Question Processing: Formulation & Expansion

Ling573
NLP Systems and Applications
May 8, 2014
Roadmap

- Query processing
- Query reformulation
- Query expansion
  - WordNet-based expansion
  - Stemming vs morphological expansion
  - Machine translation & paraphrasing for expansion
Deeper Processing for Query Formulation

- **MULDER (Kwok, Etzioni, & Weld)**
- Converts question to multiple search queries
  - Forms which match target
  - Vary specificity of query
    - Most general bag of keywords
    - Most specific partial/full phrases
  - Generates 4 query forms on average
- Employs full parsing augmented with morphology
Question Parsing

- Creates full syntactic analysis of question
  - Maximum Entropy Inspired (MEI) parser
    - Trained on WSJ

- Challenge: Unknown words
  - Parser has limited vocabulary
    - Uses guessing strategy
      - Bad: “tungsten” ➔ number

- Solution:
  - Augment with morphological analysis: PC-Kimmo
  - If PC-KIMMO fails? Guess Noun
Syntax for Query Formulation

- Parse-based transformations:
  - Applies transformational grammar rules to questions
  - Example rules:
    - Subject-auxiliary movement:
      - Q: Who was the first American in space?
      - Alt: was the first American...; the first American in space was
    - Subject-verb movement:
      - Who shot JFK? => shot JFK
    - Etc
More General Query Processing

- WordNet Query Expansion
  - Many lexical alternations: ‘How tall’ → ‘The height is’
  - Replace adjectives with corresponding ‘attribute noun’

- Verb conversion:
  - Morphological processing
    - DO-AUX .... V-INF ➔ V+inflection
    - Generation via PC-KIMMO

- Phrasing:
  - Some noun phrases should treated as units, e.g.:
    - Proper nouns: “White House”; phrases: “question answering”

- Query formulation contributes significantly to effectiveness
Query Expansion
Query Expansion

• Basic idea:
  • Improve matching by adding words with similar meaning/similar topic to query

• Alternative strategies:
  • Use fixed lexical resource
    • E.g. WordNet
  • Use information from document collection
    • Pseudo-relevance feedback
WordNet Based Expansion

- In Information Retrieval settings, mixed history
  - Helped, hurt, or no effect
  - With long queries & long documents, no/bad effect

- Some recent positive results on short queries
  - E.g. Fang 2008
  - Contrasts different WordNet, Thesaurus similarity
  - Add semantically similar terms to query
    - Additional weight factor based on similarity score
Similarity Measures

- Definition similarity: $S_{\text{def}}(t_1, t_2)$
  - Word overlap between glosses of all synsets
    - Divided by total numbers of words in all synsets glosses

- Relation similarity:
  - Get value if terms are:
    - Synonyms, hypernyms, hyponyms, holonyms, or meronyms

- Term similarity score from Lin’s thesaurus
Results

- Definition similarity yields significant improvements
  - Allows matching across POS
  - More fine-grained weighting than binary relations

- Evaluated on IR task with MAP

<table>
<thead>
<tr>
<th></th>
<th>BL</th>
<th>Def</th>
<th>Syn</th>
<th>Hype</th>
<th>Hypo</th>
<th>Mer</th>
<th>Hol</th>
<th>Lin</th>
<th>Com</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAP</td>
<td>0.19</td>
<td>0.22</td>
<td>0.19</td>
<td>0.19</td>
<td>0.19</td>
<td>0.19</td>
<td>0.19</td>
<td>0.19</td>
<td>0.21</td>
</tr>
<tr>
<td>Imp</td>
<td>16%</td>
<td>4.3%</td>
<td>0</td>
<td>0</td>
<td>0.5%</td>
<td>3%</td>
<td>4%</td>
<td>15%</td>
<td></td>
</tr>
</tbody>
</table>
Managing Morphological Variants

- Bilotti et al. 2004
- “What Works Better for Question Answering: Stemming or Morphological Query Expansion?”

Goal:
- Recall-oriented document retrieval for QA
  - Can’t answer questions without relevant docs

Approach:
- Assess alternate strategies for morphological variation
Question

- Comparison
  - Index time stemming
    - Stem document collection at index time
    - Perform comparable processing of query
    - Common approach
      - Widely available stemmer implementations: Porter, Krovetz
  
- Query time morphological expansion
  - No morphological processing of documents at index time
  - Add additional morphological variants at query time
    - Less common, requires morphological generation
Prior Findings

- Mostly focused on stemming
- Mixed results (in spite of common use)
  - Harman found little effect in ad-hoc retrieval: Why?
    - Morphological variants in long documents
    - Helps some, hurts others: How?
      - Stemming captures unrelated senses: e.g. AIDS → aid
  - Others:
    - Large, obvious benefits on morphologically rich langs.
    - Improvements even on English
Overall Approach

- Head-to-head comparison
- AQUAIN’T documents
  - Enhanced relevance judgments
- Retrieval based on Lucene
  - Boolean retrieval with tf-idf weighting
- Compare retrieval varying stemming and expansion
- Assess results
Example

- **Q:** What is the name of the volcano that destroyed the ancient city of Pompeii?  
  **A:** Vesuvius

- **New search query:** “Pompeii” and “Vesuvius”

- **Relevant:** In A.D. 79, long-dormant Mount Vesuvius erupted, burying the Roman cities of Pompeii and Herculaneum in volcanic ash.”

- **Unsupported:** Pompeii was pagan in A.D. 79, when Vesuvius erupted.

- **Irrelevant:** Vineyards near Pompeii grow in volcanic soil at the foot of Mt. Vesuvius
Stemming & Expansion

- Base query form: Conjunct of disjuncts
  - Disjunction over morphological term expansions
  - Rank terms by IDF
  - Successive relaxation by dropping lowest IDF term

- Contrasting conditions:
  - Baseline: No nothing (except stopword removal)
  - Stemming: Porter stemmer applied to query, index
  - Unweighted inflectional expansion:
    - POS-based variants generated for non-stop query terms
  - Weighted inflectional expansion: prev. + weights
Example

- Q: What lays blue eggs?
- Baseline: blue AND eggs AND lays
- Stemming: blue AND egg AND lai
- UIE: blue AND (eggs OR egg) AND (lays OR laying OR lay OR laid)
- WIE: blue AND (eggs OR egg\textsuperscript{w}) AND (lays OR laying\textsuperscript{w} OR lay\textsuperscript{w} OR laid\textsuperscript{w})
Evaluation Metrics

- Recall-oriented: why?
  - All later processing filters

- Recall @ n:
  - Fraction of relevant docs retrieved at some cutoff

- Total document reciprocal rank (TDRR):
  - Compute reciprocal rank for rel. retrieved documents
  - Sum overall documents
  - Form of weighted recall, based on rank
## Results

<table>
<thead>
<tr>
<th>Limit</th>
<th>Experiment</th>
<th>Recall</th>
<th>TDRR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>relevant</td>
<td>both</td>
</tr>
<tr>
<td>100</td>
<td>unstemmed</td>
<td>0.2720</td>
<td>0.2595</td>
</tr>
<tr>
<td></td>
<td>stemmed</td>
<td>0.2589</td>
<td>0.2460</td>
</tr>
<tr>
<td></td>
<td>expanded</td>
<td>0.2748</td>
<td>0.2612</td>
</tr>
<tr>
<td></td>
<td>w. expanded</td>
<td>0.2944</td>
<td>0.2798</td>
</tr>
<tr>
<td></td>
<td></td>
<td>+4.82%</td>
<td>+7.82%</td>
</tr>
<tr>
<td>250</td>
<td>unstemmed</td>
<td>0.3738</td>
<td>0.3584</td>
</tr>
<tr>
<td></td>
<td>stemmed</td>
<td>0.3626</td>
<td>0.3474</td>
</tr>
<tr>
<td></td>
<td>expanded</td>
<td>0.3682</td>
<td>0.3533</td>
</tr>
<tr>
<td></td>
<td>w. expanded</td>
<td>0.3776</td>
<td>0.3618</td>
</tr>
<tr>
<td></td>
<td></td>
<td>+8.24%</td>
<td>+7.82%</td>
</tr>
<tr>
<td>500</td>
<td>unstemmed</td>
<td>0.5393</td>
<td>0.5123</td>
</tr>
<tr>
<td></td>
<td>stemmed</td>
<td>0.5364</td>
<td>0.5097</td>
</tr>
<tr>
<td></td>
<td>expanded</td>
<td>0.5467</td>
<td>0.5182</td>
</tr>
<tr>
<td></td>
<td>w. expanded</td>
<td>0.5551</td>
<td>0.5258</td>
</tr>
<tr>
<td></td>
<td></td>
<td>+1.02%</td>
<td>+2.64%</td>
</tr>
<tr>
<td>750</td>
<td>unstemmed</td>
<td>0.5981</td>
<td>0.5689</td>
</tr>
<tr>
<td></td>
<td>stemmed</td>
<td>0.5934</td>
<td>0.5638</td>
</tr>
<tr>
<td></td>
<td>expanded</td>
<td>0.6093</td>
<td>0.5799</td>
</tr>
<tr>
<td></td>
<td>w. expanded</td>
<td>0.6112</td>
<td>0.5816</td>
</tr>
<tr>
<td></td>
<td></td>
<td>+1.05%</td>
<td>+2.23%</td>
</tr>
<tr>
<td>1000</td>
<td>unstemmed</td>
<td>0.6196</td>
<td>0.5917</td>
</tr>
<tr>
<td></td>
<td>stemmed</td>
<td>0.6131</td>
<td>0.5824</td>
</tr>
<tr>
<td></td>
<td>expanded</td>
<td>0.6290</td>
<td>0.5993</td>
</tr>
<tr>
<td></td>
<td>w. expanded</td>
<td>0.6290</td>
<td>0.5993</td>
</tr>
<tr>
<td></td>
<td></td>
<td>+1.52%</td>
<td>+1.28%</td>
</tr>
</tbody>
</table>
Overall Findings

- **Recall:**
  - Porter stemming performs WORSE than baseline
    - At all levels
  - Expansion performs BETTER than baseline
    - Tuned weighting improves over uniform
  - Most notable at lower cutoffs

- **TDRR:**
  - Everything’s worse than baseline
  - Irrelevant docs promoted more
Observations

- Why is stemming so bad?
  - Porter stemming linguistically naïve, over-conflates
    - police = policy; organization = organ; European != Europe
  - Expansion better motivated, constrained

- Why does TDRR drop when recall rises?
  - TDRR – and RR in general – very sensitive to swaps at higher ranks
    - Some erroneous docs added higher

- Expansion approach provides flexible weighting
Local Context and SMT for Question Expansion


- Investigates data-driven approaches to query exp.
  - Local context analysis (pseudo-rel. feedback)
  - Contrasts: Collection global measures
    - Terms identified by statistical machine translation
    - Terms identified by automatic paraphrasing

- Now, huge paraphrase corpus: wikianswers
  - /corpora/UWCSE/wikianswers-paraphrases-1.0.
Motivation

- Fundamental challenge in QA (and IR)
  - Bridging the “lexical chasm”
    - Divide between user’s info need, author’s lexical choice
    - Result of linguistic ambiguity

- Many approaches:
  - QA
    - Question reformulation, syntactic rewriting
    - Ontology-based expansion
    - MT-based reranking
  - IR: query expansion with pseudo-relevance feedback
Task & Approach

- **Goal:**
  - Answer retrieval from FAQ pages
    - IR problem: matching queries to docs of Q-A pairs
    - QA problem: finding answers in restricted document set

- **Approach:**
  - Bridge lexical gap with statistical machine translation
  - Perform query expansion
    - Expansion terms identified via phrase-based MT
Creating the FAQ Corpus

- Prior FAQ collections limited in scope, quality
  - Web search and scraping ‘FAQ’ in title/url
  - Search in proprietary collections
  - 1-2.8M Q-A pairs
    - Inspection shows poor quality

- Extracted from 4B page corpus (they’re Google)
  - Precision-oriented extraction
    - Search for ‘faq’, Train FAQ page classifier ➔ ~800K pages
    - Q-A pairs: trained labeler: features?
      - punctuation, HTML tags (<p>,...), markers (Q:), lexical (what,how)
      - ➔ 10M pairs (98% precision)
Machine Translation Model

- SMT query expansion:
  - Builds on alignments from SMT models

- Basic noisy channel machine translation model:
  - \( e: \) English; \( f: \) French
  - \( \arg \max_e p(e \mid f) = \arg \max_e p(f \mid e)p(e) \)
  - \( p(e): \) ‘language model’; \( p(f \mid e): \) translation model
  - Calculated from relative frequencies of phrases
  - Phrases: larger blocks of aligned words
  - Sequence of phrases:
    \[
    p(f_1^l \mid e_1^l) = \prod_{i=1}^{l} p(f_i \mid e_i)
    \]
Question-Answer Translation

- View Q-A pairs from FAQ as translation pairs
  - Q as translation of A (and vice versa)

- Goal:
  - Learn alignments b/t question words & synonymous answer words
    - Not interested in fluency, ignore that part of MT model

- Issues: Differences from typical MT
  - Length differences ➔ Modify null alignment weights
  - Less important words ➔ Use intersection of bidirectional alignments
Example

- Q: “How to live with cat allergies”
- Add expansion terms
  - Translations not seen in original query
SMT-based Paraphrasing

- Key approach intuition:
  - Identify paraphrases by translating to and from a ‘pivot’ language
  - Paraphrase rewrites yield phrasal ‘synonyms’
    - E.g. translate E -> C -> E: find E phrases aligned to C

- Given paraphrase pair (trg, syn): pick best pivot

\[
p(syn | trg) = \max_{src} p(src | trg) p(syn | src) \\
p(trg | syn) = \max_{src} p(src | syn) p(trg | src)
\]
SMT-based Paraphrasing

- Features employed:
  - Phrase translation probabilities, lexical translation probabilities, reordering score, # words, # phrases, LM

- Trained on NIST multiple Chinese-English translations

\[
p(syn_1^I \mid trg_1^I) = \left( \prod_{i=1}^{I} p_{\phi}(syn_i \mid trg_i) \right)^{\lambda_{\phi}}
\]
\[
\times p_{\phi'}(trg_i \mid syn_i)^{\lambda_{\phi'}} \times p_w(syn_i \mid trg_i)^{\lambda_w}
\]
\[
\times p_w(trg_i \mid syn_i)^{\lambda_w} \times p_d(syn_i, trg_i)^{\lambda_d}
\]
\[
\times l_w(syn_1^I)^{\lambda_i} \times c_{\phi}(syn_1^I)^{\lambda_c} \times p_{LM}(syn_1^I)^{\lambda_{LM}}
\]
Example

- Q: “How to live with cat allergies”
- Expansion approach:
  - Add new terms from n-best paraphrases
Retrieval Model

- Weighted linear combination of vector similarity vals
  - Computed between query and fields of Q-A pair

- 8 Q-A pair fields:
  - 1) Full FAQ text; 2) Question text; 3) answer text;
  - 4) title text; 5-8) 1-4 without stopwords

- Highest weights: Raw Q text;
  - Then stopped full text, stopped Q text
  - Then stopped A text, stopped title text

- No phrase matching or stemming
Query Expansion

- SMT Term selection:
  - New terms from 50-best paraphrases
    - 7.8 terms added
  - New terms from 20-best translations
    - 3.1 terms added
    - Why? - paraphrasing more constrained, less noisy

- Weighting: Paraphrase: same; Trans: higher A text

- Local expansion (Xu and Croft)
  - top 20 docs, terms weighted by tfidf of answers
    - Use answer preference weighting for retrieval
    - 9.25 terms added
Experiments

- Test queries from MetaCrawler query logs
  - 60 well-formed NL questions

- Issue: Systems fail on 1/3 of questions
  - No relevant answers retrieved
    - E.g. “how do you make a cornhusk doll?”,”what does 8x certification mean”, etc
  - Serious recall problem in QA DB

- Retrieve 20 results:
  - Compute evaluation measures @10, 20
Evaluation

- Manually label top 20 answers by 2 judges

- Quality rating: 3 point scale
  - adequate (2): Includes the answer
  - material (1): Some relevant information, no exact ans
  - unsatisfactory (0): No relevant info

- Compute ‘Success_{type} @ n’
  - Type: 2,1,0 above
  - n: # of documents returned

- Why not MRR? - Reduce sensitivity to high rank
  - Reward recall improvement
  - MRR rewards systems with answers in top 1, but poorly on everything else
## Results

<table>
<thead>
<tr>
<th></th>
<th>$S_2@10$</th>
<th>$S_2@20$</th>
<th>$S_{1,2}@10$</th>
<th>$S_{1,2}@20$</th>
</tr>
</thead>
<tbody>
<tr>
<td>baseline $\textit{tfidf}$</td>
<td>27</td>
<td>35</td>
<td>58</td>
<td>65</td>
</tr>
<tr>
<td>local expansion</td>
<td>30 (+11.1)</td>
<td>40 (+14.2)</td>
<td>57 (-1)</td>
<td>63 (-3)</td>
</tr>
<tr>
<td>SMT-based expansion</td>
<td>38 (+40.7)</td>
<td>43 (+22.8)</td>
<td>58</td>
<td>65</td>
</tr>
</tbody>
</table>
## Example Expansions

<table>
<thead>
<tr>
<th>how to live with cat allergies</th>
<th>how to design model rockets</th>
</tr>
</thead>
<tbody>
<tr>
<td>allergens allergic infections filter plasmac</td>
<td>models represented orientation drawings analysis element environment different structure</td>
</tr>
<tr>
<td>rhinitis introduction effective replacement</td>
<td>models rocket</td>
</tr>
<tr>
<td>allergy cats pet food</td>
<td>missiles missile rocket grenades arrow designing prototype models ways paradigm</td>
</tr>
<tr>
<td>way allergens life allergy feline ways living allergen</td>
<td>what is dna hybridization</td>
</tr>
<tr>
<td></td>
<td>instructions individual blueprint characteristics chromosomes deoxyribonucleic information</td>
</tr>
<tr>
<td></td>
<td>genetic molecule</td>
</tr>
<tr>
<td></td>
<td>slides clone cdna sitting sequences</td>
</tr>
<tr>
<td></td>
<td>hibridization hybrids hybridation anything hibridacion hybridising adn hybridisation nothing</td>
</tr>
<tr>
<td></td>
<td>how to enhance competitiveness of indian industries</td>
</tr>
<tr>
<td></td>
<td>resources production quality processing established investment development facilities increase industry</td>
</tr>
<tr>
<td></td>
<td>promote raise improve increase industry strengthen</td>
</tr>
<tr>
<td></td>
<td>how to induce labour</td>
</tr>
<tr>
<td></td>
<td>experience induction practice imagination concentration information consciousness different relaxation</td>
</tr>
<tr>
<td></td>
<td>birth industrial induced induces</td>
</tr>
<tr>
<td></td>
<td>way workers inducing employment ways labor working child work job action unions</td>
</tr>
</tbody>
</table>
Observations

- Expansion improves for rigorous criteria
  - Better for SMT than local RF

- Why?
  - Both can introduce some good terms
  - Local RF introduces more irrelevant terms
  - SMT more constrained
  - Challenge: Balance introducing info vs noise
Machine Learning Approaches

- Diverse approaches:
  - Assume annotated query logs, annotated question sets, matched query/snippet pairs
  - Learn question paraphrases (MSRA)
    - Improve QA by setting question sites
    - Improve search by generating alternate question forms