**Query Understanding Subtask**

**Step 1: Subtopic Candidate Retrieval**
- Query Suggestion (Bing, Yahoo!, Baidu, Google)
- Disambiguation page: Wikipedia, Baidu Baike

**Step 2: Subtopic Candidate Clustering**
- Use top 300 search results to represent a subtopic
- Generate a tf-idf vector for each subtopic candidate
- Clustering: K-medoids, Quality Threshold (QT)
- Average linkage
- A cluster -> a subtopic (the centroid)

**Step 3: Subtopic Ranking and Diversification**
- Rank the subtopics using MMR
  \[ d_{i+1} = \arg \max\{\lambda Rel(d) + (1 - \lambda) Div(d, D_i)\} \]
- Relevance: \( \beta \cdot \text{NumOfCandidates} + (1 - \beta) \cdot \text{IAL} \)
- Novelty: cosine similarity of tf-idf vectors

**Step 4: Vertical Intent Classification**
- (1) Rule-based
  - Keyword-based rules (true if x% results contain specific keywords), such as:
    - what/how -> Encyclopedia
    - New, latest, daily -> News
    - Sale, deal, coupon -> Shopping
- (2) Trained Classifier
  - Use Imine-1 topics and their query suggestions as training data
  - Use Bing to generate labels: for a query, check whether Bing’s SERP include answers/results from a specific vertical
  - Use word occurrences as features
  - SVM

**Run Name** | **System Description** | **QU-score** |
--- | --- | ---
rucir-Q-C/E-1Q | Suggestions + Wikipedia, k-medoids, trained classifier | 0.5767
          | 0.5613 |
rucir-Q-C/E-2Q | Suggestions + Wikipedia, k-medoids, rule-based classifier | 0.5495
          | 0.5904 |
rucir-Q-C/E-3Q | Suggestions + Wikipedia, QT clustering, trained classifier | 0.4489
          | 0.4166 |
rucir-Q-C/E-4Q | Suggestions + k-medoids, trained classifier | 0.5311
          | 0.5583 |
rucir-Q-C/E-5Q | Suggestions + Ranking + Diversification | 0.6849
          | 0.6911 |

**Results**
- (1) Trained classifiers >Rules
- (2) Clustering: K-medoids > QT
- (3) Query suggestions + Wikipedia > Query suggestions

**Vertical Incorporating Subtask**

**Subtopic expansion**
- **Subtopic**
  - PlayStation 4
- **Bing’s Related Suggestions**
  - PlayStation 4 new video game
  - PlayStation 4 best price
  - PlayStation 4 controller
- **Expanded Subtopic**
  - PlayStation 4 video game best price controller

**Key Choices**
- Subtopic is short, hence we expand it to a longer query containing more keywords
- Use BM25 as the basic ranking function
- Assume that the virtual result is highly relevant to the corresponding subtopic, and it is treated as a normal document.

**Diversification model – PM2**
\[ q^*_i = \frac{q_i}{2s_i + 1} \]
\[ d^* = \arg \max_{d \in D} \left\{ q^* \cdot \text{rel}(d, t^*) + (1 - \lambda) \sum_{t \in s^*} q_i \cdot \text{rel}(d, t_i) \right\} \]
\[ s^*_i = s_i + \frac{\text{rel}(d^*, t^*_i)}{\sum_{t \in s^*} \text{rel}(d^*, t^*_i)} \text{rel}(d^*, t^*_i) \]

In an iterative procedure, first we compute the quotient q for each subtopic t. Then we check the unselected documents to select the next best document d*. And finally we update the occupied seat s.

**Experimental Results**

**Run Name** | **Chinese Relevant Queries** | **English Relevant Queries** |
--- | --- | ---
rucir-V-C/E-1M | SExp+QU | D-DCG@10 | D-DCG@10 | D-Recall | D-Recall |
          | 0.7935*17 | 0.5442*15 | 0.9449*17 | 0.7310 | 0.5206*15 | 0.8986 |
rucir-V-C/E-2M | SExp+Sign | 0.7079 | 0.5684 | 0.4891 | 0.8267 |
rucir-V-C/E-3M | isSExp+QU | 0.6061 | 0.3999 | 0.8892 |
rucir-V-C/E-4M | isSExp+Sign | 0.6093 | 0.4144 | 0.5742 |
rucir-V-C/E-5M | [Baseline] | 0.6093 | 0.4141 | 0.5742 |

**Results**
- (1) Using subtopics mined by **rucir-Q-C/E-1Q** is better than directly using suggestions
- (2) Subtopic expansion works well