THUIR at NTCIR-12 IMine Task

Zeyang Liu, Ye Chen, Rongjie Cai, Jiaxin Mao, Chao Wang, Cheng Luo, Xin Li, Yiqun Liu, Min Zhang, Huanbo Luan, Shaoping Ma
Information Retrieval Group, Department of Computer Sci. & Tech., Tsinghua University
chenye617@gmail.com

Framework Overview

- 4-step framework in query understanding subtask
  - Candidate mining, candidate clustering, candidate ranking and vertical predicting.

Candidate Mining From Various Resources

- Disambiguation Items from Wikipedia, Baidu Baie and Hudong Baie.
- Query Facets: Query Completions and Query Suggestions
- Query Reformulations extracted from SogouT
- Query Recommendations from Sogou Search Engine

Candidate Clustering with K-Means

- Goal: Find diversified candidates
- Cluster candidates with K-means algorithm
- Query vector representation
  - Word embedding trained based on SogouT dataset
- Long query candidate: average word embedding of the words
- Cluster candidates into n clusters
  - n = 5 or 10
- Candidate evaluation
  - Clusters
    - Inner distance
      \[ S_{\text{inner}}(q) = \sum_{i=1}^{n} \sum_{q_i} d_{\text{dist}}(q_i, q_i') / |1 - 1|, q_i = q_i' \]
    - Outer distance
      \[ S_{\text{outer}}(q) = \min_{j \neq i} \left( \frac{S_{\text{max}}(\text{dist}(q_i, q_j))}{S_{\text{max}}(\text{dist}(q_i, q_i'))} \right) \]
  - Candidate score
    \[ S(q) = \frac{S_{\text{outer}}(q) - S_{\text{inner}}(q)}{\max(S_{\text{outer}}(q), S_{\text{inner}}(q))} \]

Candidate Ranking with Learning to Rank

- Goal: Find high quality subjective candidates
- Rank candidates with Learning To Rank algorithm (RankBoost)
- Features:
  - Text similarity: length difference, Jaccard similarity, edit distance
  - Word embedding: average, medium, top 3 average of cosine similarities
  - Search Result Similarity: number of shared results
- Metric to optimize: NDCG@10
- Training set: Ranked Subtopics from NTCIR-11 Imine

Vertical Predicting

- Training query
  - Seed URLs generated from an Open Directory Project
  - Random walk on a click-through bipartite graph
- Vertical Distribution
  - Vertical information collected from search result pages
  \[ P = \log(1 + R_i) \]
- Model Construction
  - Logistic Regression
  - Word embedding query representation
  - Six prediction models for each type of verticals
- Vertical Diversification
  - Empirical rules
  - Replace the top 3 Web intent with corresponding vertical intent
  - The top vertical result is chosen to be the vertical intent if there are more than two verticals in the top 3 positions

Experimental Results

<table>
<thead>
<tr>
<th>RUNNAME</th>
<th>SYSTEM DESC.</th>
<th>D9- nDCG</th>
<th>V score</th>
<th>QU score</th>
</tr>
</thead>
<tbody>
<tr>
<td>THUIR-QU-1A</td>
<td>Cluster all subtopic candidates into 10 clusters and select the candidate with the highest S(q) from each cluster.</td>
<td>0.5204</td>
<td>0.5579</td>
<td>0.5392</td>
</tr>
<tr>
<td>THUIR-QU-2A</td>
<td>Cluster all subtopic candidates into 5 clusters and select the candidate with the highest S(q) from each cluster.</td>
<td>0.5550</td>
<td>0.5506</td>
<td>0.5528</td>
</tr>
<tr>
<td>THUIR-QU-1B</td>
<td>Rerank the 10 subtopics generated by THUIR-QU-1A with learning to rank algorithm.</td>
<td>0.5368</td>
<td>0.5763</td>
<td>0.5565</td>
</tr>
<tr>
<td>THUIR-QU-2B</td>
<td>Rerank the 10 subtopics generated by THUIR-QU-2A with learning to rank algorithm.</td>
<td>0.5436</td>
<td>0.5686</td>
<td>0.5561</td>
</tr>
<tr>
<td>THUIR-QU-3A</td>
<td>Cluster all subtopic candidates into 5 clusters and select the candidate with the highest ten S(q).</td>
<td>0.4973</td>
<td>0.5942</td>
<td>0.5458</td>
</tr>
</tbody>
</table>

Retrieval Models

- Probabilistic model based on BM25 and our previous proposed word pair model.
- Relevance score for subtopic
  \[ R(q, D) = W_{BM25} + \alpha \cdot W_{wp} \]
  \[ W_{BM25} = \sum_{i=1}^{m} \log \frac{n - n(q_i) + 0.5}{n(q_i) + 0.5} \cdot \frac{f(q_i, D)}{d_i} \cdot \frac{f(q_i, D)}{d_i} \cdot \frac{f(q_i, D)}{d_i} \cdot \frac{f(q_i, D)}{d_i} \cdot \frac{f(q_i, D)}{d_i} \cdot \frac{f(q_i, D)}{d_i} \cdot \frac{f(q_i, D)}{d_i} \cdot \frac{f(q_i, D)}{d_i} \cdot \frac{f(q_i, D)}{d_i} \]
  \[ W_{wp} = \sum_{i=1}^{m} \log \frac{n - n(q_i) + 0.5}{n(q_i) + 0.5} \cdot \frac{f(q_i, D)}{d_i} \cdot \frac{f(q_i, D)}{d_i} \cdot \frac{f(q_i, D)}{d_i} \cdot \frac{f(q_i, D)}{d_i} \cdot \frac{f(q_i, D)}{d_i} \cdot \frac{f(q_i, D)}{d_i} \cdot \frac{f(q_i, D)}{d_i} \cdot \frac{f(q_i, D)}{d_i} \cdot \frac{f(q_i, D)}{d_i} \]
- Relevance score for query
  \[ R(Q, D) = \sum_{i} R(q_i, D) \times S(q_i) \]
- Vertical importance based on subtopic candidate score
  \[ R(v) = \alpha \cdot S_{v}(\text{score(v)}) \]
- Combination of \( R(Q, D) \) and \( I(v) \)

Experimental Results

<table>
<thead>
<tr>
<th>RUNNAME</th>
<th>D9-nDCG (unclear topics)</th>
<th>nDCG (clear topics)</th>
<th>D9-nDCG+nDCG (all topics)</th>
</tr>
</thead>
<tbody>
<tr>
<td>THUIR-QU-1A</td>
<td>0.6677</td>
<td>0.5756</td>
<td>0.6594</td>
</tr>
<tr>
<td>THUIR-QU-2A</td>
<td>0.6664</td>
<td>0.5652</td>
<td>0.6573</td>
</tr>
<tr>
<td>THUIR-QU-1B</td>
<td>0.6594</td>
<td>0.5416</td>
<td>0.6488</td>
</tr>
<tr>
<td>THUIR-QU-2B</td>
<td>0.6632</td>
<td>0.5442</td>
<td>0.6525</td>
</tr>
<tr>
<td>THUIR-QU-3A</td>
<td>0.6429</td>
<td>0.5506</td>
<td>0.6346</td>
</tr>
</tbody>
</table>