quantitative Evaluation of Passage Retrieval for QA

- Tellex et al.
- Compare alternative passage ranking approaches
  - 8 different strategies + voting ranker
- Assess interaction with document retrieval

Comparative IR Systems

- PRISE
  - Developed at NIST
  - Vector Space retrieval system
  - Optimized weighting scheme

Comparative IR Systems

- PRISE
  - Developed at NIST
  - Vector Space retrieval system
  - Optimized weighting scheme
- Lucene
  - Boolean + Vector Space retrieval
  - Results Boolean retrieval RANKED by tf-idf
    - Little control over hit list

Comparative IR Systems

- PRISE
  - Developed at NIST
  - Vector Space retrieval system
  - Optimized weighting scheme
- Lucene
  - Boolean + Vector Space retrieval
  - Results Boolean retrieval RANKED by tf-idf
    - Little control over hit list
- Oracle: NIST-provided list of relevant documents

Comparing Passage Retrieval

- Eight different systems used in QA
  - Units
  - Factors

Comparing Passage Retrieval

- Eight different systems used in QA
  - Units
  - Factors
- MITRE:
  - Simplest reasonable approach: baseline
  - Unit: sentence
  - Factor: Term overlap count

Comparing Passage Retrieval

- Eight different systems used in QA
  - Units
  - Factors
- MITRE:
  - Simplest reasonable approach: baseline
  - Unit: sentence
  - Factor: Term overlap count
- MITRE+stemming:
  - Factor: stemmed term overlap

Comparing Passage Retrieval

- Okapi bm25
  - Unit: fixed width sliding window
  - Factor: $$Score(q,d) = \sum_{i=1}^{N} idf(q_i) \frac{tf_{q_i,d}(k_1 + 1)}{tf_{q_i,d} + k_1(1 - b + (b * \frac{|D|}{avgdl}))}$$
    - k1=2.0; b=0.75

Comparing Passage Retrieval

- Okapi bm25
  - Unit: fixed width sliding window
  - Factor: $$Score(q,d) = \sum_{i=1}^{N} idf(q_i) \frac{tf_{q_i,d}(k_1+1)}{tf_{q_i,d} + k_1(1-b+(b*\frac{|D|}{avgdl}))}$$
    - k1=2.0; b=0.75
- MultiText:
  - Unit: Window starting and ending with query term
  - Factor:
    - Sum of IDFs of matching query terms
    - Length based measure * Number of matching terms

Comparing Passage Retrieval

- IBM:
  - Fixed passage length
  - Sum of:
    - Matching words measure: Sum of idfs of overlap terms
    - Thesaurus match measure:
      - Sum of idfs of question wds with synonyms in document
    - Mis-match words measure:
      - Sum of idfs of questions wds NOT in document
    - Dispersion measure: # words b/t matching query terms
    - Cluster word measure: longest common substring

Comparing Passage Retrieval

- SiteQ:
  - Unit: n (=3) sentences
  - Factor: Match words by literal, stem, or WordNet syn
    - Sum of
      - Sum of idfs of matched terms
      - Density weight score * overlap count, where

Comparing Passage Retrieval

- SiteQ:
  - Unit: n (=3) sentences
  - Factor: Match words by literal, stem, or WordNet syn
    - Sum of
      - Sum of idfs of matched terms
      - Density weight score * overlap count, where

$$dw(q,d) = \frac{\sum_{j=1}^{k-1} \frac{idf(q_j) + idf(q_{j+1})}{\alpha \times dist(j, j+1)^2}}{k-1} \times overlap$$

Comparing Passage Retrieval

- Alicante:
  - Unit: n (= 6) sentences
  - Factor: non-length normalized cosine similarity

Comparing Passage Retrieval

- Alicante:
  - Unit: n (= 6) sentences
  - Factor: non-length normalized cosine similarity

- ISI:
  - Unit: sentence
  - Factors: weighted sum of
    - Proper name match, query term match, stemmed match

Experiments

- Retrieval:
  - PRISE:
    - Query: Verbatim question
  - Lucene:
    - Query: Conjunctive boolean query (stopped)

Experiments

- Retrieval:
  - PRISE:
    - Query: Verbatim quesiton
  - Lucene:
    - Query: Conjunctive boolean query (stopped)
- Passage retrieval: 1000 word passages
  - Uses top 200 retrieved docs
  - Find best passage in each doc
  - Return up to 20 passages
    - Ignores original doc rank, retrieval score

Pattern Matching

- Litkowski pattern files:
  - Derived from NIST relevance judgments on systems
  - Format:
    - Qid answer_pattern doc_list
      - Passage where answer_pattern matches is correct
        - If it appears in one of the documents in the list

Pattern Matching

- Litkowski pattern files:
  - Derived from NIST relevance judgments on systems
  - Format:
    - Qid answer_pattern doc_list
      - Passage where answer_pattern matches is correct
        - If it appears in one of the documents in the list
- MRR scoring
  - Strict: Matching pattern in official document
  - Lenient: Matching pattern

Examples

- Example
  - Patterns
    - 1894 (190|249|416|440)(\s|\-)million(\s|\-)miles? APW19980705.0043 NYT19990923.0315 NYT19990923.0365 NYT20000131.0402 NYT19981212.0029
    - 1894 700-million-kilometer APW19980705.0043
    - 1894 416 - million - mile NYT19981211.0308
  - Ranked list of answer passages
    - 1894 0 APW19980601.0000 the casta way weas
    - 1894 0 APW19980601.0000 440 million miles
    - 1894 0 APW19980705.0043 440 million miles

Evaluation

- MRR
  - Strict and lenient
- Percentage of questions with NO correct answers

Evaluation

- MRR
  - Strict: Matching pattern in official document
  - Lenient: Matching pattern
- Percentage of questions with NO correct answers

	Strict				
	Lucene		PRISE		TREC
Algorithm	MRR	% Inc.	MRR	% Inc.	% Inc.
IBM	0.326	49.20%	0.331	39.60%	44.3%
ISI	0.329	48.80%	0.287	41.80%	41.7%
SiteQ	0.323	48.00%	0.358	40.40%	56.1%
MultiText	0.354	46.40%	0.325	41.60%	43.1%
Alicante	0.296	50.00%	0.321	42.60%	60.4%
bm25	0.312	48.80%	0.252	46.00%	n/a
stemmed MITRE	0.250	52.60%	0.242	58.60%	n/a
MITRE	0.271	49.40%	0.189	52.00%	n/a
Averages	0.309	49.15%	0.297	45.33%	n/a
Voting with IBM, ISI, SiteQ	0.350	39.80%	0.352	39.00%	n/a

Evaluation on Oracle Docs

Algorithm	# Incorrect	% Incorrect	MRR
IBM	31	7.18%	0.851
SiteQ	32	7.41%	0.859
ISI	37	8.56%	0.852
Alicante	39	9.03%	0.816
MultiText	44	10.19%	0.845
bm25	45	10.42%	0.810
MITRE	45	10.42%	0.800
stemmed MITRE	63	14.58%	0.762

Overall

- PRISE:
  - Higher recall, more correct answers

Overall

- PRISE:
  - Higher recall, more correct answers
- Lucene:
  - Higher precision, fewer correct, but higher MRR

Overall

- PRISE:
  - Higher recall, more correct answers
- Lucene:
  - Higher precision, fewer correct, but higher MRR
- Best systems:
  - IBM, ISI, SiteQ
  - Relatively insensitive to retrieval engine

Analysis

- Retrieval:
  - Boolean systems (e.g. Lucene) competitive, good MRR
    - Boolean systems usually worse on ad-hoc

Analysis

- Retrieval:
  - Boolean systems (e.g. Lucene) competitive, good MRR
    - Boolean systems usually worse on ad-hoc
- Passage retrieval:
  - Significant differences for PRISE, Oracle
  - Not significant for Lucene -> boost recall

Analysis

- Retrieval:
  - Boolean systems (e.g. Lucene) competitive, good MRR
    - Boolean systems usually worse on ad-hoc

- Passage retrieval:
  - Significant differences for PRISE, Oracle
  - Not significant for Lucene -> boost recall
- Techniques: Density-based scoring improves
  - Variants: proper name exact, cluster, density score

Error Analysis

- 'What is an ulcer?'

Error Analysis

- ‘What is an ulcer?’
  - After stopping -> ‘ulcer’
  - Match doesn’t help

Error Analysis

- ‘What is an ulcer?’
  - After stopping -> ‘ulcer’
  - Match doesn’t help
  - Need question type!!
- Missing relations
  - ‘What is the highest dam?’
    - Passages match ‘highest’ and ‘dam’ – but not together
  - Include syntax?

Learning Passage Ranking

- Alternative to heuristic similarity measures
- Identify candidate features
- Allow learning algorithm to select

Learning Passage Ranking

- Alternative to heuristic similarity measures
- Identify candidate features
- Allow learning algorithm to select
- Learning and ranking:
  - Employ general classifiers
    - Use score to rank (e.g., SVM, Logistic Regression)

Learning Passage Ranking

- Alternative to heuristic similarity measures
- Identify candidate features
- Allow learning algorithm to select
- Learning and ranking:
  - Employ general classifiers
    - Use score to rank (e.g., SVM, Logistic Regression)
  - Employ explicit rank learner
    - E.g. RankBoost

Shallow Features & Ranking

- Is Question Answering an Acquired Skill?
  - Ramakrishnan et al, 2004
- Full QA system described
  - Shallow processing techniques
  - Integration of Off-the-shelf components
  - Focus on rule-learning vs hand-crafting
  - Perspective: questions as noisy SQL queries

Architecture

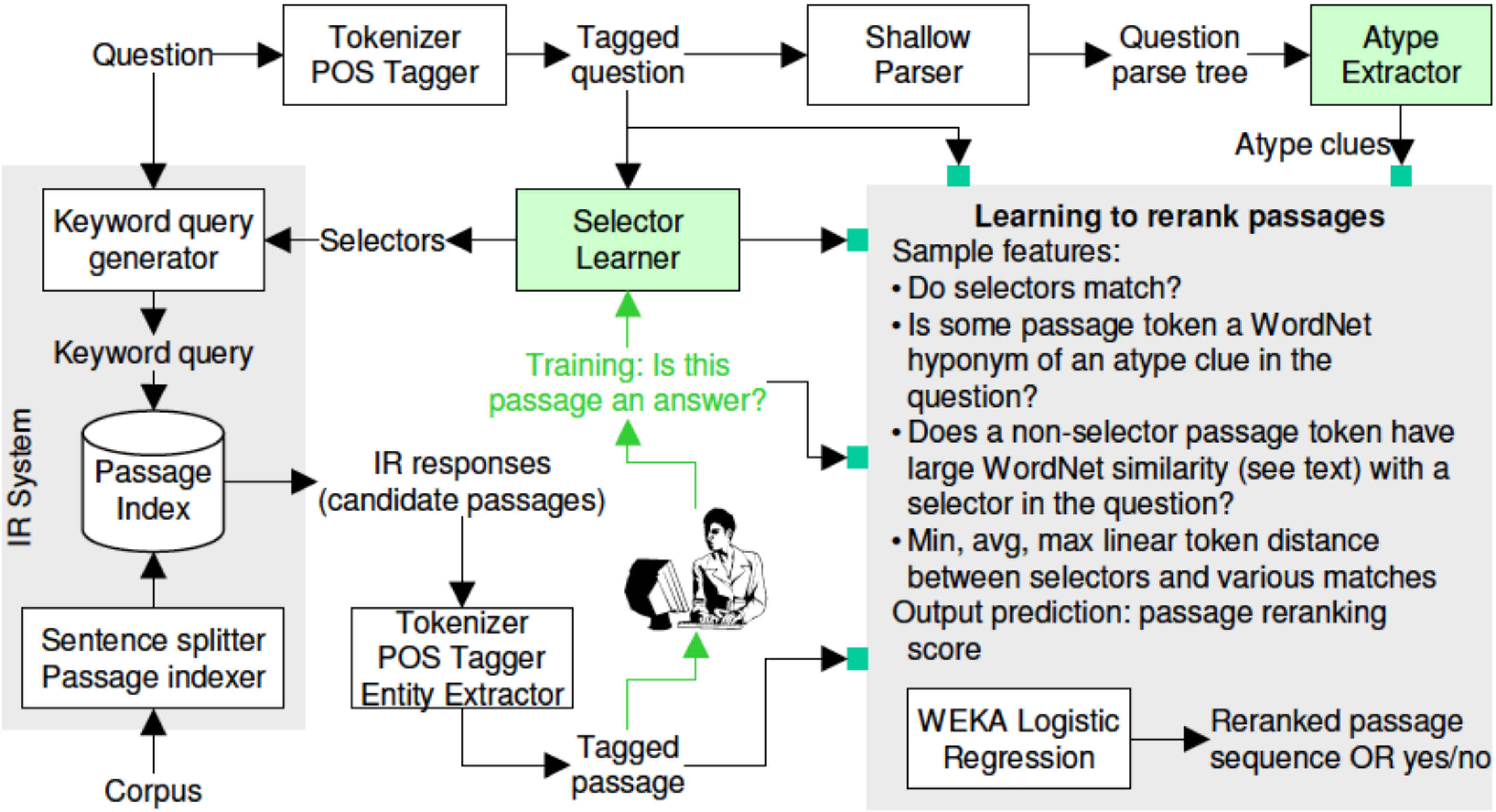


Figure 2: Overall architecture of our trainable QA system.

Basic Processing

- Initial retrieval results:
  - IR ‘documents’:
    - 3 sentence windows (Tellex et al)
  - Indexed in Lucene
  - Retrieved based on reformulated query

Basic Processing

- Initial retrieval results:
  - IR ‘documents’:
    - 3 sentence windows (Tellex et al)
  - Indexed in Lucene
  - Retrieved based on reformulated query
- Question-type classification
  - Based on shallow parsing
  - Synsets or surface patterns

Selectors

- Intuition:
  - ‘Where’ clause in an SQL query – selectors

Selectors

- Intuition:
  - ‘Where’ clause in an SQL query – selectors
  - Portion(s) of query highly likely to appear in answer
- Train system to recognize these terms
  - Best keywords for query
  - *Tokyo is the capital of which country?*
    - Answer probably includes.....

Selectors

- Intuition:
  - ‘Where’ clause in an SQL query – selectors
  - Portion(s) of query highly likely to appear in answer
- Train system to recognize these terms
  - Best keywords for query
  - *Tokyo is the capital of which country?*
    - Answer probably includes.....
      - Tokyo+++
      - Capital+
      - Country?

Selector Recognition

- Local features from query:
  - POS of word
  - POS of previous/following word(s), in window
  - Capitalized?

Selector Recognition

- Local features from query:
  - POS of word
  - POS of previous/following word(s), in window
  - Capitalized?
- Global features of word:
  - Stopword?
  - IDF of word
  - Number of word senses
  - Average number of words per sense

Selector Recognition

- Local features from query:
  - POS of word
  - POS of previous/following word(s), in window
  - Capitalized?
- Global features of word:
  - Stopword?
  - IDF of word
  - Number of word senses
  - Average number of words per sense
    - Measures of word specificity/ambiguity

Selector Recognition

- Local features from query:
  - POS of word
  - POS of previous/following word(s), in window
  - Capitalized?
- Global features of word:
  - Stopword?
  - IDF of word
  - Number of word senses
  - Average number of words per sense
    - Measures of word specificity/ambiguity
- Train Decision Tree classifier on gold answers: +/-S

Passage Ranking

- For question q and passage r, in a good passage:

Passage Ranking

- For question q and passage r, in a good passage:
  - All selectors in q appear in r

Passage Ranking

- For question q and passage r, in a good passage:
  - All selectors in q appear in r
  - r has answer zone A w/o selectors

Passage Ranking

- For question q and passage r, in a good passage:
  - All selectors in q appear in r
  - r has answer zone A w/o selectors
  - Distances b/t selectors and answer zone A are small

Passage Ranking

- For question q and passage r, in a good passage:
  - All selectors in q appear in r
  - r has answer zone A w/o selectors
  - Distances b/t selectors and answer zone A are small
  - A has high similarity with question type

Passage Ranking

- For question q and passage r, in a good passage:
  - All selectors in q appear in r
  - r has answer zone A w/o selectors
  - Distances b/t selectors and answer zone A are small
  - A has high similarity with question type
  - Relationship b/t Qtype, A's POS and NE tag (if any)

Passage Ranking Features

- Find candidate answer zone A* as follows for (q.r)
  - Remove all matching q selectors in r
  - For each word (or compound in r) A
    - Compute Hyperpath distance b/t Qtype & A
      - Where HD is Jaccard overlap between hypernyms of Qtype & A

Passage Ranking Features

- Find candidate answer zone A* as follows for (q.r)
  - Remove all matching q selectors in r
  - For each word (or compound in r) A
    - Compute Hyperpath distance b/t Qtype & A
      - Where HD is Jaccard overlap between hypernyms of Qtype & A
- Compute L as set of distances from selectors to A*
- Feature vector:

Passage Ranking Features

- Find candidate answer zone A* as follows for (q.r)
  - Remove all matching q selectors in r
  - For each word (or compound in r) A
    - Compute Hyperpath distance b/t Qtype & A
      - Where HD is Jaccard overlap between hypernyms of Qtype & A
- Compute L as set of distances from selectors to A*
- Feature vector:
  - IR passage rank; HD score; max, mean, min of L

Passage Ranking Features

- Find candidate answer zone A* as follows for (q.r)
  - Remove all matching q selectors in r
  - For each word (or compound in r) A
    - Compute Hyperpath distance b/t Qtype & A
      - Where HD is Jaccard overlap between hypernyms of Qtype & A
- Compute L as set of distances from selectors to A*
- Feature vector:
  - IR passage rank; HD score; max, mean, min of L
  - POS tag of A*; NE tag of A*; Qwords in q

Passage Ranking

- Train logistic regression classifier
  - Positive example:

Passage Ranking

- Train logistic regression classifier
  - Positive example: question + passage with answer
  - Negative example:

Passage Ranking

- Train logistic regression classifier
  - Positive example: question + passage with answer
  - Negative example: question w/any other passage
- Classification:
  - Hard decision: 80% accurate, but

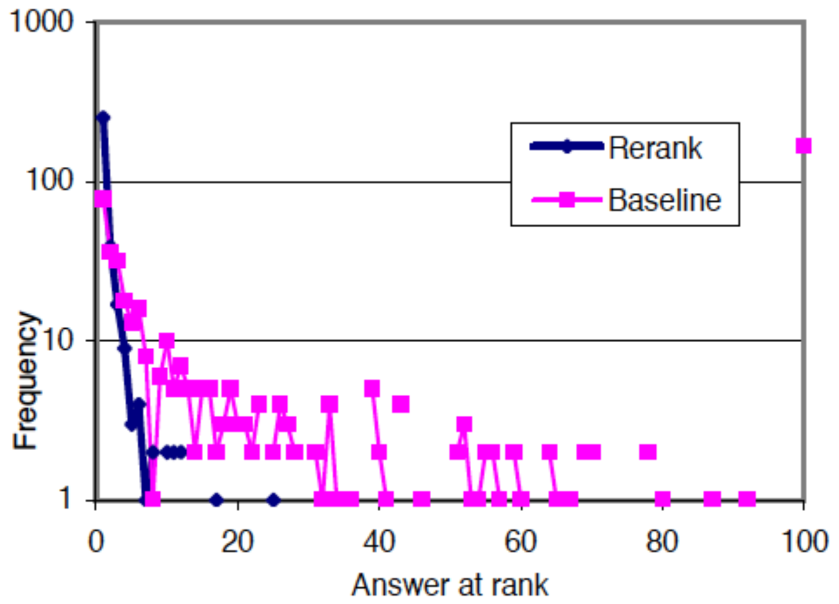
Passage Ranking

- Train logistic regression classifier
  - Positive example: question + passage with answer
  - Negative example: question w/any other passage
- Classification:
  - Hard decision: 80% accurate, but
    - Skewed, most cases negative: poor recall

Passage Ranking

- Train logistic regression classifier
  - Positive example: question + passage with answer
  - Negative example: question w/any other passage
- Classification:
  - Hard decision: 80% accurate, but
    - Skewed, most cases negative: poor recall
- Use regression scores directly to rank

Passage Ranking



igure 9: Reranking significantly improves the rank
f correct passages. The x-axis is the rank at which

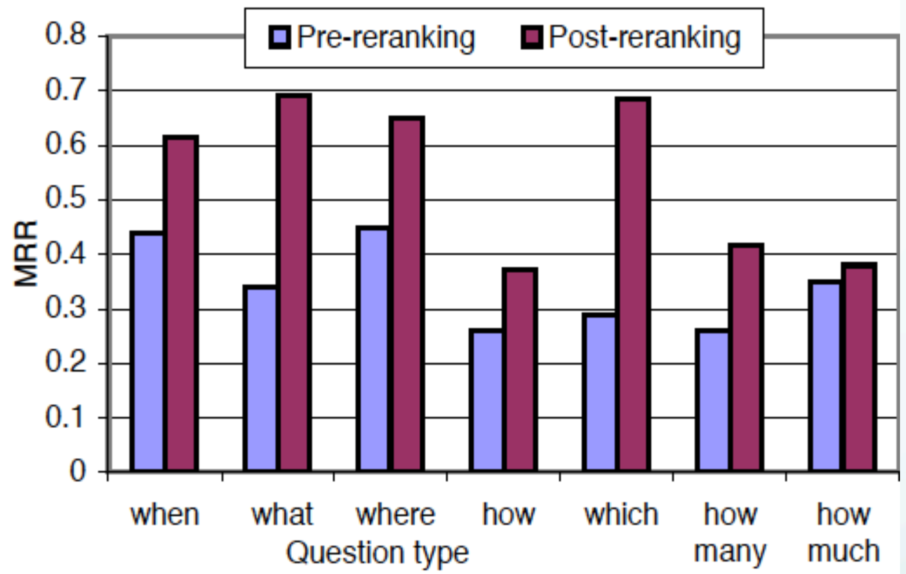


Figure 12: Sample MRR improvement via reranking separated into question categories.

Reranking with Deeper Processing

- Passage Reranking for Question Answering Using Syntactic Structures and Answer Types
  - Atkolga et al, 2011

- Reranking of retrieved passages
  - Integrates
    - Syntactic alignment
    - Answer type
    - Named Entity information

Motivation

- Issues in shallow passage approaches:
  - From Tellex et al.

Motivation

- Issues in shallow passage approaches:
  - From Tellex et al.
    - Retrieval match admits many possible answers
      - Need answer type to restrict

Motivation

- Issues in shallow passage approaches:
  - From Tellex et al.
    - Retrieval match admits many possible answers
      - Need answer type to restrict
    - Question implies particular relations
      - Use syntax to ensure

Motivation

- Issues in shallow passage approaches:
  - From Tellex et al.
    - Retrieval match admits many possible answers
      - Need answer type to restrict
    - Question implies particular relations
      - Use syntax to ensure

  - Joint strategy required
    - Checking syntactic parallelism when no answer, useless
- Current approach incorporates all (plus NER)

Baseline Retrieval

- Bag-of-Words unigram retrieval (BOW)

Baseline Retrieval

- Bag-of-Words unigram retrieval (BOW)

- Question analysis: QuAn
  - ngram retrieval, reformulation

Baseline Retrieval

- Bag-of-Words unigram retrieval (BOW)

- Question analysis: QuAn
  - ngram retrieval, reformulation

- Question analysis + Wordnet: QuAn-Wnet
  - Adds 10 synonyms of ngrams in QuAn

Baseline Retrieval

- Bag-of-Words unigram retrieval (BOW)
- Question analysis: QuAn
  - ngram retrieval, reformulation
- Question analysis + Wordnet: QuAn-Wnet
  - Adds 10 synonyms of ngrams in QuAn
- Best performance: QuAn-Wnet (baseline)

Dependency Information

- Assume dependency parses of questions, passages
  - Passage = sentence
- Extract undirected dependency paths b/t words

Dependency Information

- Assume dependency parses of questions, passages
  - Passage = sentence
- Extract undirected dependency paths b/t words
- Find path pairs between words $(q_k,a_l),(q_r,a_s)$
  - Where q/a words 'match'
    - Word match if a) same root or b) synonyms

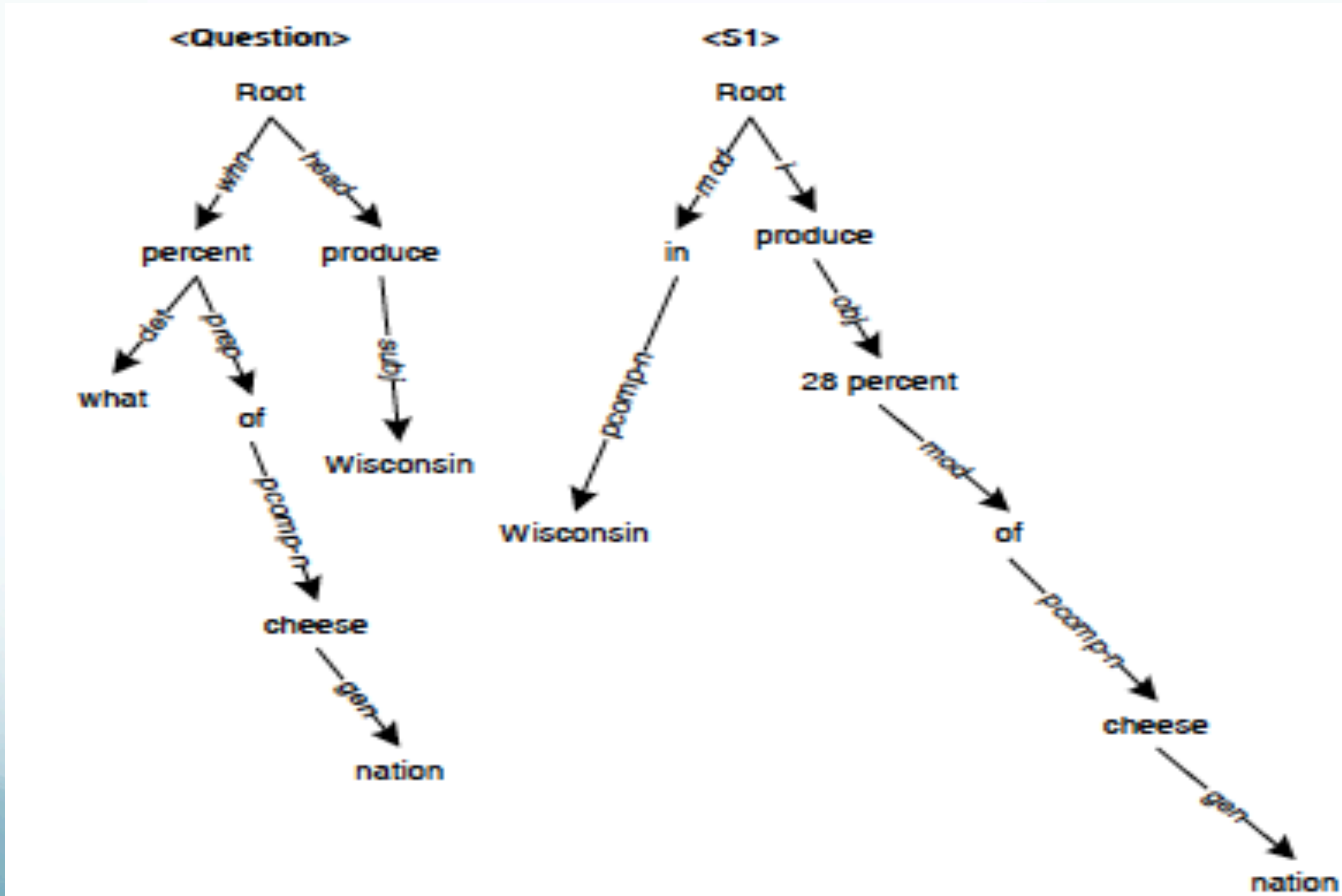
Dependency Information

- Assume dependency parses of questions, passages
  - Passage = sentence
- Extract undirected dependency paths b/t words
- Find path pairs between words $(q_k,a_l),(q_r,a_s)$
  - Where q/a words ‘match’
    - Word match if a) same root or b) synonyms
    - Later: require one pair to be question word/Answer term
- Train path ‘translation pair’ probabilities

Dependency Information

- Assume dependency parses of questions, passages
  - Passage = sentence
- Extract undirected dependency paths b/t words
- Find path pairs between words $(q_k,a_l),(q_r,a_s)$
  - Where q/a words 'match'
    - Word match if a) same root or b) synonyms
    - Later: require one pair to be question word/Answer term
- Train path 'translation pair' probabilities
  - Use true Q/A pairs, $<path_q,path_a>$
  - GIZA++, IBM model 1
    - Yields $Pr(label_a,label_q)$

Dependency Path Similarity



Dependency Path Similarity

Figure 2. Dependency trees for the sample question and sentence S1 in Figure 1 generated by Minipar. Some nodes are omitted due to lack of space.

Question:

Path_ID	Node1	Path	Node2
<P_{Q1}>	Wisconsin	*<subj>*	produce
<P_{Q2}>	produce	*<head, whn, prep, pcomp-n>*	cheese
<P_{Q3}>	nation	<gen>	cheese

S1:

Path_ID	Node1	Path	Node2
<P_{S1}>	Wisconsin	*<pcomp-n, mod, i>*	produce
<P_{S2}>	produce	*<obj, mod, pcomp-n>*	cheese
<P_{S3}>	nation	*<gen>*	cheese

Similarity

- Dependency path matching

Similarity

- Dependency path matching
  - Some paths match exactly
  - Many paths have partial overlap or differ due to question/declarative contrasts

Similarity

- Dependency path matching
  - Some paths match exactly
  - Many paths have partial overlap or differ due to question/declarative contrasts

- Approaches have employed
  - Exact match
  - Fuzzy match
  - Both can improve over baseline retrieval, fuzzy more

Dependency Path Similarity

- Cui et al scoring
- Sum over all possible paths in a QA candidate pair

Dependency Path Similarity

- Cui et al scoring
- Sum over all possible paths in a QA candidate pair

$$\sum_{path_q, path_a \in Paths} scorePair(path_q, path_a)$$

Dependency Path Similarity

- Cui et al scoring
- Sum over all possible paths in a QA candidate pair

$$\sum_{path_q, path_a \in Paths} scorePair(path_q, path_a)$$

$$\frac{1}{|path_a|} \prod_{label_{a_j}} \sum_{label_{q_t}} \Pr(label_{a_j} \mid label_{q_t})$$

Dependency Path Similarity

- Atype-DP
- Restrict first q,a word pair to Qword, ACand
  - Where Acand has correct answer type by NER

Dependency Path Similarity

- Atype-DP
- Restrict first q,a word pair to Qword, ACand
  - Where Acand has correct answer type by NER
- Sum over all possible paths in a QA candidate pair
  - with best answer candidate

Dependency Path Similarity

- Atype-DP
- Restrict first q,a word pair to Qword, ACand
  - Where Acand has correct answer type by NER
- Sum over all possible paths in a QA candidate pair
  - with best answer candidate

$$\max_i \sum_{path_q, path_a \in Paths_{ACand_i}} scorePair(path_q, path_a)$$

Comparisons

- Atype-DP-IP
  - Interpolates DP score with original retrieval score

Comparisons

- Atype-DP-IP
  - Interpolates DP score with original retrieval score
- QuAn-Elim:
  - Acts a passage answer-type filter
  - Excludes any passage w/o correct answer type

Results

- Atype-DP-IP best

Table 2. Evaluation of Reranking Techniques. All results are averages from the testing datasets TREC 2000 and TREC 2001, evaluated on the top 100 retrieved passages.

Model	*MRR@1*	*MRR@5*	*MRR@10*	*MRR@20*	*MRR@50*	*MRR@100*
Q-BOW	0.168	0.266	0.286	0.293	0.299	0.301
QuAn-Wnet	0.193	0.289	0.308	0.319	0.324	0.325
Cui	0.202	0.307	0.325	0.335	0.339	0.341
Atype-DP	0.148	0.24	0.26	0.273	0.279	0.28
Atype-DP-IP	**0.261***	**0.363***	**0.38***	**0.389***	**0.393***	**0.394***
% Improvement over Cui	**+29.2**	+18.24	+16.9	+16.12	+15.9	+15.54
% Improvement over QuAn-Wnet	**+35.2**	+25.6	+23.4	+21.9	+21.3	+ 21.2

Results

- Atype-DP-IP best
  - Raw dependency: 'brittle'; NE failure backs off to IP

Table 2. Evaluation of Reranking Techniques. All results are averages from the testing datasets TREC 2000 and TREC 2001, evaluated on the top 100 retrieved passages.

Model	*MRR@1*	*MRR@5*	*MRR@10*	*MRR@20*	*MRR@50*	*MRR@100*
Q-BOW	0.168	0.266	0.286	0.293	0.299	0.301
QuAn-Wnet	0.193	0.289	0.308	0.319	0.324	0.325
Cui	0.202	0.307	0.325	0.335	0.339	0.341
Atype-DP	0.148	0.24	0.26	0.273	0.279	0.28
Atype-DP-IP	**0.261***	**0.363***	**0.38***	**0.389***	**0.393***	**0.394***
% Improvement over Cui	**+29.2**	+18.24	+16.9	+16.12	+15.9	+15.54
% Improvement over QuAn-Wnet	**+35.2**	+25.6	+23.4	+21.9	+21.3	+ 21.2

Results

- Atype-DP-IP best
  - Raw dependency:'brittle'; NE failure backs off to IP
- QuAn-Elim: NOT significantly worse

Table 2. Evaluation of Reranking Techniques. All results are averages from the testing datasets TREC 2000 and TREC 2001, evaluated on the top 100 retrieved passages.

Model	*MRR@1*	*MRR@5*	*MRR@10*	*MRR@20*	*MRR@50*	*MRR@100*
Q-BOW	0.168	0.266	0.286	0.293	0.299	0.301
QuAn-Wnet	0.193	0.289	0.308	0.319	0.324	0.325
Cui	0.202	0.307	0.325	0.335	0.339	0.341
Atype-DP	0.148	0.24	0.26	0.273	0.279	0.28
Atype-DP-IP	**0.261***	**0.363***	**0.38***	**0.389***	**0.393***	**0.394***
% Improvement over Cui	**+29.2**	+18.24	+16.9	+16.12	+15.9	+15.54
% Improvement over QuAn-Wnet	**+35.2**	+25.6	+23.4	+21.9	+21.3	+ 21.2