A Computational Framework for Question Processing in Community Question Answering Services

Baichuan Li
January 9, 2014

Thesis Committee:
Prof. FU Wai Chee Ada (Chair)
Prof. LEE Ho Man Jimmy (Internal Examiner)
Prof. ZHU Xiaoyan (External Examiner)
Prof. KING Kuo Chin Irwin (Supervisor)
Prof. LYU Rung Tsong Michael (Supervisor)
Agenda

• Introduction
• Background
• Question Quality Analysis and Prediction
• Question Routing
  – Quality and Availability
  – Category
• Question Structuralization
• Conclusion and Future Work
Agenda

• Introduction
• Background
• Question Quality Analysis and Prediction
• Question Routing
  – Quality and Availability
  – Category
• Question Structuralization
• Conclusion and Future Work
Community Question Answering

- What is CQA?
- Why CQA?
Example: Yahoo! Answers

- The most popular CQA portal among the world
- Two questions are asked and six are answered every second
- 300 million questions have been asked by July, 2012
Challenges in CQA

• Inefficient Question Answering
  – Sharp increase of questions
  – Time lag between Q&A

• Straightforward Content Organization
Objective of Thesis

• User
  – Facilitate answerers access to proper questions
  – Help askers obtain information more effectively

• System
  – Improve content organization
  – Enhance QA efficiency

Solution

A computational framework for question processing
Structure of Thesis

- Quality Analysis and Prediction
- Routing
- Structuralization

- Quality and Availability
- Category

- Chapter 3
- Chapter 4
- Chapter 5
- Chapter 6
Agenda

• Introduction
• Background
• Question Quality Analysis and Prediction
• Question Routing
  – Quality and Availability
  – Category
• Question Structuralization
• Conclusion and Future Work
Question Processing

• Question Retrieval
  – Basic models (Jeon et al., 2005; Duan et al., 2008)
  – Extra information: category (Cao et al., 2010), syntactic knowledge (Wang et al., 2009), answer (Bian et al., 2008), etc.

• Question Classification
  – Properties: urgency, subjectivity
  – Models: SVM (Li et al., 2008), Co-training (Li et al., 2008), sequential minimal optimization (Harper et al., 2009)

• Question Routing
  – User Profiling
  – Question Profiling
  – Matching
Answer Processing

• Answer Quality Evaluation
  – Classification-based (Jeon et al., 2006; Eugene et al., 2008; Shah et al., 2010)
  – Ranking-based (Suryanto et al., 2009; Wang et al., 2009)

• Answer Summarization
  – Question type-based (Liu et al., 2008)
  – Constraint-based (Tomasoni et al., 2010; Liu et al., 2011)
  – Graph-based (Chan et al., 2012; Pande et al., 2013)
User Processing

- **Expert Finding**
  - Link analysis (Jurczyk et al., 2007; Zhang et al., 2007)
  - Content analysis (Liu et al., 2005; Budalakoti, 2013)

- **User Analysis**
  - User behavior (Gazan, 2006; Rodrigues et al., 2008)
  - Community (Li et al., 2012)

- **User Satisfaction Prediction**
  - Classification (Liu et al., 2008; Liu et al., 2010)
Agenda

- Introduction
- Background
- **Question Quality Analysis and Prediction** *(Chapter 3)*
- Question Routing
  - Quality and Availability *(Chapter 4)*
  - Category *(Chapter 5)*
- **Question Structuralization** *(Chapter 6)*
- Conclusion and Future Work
Question Quality Analysis and Prediction

- Motivation and Definition
- Study One: Factors Affecting Question Quality
- Study Two: Question Quality Prediction
- Summary
Question Quality

Number of tag-of-interests vs Number of answers

- In ten years, how do you think we’ll be using the Internet? 3 answers - 4 years ago
- What is one thing you cannot do online that you wish you could? 3 answers - 4 years ago
- What is a blog? 2 answers - 7 years ago
- Who invented the mouse? 2 answers - 7 years ago
Definition of Question Quality

Construct of question quality in CQA
Motivation

• Question quality affects answer quality
  – Low quality questions hinder QA efficiency
  – High quality questions promote the development of the community
• Question routing
• Identifying question quality facilitates question search and recommendation
Study One: Factors Affecting Question Quality

• Factors

• Process
  – Select the two most popular subcategories (say, Music and Movies) and check their distributions of question quality
  – Track askers with at least five questions in both these two subcategories (22 in total)
Data Description

Summary of data (crawled from Jul 7, 2010 to Sep 6, 2010)

<table>
<thead>
<tr>
<th>Subcategory</th>
<th># of questions</th>
<th># of askers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Celebrities</td>
<td>11,817</td>
<td>7,087</td>
</tr>
<tr>
<td>Comics &amp; Animation</td>
<td>11,327</td>
<td>6,801</td>
</tr>
<tr>
<td>Horoscopes</td>
<td>7,235</td>
<td>2,203</td>
</tr>
<tr>
<td>Jokes &amp; Riddles</td>
<td>3,685</td>
<td>2,569</td>
</tr>
<tr>
<td>Magazines</td>
<td>548</td>
<td>462</td>
</tr>
<tr>
<td>Movies</td>
<td>15,121</td>
<td>10,996</td>
</tr>
<tr>
<td>Music</td>
<td>32,948</td>
<td>18,589</td>
</tr>
<tr>
<td>Other - Entertainment</td>
<td>2,244</td>
<td>2,003</td>
</tr>
<tr>
<td>Polls &amp; Surveys</td>
<td>138,507</td>
<td>18,685</td>
</tr>
<tr>
<td>Radio</td>
<td>640</td>
<td>272</td>
</tr>
<tr>
<td>Television</td>
<td>14,477</td>
<td>10,146</td>
</tr>
<tr>
<td>All</td>
<td>238,549</td>
<td>62,853</td>
</tr>
</tbody>
</table>

Questions are assigned to four classes according to manually crafted rules.
Observations

- The distributions of question quality in these subcategories are **similar**
- Topics only **cannot** distinguish good questions from bad ones
### Observations

#### Summary of question quality for different askers

<table>
<thead>
<tr>
<th>User</th>
<th>Music</th>
<th>Movies</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>Std</td>
</tr>
<tr>
<td>1</td>
<td>2.50</td>
<td>0.93</td>
</tr>
<tr>
<td>2</td>
<td>2.45</td>
<td>0.52</td>
</tr>
<tr>
<td>3</td>
<td>1.86</td>
<td>0.90</td>
</tr>
<tr>
<td>4</td>
<td>2.65</td>
<td>0.72</td>
</tr>
<tr>
<td>5</td>
<td>1.90</td>
<td>0.74</td>
</tr>
<tr>
<td>6</td>
<td>2.62</td>
<td>0.87</td>
</tr>
<tr>
<td>7</td>
<td>2.48</td>
<td>0.68</td>
</tr>
<tr>
<td>8</td>
<td>2.86</td>
<td>0.92</td>
</tr>
<tr>
<td>9</td>
<td>2.38</td>
<td>0.92</td>
</tr>
<tr>
<td>10</td>
<td>2.50</td>
<td>0.53</td>
</tr>
<tr>
<td>11</td>
<td>2.00</td>
<td>0.71</td>
</tr>
<tr>
<td>12</td>
<td>2.48</td>
<td>0.95</td>
</tr>
<tr>
<td>13</td>
<td>2.84</td>
<td>0.68</td>
</tr>
<tr>
<td>14</td>
<td>1.33</td>
<td>0.52</td>
</tr>
<tr>
<td>15</td>
<td>1.90</td>
<td>0.74</td>
</tr>
<tr>
<td>16</td>
<td>1.80</td>
<td>0.84</td>
</tr>
<tr>
<td>17</td>
<td>2.15</td>
<td>0.55</td>
</tr>
<tr>
<td>18</td>
<td>2.36</td>
<td>0.92</td>
</tr>
<tr>
<td>19</td>
<td>2.00</td>
<td>1.00</td>
</tr>
<tr>
<td>20</td>
<td>2.00</td>
<td>0.67</td>
</tr>
<tr>
<td>21</td>
<td>2.69</td>
<td>0.68</td>
</tr>
<tr>
<td>22</td>
<td>2.13</td>
<td>0.99</td>
</tr>
</tbody>
</table>

- **For the same topic**
  - Different askers obtain various question quality
    - User 8 VS User 16 in *Music*
    - User 2 VS User 3 in *Movies*

- **For the same asker**
  - Question quality varies on different topics
    - User 14
Observations

Question Quality

A Computational Framework for Question Processing in CQA Services
Challenges

• A new question comes…
• No answers, no tags
• Can we predict a new question’s quality?
Study Two: Question Quality Prediction

- Modeling the relationships among questions, topics and askers as a bipartite graph
Mutual Reinforcement Label Propagation for Predicting Question Quality

Algorithm 1 MRLP

Input: user asking expertise vector $U_0^k$, question quality vector $Q_0^k$, $E$, transition matrixes $M$ and $N$, weighting coefficients $\alpha$ and $\beta$, some manual labels of $U_0^k$ and/or $Q_0^k$.

1: Set $c = 0$.
2: while not convergence do
3:   Propagate user expertise. $U_{c+1}^k = \alpha \cdot M \cdot U_c^k + \left(1 - \frac{\alpha}{\alpha}\right) \cdot E' \cdot Q_c^k$.
4:   Propagate question quality. $Q_{c+1}^k = \beta \cdot N \cdot Q_c^k + \left(1 - \frac{\beta}{\beta}\right) \cdot E^T \cdot U_{c+1}^k$, where $E^T$ is the transpose of $E$.
5:   Clamp the labeled data of $U_{c+1}^k$ and $Q_{c+1}^k$.
6:   Set $c = c + 1$.
7: end while
Example

A Computational Framework for Question Processing in CQA Services
Question Quality Estimation

A Computational Framework for Question Processing in CQA Services
Asking Expertise Estimation

A Computational Framework for Question Processing in CQA Services
## Features

### Summary of features

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>IG</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Question-related features</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sub_len</td>
<td>Number of words in question subject (title)</td>
<td>0.0115</td>
</tr>
<tr>
<td>Con_len</td>
<td>Number of words in question content</td>
<td>0.0029</td>
</tr>
<tr>
<td>Wh-type</td>
<td>Whether the question subject starts with Wh-word (e.g., “what”, “where”, etc.)</td>
<td>0.0001</td>
</tr>
<tr>
<td>Sub_punc_den</td>
<td>Number of question subject’s punctuation over length</td>
<td>0.0072</td>
</tr>
<tr>
<td>Sub_typo_den</td>
<td>Number of question subject’s typos over length</td>
<td>0.0021</td>
</tr>
<tr>
<td>Sub_space_den</td>
<td>Number of question subject’s spaces over length</td>
<td>0.0138</td>
</tr>
<tr>
<td>Con_punc_den</td>
<td>Number of question content’s punctuation over length</td>
<td>0.0096</td>
</tr>
<tr>
<td>Con_typo_den</td>
<td>Number of question content’s typos over length</td>
<td>0.0006</td>
</tr>
<tr>
<td>Con_space_den</td>
<td>Number of question content’s spaces over length</td>
<td>0.0113</td>
</tr>
<tr>
<td>Avg_word</td>
<td>Number of words per sentence in question’s subject and content</td>
<td>0.0048</td>
</tr>
<tr>
<td>Cap_error</td>
<td>The fraction of sentences which are started with a small letter</td>
<td>0.0064</td>
</tr>
<tr>
<td>POS_entropy</td>
<td>The entropy of the part-of-speech tags of the question</td>
<td>0.0004</td>
</tr>
<tr>
<td>NF_ratio</td>
<td>The fraction of words that are not the top 10 frequent words in the collection</td>
<td>0.0009</td>
</tr>
<tr>
<td><strong>Askers-related features</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total_points</td>
<td>Total points the asker earns</td>
<td>0.0339</td>
</tr>
<tr>
<td>Total_answers</td>
<td>Number of answers the asker provided</td>
<td>0.0436</td>
</tr>
<tr>
<td>Best_answers</td>
<td>Number of best answers the asker provided</td>
<td>0.0331</td>
</tr>
<tr>
<td>Total_questions</td>
<td>Number of questions the asker provided</td>
<td>0.0339</td>
</tr>
<tr>
<td>Resolved_questions</td>
<td>Number of resolved questions asked by the asker</td>
<td>0.0357</td>
</tr>
<tr>
<td>Star_received</td>
<td>Number of stars received for all questions</td>
<td>0.0367</td>
</tr>
</tbody>
</table>
Methods for Comparison

- Logistic Regression
  - LG_Q and LG_QA

- Stochastic Gradient Boosted Tree (Friedman, J. H., 1999)
  - SGBT_Q and SGBT_QA

- Harmonic Function (Zhou et al., 2007)
  - HF_Q and HF_QA
Results: Accuracy

Different algorithms’ accuracy (*Music*)

- HF_Q
- HF_QA
- SGBT_Q
- SGBT_QA
- MRLP

Accuracy vs. Training rate (%)
Sensitivity & Specificity

- Sensitivity measures the algorithm’s ability to identify (recall) high-quality questions
  \[ \text{Sensitivity} = \frac{TP}{TP+FN} \]

- Specificity measures the algorithm’s ability to identify (recall) low-quality questions
  \[ \text{Specificity} = \frac{TN}{TN+FP} \]
Results: Sensitivity & Specificity

Different algorithms’ Sensitivity and Specificity (Music)
Contribution of Chapter 3

• First to investigate question quality in CQA
• Define question quality in CQA
• Conduct two studies
  – Analyze the factors influencing question quality
  – Propose a mutual reinforcement-based label propagation algorithm to predict question quality
Agenda

• Introduction
• Background
• Question Quality Analysis and Prediction (Chapter 3)
• Question Routing  
  – Quality and Availability (Chapter 4)  
  – Category (Chapter 5)
• Question Structuralization (Chapter 6)
• Conclusion and Future Work
Motivation

User participation in Yahoo! Answers (Guo et al., 2008)

User Participation

Low participation rate

<table>
<thead>
<tr>
<th>Answers</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>AnsQ=0</td>
<td>19.68%</td>
</tr>
<tr>
<td>AnsQ=1</td>
<td>31.77%</td>
</tr>
<tr>
<td>AnsQ=2</td>
<td>12.88%</td>
</tr>
<tr>
<td>AnsQ=3</td>
<td>7.53%</td>
</tr>
<tr>
<td>AnsQ=4</td>
<td>3.52%</td>
</tr>
<tr>
<td>AnsQ=5</td>
<td>4.95%</td>
</tr>
<tr>
<td>AnsQ&gt;=6</td>
<td>19.67%</td>
</tr>
</tbody>
</table>
Motivation

Status of tracked questions in Yahoo! Answers and Baidu Zhidao within 48 hours

Long wait time
Question Routing

• Definition
• Framework
  – Expertise Estimation
  – Availability Estimation
• Experiments
• Summary
Question Routing (QR)

• What is QR?
  – The process of routing a new posted question to the users who are most likely to give good answers in a short period

• Two requirements
  – Expertise
  – Availability
The framework of Question Routing

\[ QR(u_i, q_r, T) = \gamma \cdot E(u_i, q_r) + (1 - \gamma) \cdot A(u_i, T). \]

- \( u_i \)'s expertise on \( q_r \)
- \( u_i \)'s availability during \( T \)

A Computational Framework for Question Processing in CQA Services
Expertise Estimation

• Without answer quality
  – Query-likelihood language model

\[
E(u_i, q_r) = P(q_r|q_{u_i}) = \prod_{\omega \in q_r} P(\omega|q_{u_i})
\]

\[
P(\omega|q_{u_i}) = (1 - \lambda)P_{ml}(\omega|q_{u_i}) + \lambda P_{ml}(\omega|C)
\]

\[
P(\omega|q_{u_i}) = \frac{tf(\omega, q_{u_i})}{\sum_{\omega' \in q_{u_i}} tf(\omega', q_{u_i})}
\]

\[
P(\omega|C) = \frac{tf(\omega, C)}{\sum_{\omega' \in C} tf(\omega', C)}
\]
Expertise Estimation

• With answer quality

\[ E(u_i, q_r) = \alpha \cdot P(q_r | q_{u_i}) + (1 - \alpha) \cdot Q(u_i, q_r) \]

• Quality score
  – Basic model
    • Weighted average answer quality of similar questions
  – Smoothed model
    • Leverage other similar users’ answer quality of similar questions
  – Quality estimation
    • Logistic regression

<table>
<thead>
<tr>
<th></th>
<th>q₁</th>
<th>q₂</th>
<th>q₃</th>
<th>q₄</th>
<th>q_{new}</th>
</tr>
</thead>
<tbody>
<tr>
<td>u₁</td>
<td>0.7</td>
<td></td>
<td></td>
<td></td>
<td>?</td>
</tr>
<tr>
<td>u₂</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>u₃</td>
<td>0.9</td>
<td></td>
<td></td>
<td>0.8</td>
<td></td>
</tr>
<tr>
<td>u₄</td>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Availability Estimation

- Model it as a trend analysis problem
- Employ an auto-regressive model

\[ A(u_i, t) = \lambda_1 A(u_i, t - 1) + \ldots + \lambda_p A(u_i, t - p) + \varepsilon \]

- The answerer \( u_i \)'s availability for a period of time \( T \)

\[ A(u_i, T) = 1 - \prod_{j=1}^{s} (1 - A(u_i, t_j)) \]
## Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>QR score</th>
</tr>
</thead>
<tbody>
<tr>
<td>QLL</td>
<td>$QR(u_i, q_r, T) = P(q_r</td>
</tr>
<tr>
<td>Basic Q</td>
<td>$QR(u_i, q_r, T) = \alpha \cdot P(q_r</td>
</tr>
<tr>
<td>Smoothed Q</td>
<td>$QR(u_i, q_r, T) = \alpha \cdot P(q_r</td>
</tr>
<tr>
<td>QLL + AE</td>
<td>$QR(u_i, q_r, T) = \gamma \cdot P(q_r</td>
</tr>
<tr>
<td>Basic Q + AE</td>
<td>$QR(u_i, q_r, T) = \gamma \cdot [\alpha \cdot P(q_r</td>
</tr>
<tr>
<td>Smoothed Q + AE</td>
<td>$QR(u_i, q_r, T) = \gamma \cdot [\alpha \cdot P(q_r</td>
</tr>
</tbody>
</table>
Results

Different methods’ MRR for QR

<table>
<thead>
<tr>
<th>QLL</th>
<th>Basic Q</th>
<th>Smoothed Q</th>
<th>QLL + AE</th>
<th>Basic Q + AE</th>
<th>Smoothed Q + AE</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0389</td>
<td>0.0494</td>
<td>0.052</td>
<td>0.0405</td>
<td>0.0511</td>
<td>0.0541</td>
</tr>
</tbody>
</table>

MRR value of Basic Q and Smoothed Q versus various $\alpha$

MRR versus $\gamma$ across different methods
Contribution of Chapter 4

• Propose a *Question Routing* framework
  – User expertise
  – Answering availability

• Design user expertise estimation and availability estimation models

• Demonstrate the effectiveness of proposed framework
A Computational Framework for Question Processing in CQA Services

Agenda

• Introduction
• Background
• Question Quality Analysis and Prediction (Chapter 3)
  • Question Routing
    – Quality and Availability (Chapter 4)
    – Category (Chapter 5)
• Question Structuralization (Chapter 6)
• Conclusion and Future Work
Motivation

• Previous Methods for Expertise Estimation
  – PLSA (Qu et al. 2009)
  – LDA + LM (Liu et al. 2010)

• Limitations
  – Irrelevant answerers
    • All answerers’ expertise is estimated
  – Irrelevant profiles
    • All previous answered questions are employed as user profile
Two improvements in efficiency of QR
- Higher accuracy
- Lower cost
Category-Sensitive Question Routing

- Category for QR
  - Category-Answerer Indexes
  - Category-Sensitive Language Models
- Experiments
- Summary
Question Category for QR

A Computational Framework for Question Processing in CQA Services
Category-Answerer Indexes

• Severe index
  – Leaf category-based

• Lenient index
  – Top category-based
Category-Sensitive LMs

• Basic category-sensitive QLLM (BCS-LM)
  – Only consider profiles in the new question’s leaf category

• Transferred category-sensitive QLLM (TCS-LM)
  – Incorporate profiles in similar leaf categories
\[ E(u_i, q_r, c_j) \equiv P_{bcs}(u_i | q_r, c_j), \]

\[ P_{bcs}(u_i | q_r, c_j) \propto P_{bcs}(q_r, c_j | u_i) P(u_i), \]

\[ P_{bcs}(q_r, c_j | u_i) = P_{bcs}(q_r | c_j, u_i) P(c_j | u_i), \]

\[ P_{bcs}(q_r | c_j, u_i) = P_{bcs}(q_r | c_j, q_{u_i}) = \prod_{\omega \in q_r} P(\omega | q_{u_{ij}}), \]

\[ P(\omega | q_{u_{ij}}) = (1 - \lambda) P_{ml}(\omega | q_{u_{ij}}) + \lambda P_{ml}(\omega | Coll), \]

where \( c_j \) is \( q_r \)'s category, \( P(c_j | u_i) \) denotes the probability of answering questions in \( c_j \) for \( u_i \), and \( q_{u_{ij}} \) represents the question texts of all previously answered questions in \( c_j \) for \( u_i \).
TCS-LM

A Computational Framework for Question Processing in CQA Services
TCS-LM

\[ P_{tes}(q_r, c_j | u_i) = \frac{\beta P_{bcs}(q_r, c_j | u_i) + \sum_{c_k \in Tran(c_j)} T(c_k \rightarrow c_j) P_{bcs}(q_r, c_k | u_i)}{\beta + \sum_{c_k \in Tran(c_j)} T(c_k \rightarrow c_j)} , \]

\[ c_k \in Tran(c_j) \text{ if } T(c_k \rightarrow c_j) \geq \delta \]

\[
E = \begin{bmatrix}
2 & 0 & 3 & 4 & 0 \\
0 & 3 & 0 & 0 & 0 \\
4 & 8 & 4 & 5 & 1 \\
7 & 0 & 6 & 0 & 2 \\
0 & 4 & 5 & 6 & 1 \\
\end{bmatrix}
\]

\[ T_{ans}(c_j \rightarrow c_k) = T_{ans}(c_k \rightarrow c_j) = \frac{\mathbf{e}_j \cdot \mathbf{e}_k}{|\mathbf{e}_j||\mathbf{e}_k|} . \]
Methods for Comparison

• Cluster-Based Language Model (CBLM)

\[ P(q_r|u_i) = \sum_{\text{Cluster } \omega \in q_r} \prod_{\omega \in q_r} P(\omega|\theta_{\text{Cluster}})^{n(\omega, q_r)} \text{con(Cluster, } u) \]

and

\[ P(\omega|\theta_{\text{Cluster}}) = (1 - \lambda)P(\omega|\text{Cluster}) + \lambda P(\omega|\text{Coll}) \]

\[ \text{con(Cluster, } u_i) = \sum_{\text{qa}} \text{con(qa, } u_i) \]

\[ \text{con(qa, } u_i) = \frac{\prod_{\omega \in q} P(\omega|\theta_{a_{u_i}})}{\sum_{\text{qa}}, \prod_{\omega \in q} P(\omega|\theta_{a_{u_i}})} \]

\[ P(\omega|\theta_{a_{u_i}}) = (1 - \lambda)P(\omega|a_{u_i}) + \lambda P(\omega|\text{Coll}) \]

• Mixture of LDA and QLLM (LDALM)

\[ P(q_r|u_i) = \prod_{\omega \in q_r} P(\omega|\theta_{u_i})^{n(\omega, q_r)} \]

\[ P(\omega|\theta_{u_i}) = \delta P_{LDA}(\omega|\theta_{u_i}) + (1 - \delta) P_{LM}(\omega|\theta_{u_i}) \]

\[ P_{LDA}(\omega|\hat{\theta}, \phi, \theta_{u_i}) = \sum_{z=1}^{Z} P(\omega|z, \phi) P(z|\hat{\theta}, \theta_{u_i}) \]
Experimental Setting

• Data
  – Crawled from Yahoo! Answers
  – 433,072 questions and 270,043 answerers

• Ground Truth
  – GT-A: Answerers who answered the routed question
  – GT-BA: The answerer who gave the best answer of the routed question

• Evaluation Metrics
  – Precision at K (Prec@K)
  – Mean Average Precision (MAP)
  – Mean Reciprocal Rank (MRR)
**Experimental Results**

**Table 1** Different methods’ *Prec@K* in QR versus various *K*s using GT-A (best results are shown in bold)

<table>
<thead>
<tr>
<th>K</th>
<th>QLLM</th>
<th>BCS-LM</th>
<th>TCS-LM</th>
<th>LDALM</th>
<th>CBLM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0795</td>
<td>0.1114 (↑40.13%)</td>
<td>0.1227 (↑54.34%)</td>
<td>0.0989 (↑24.40%)</td>
<td>0.0000</td>
</tr>
<tr>
<td>3</td>
<td>0.1659</td>
<td><strong>0.2364 (↑42.50%)</strong></td>
<td>0.2340 (↑41.05%)</td>
<td>0.1950 (↑17.54%)</td>
<td>0.0000</td>
</tr>
<tr>
<td>5</td>
<td>0.2091</td>
<td><strong>0.2727 (↑30.42%)</strong></td>
<td>0.2705 (↑29.36%)</td>
<td>0.2455 (↑17.41%)</td>
<td>0.0000</td>
</tr>
<tr>
<td>10</td>
<td>0.2705</td>
<td>0.3386 (↑25.18%)</td>
<td><strong>0.3455 (↑27.73%)</strong></td>
<td>0.3102 (↑14.68%)</td>
<td>0.0000</td>
</tr>
<tr>
<td>20</td>
<td>0.3386</td>
<td>0.3909 (↑15.45%)</td>
<td><strong>0.3932 (↑16.13%)</strong></td>
<td>0.3710 (↑9.57%)</td>
<td>0.0091</td>
</tr>
<tr>
<td>40</td>
<td>0.4136</td>
<td>0.4523 (↑9.36%)</td>
<td><strong>0.4591 (↑11.00%)</strong></td>
<td>0.4392 (↑6.19%)</td>
<td>0.0273</td>
</tr>
<tr>
<td>60</td>
<td>0.4477</td>
<td><strong>0.4818 (↑7.62%)</strong></td>
<td>0.4795 (↑7.10%)</td>
<td>0.4649 (↑3.84%)</td>
<td>0.0545</td>
</tr>
<tr>
<td>80</td>
<td>0.4727</td>
<td><strong>0.4955 (↑4.82%)</strong></td>
<td>0.4909 (↑3.85%)</td>
<td>0.4867 (↑2.96%)</td>
<td>0.0727</td>
</tr>
<tr>
<td>100</td>
<td>0.4909</td>
<td><strong>0.5159 (↑5.09%)</strong></td>
<td>0.5114 (↑4.18%)</td>
<td>0.4979 (↑1.43%)</td>
<td>0.0795</td>
</tr>
</tbody>
</table>

**Table 2** Different methods’ *Prec@K* in QR versus various *K*s using GT-BA (best results are shown in bold)

<table>
<thead>
<tr>
<th>K</th>
<th>QLLM</th>
<th>BCS-LM</th>
<th>TCS-LM</th>
<th>LDALM</th>
<th>CBLM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0568</td>
<td>0.0682 (↑20.07%)</td>
<td><strong>0.0773 (↑36.09%)</strong></td>
<td>0.0668 (↑17.61%)</td>
<td>0.0000</td>
</tr>
<tr>
<td>3</td>
<td>0.1091</td>
<td><strong>0.1477 (↑35.38%)</strong></td>
<td>0.1409 (↑29.15%)</td>
<td>0.1258 (↑15.31%)</td>
<td>0.0000</td>
</tr>
<tr>
<td>5</td>
<td>0.1363</td>
<td><strong>0.1705 (↑25.09%)</strong></td>
<td>0.1659 (↑21.72%)</td>
<td>0.1655 (↑21.42%)</td>
<td>0.0000</td>
</tr>
<tr>
<td>10</td>
<td>0.1705</td>
<td>0.2068 (↑21.29%)</td>
<td><strong>0.2091 (↑22.58%)</strong></td>
<td>0.1950 (↑14.40%)</td>
<td>0.0000</td>
</tr>
<tr>
<td>20</td>
<td>0.2205</td>
<td><strong>0.2591 (↑17.51%)</strong></td>
<td>0.2523 (↑14.42%)</td>
<td>0.2472 (↑12.11%)</td>
<td>0.0023</td>
</tr>
<tr>
<td>40</td>
<td>0.2750</td>
<td>0.3114 (↑13.24%)</td>
<td><strong>0.3136 (↑14.04%)</strong></td>
<td>0.2891 (↑5.13%)</td>
<td>0.0091</td>
</tr>
<tr>
<td>60</td>
<td>0.3023</td>
<td><strong>0.3386 (↑12.01%)</strong></td>
<td>0.3386 (↑12.01%)</td>
<td>0.3109 (↑2.84%)</td>
<td>0.0295</td>
</tr>
<tr>
<td>80</td>
<td>0.3182</td>
<td>0.3432 (↑7.86%)</td>
<td><strong>0.3455 (↑8.58%)</strong></td>
<td>0.3225 (↑1.35%)</td>
<td>0.0386</td>
</tr>
<tr>
<td>100</td>
<td>0.3364</td>
<td><strong>0.3614 (↑7.43%)</strong></td>
<td>0.3591 (↑6.75%)</td>
<td>0.3365</td>
<td>0.0386</td>
</tr>
</tbody>
</table>
Experimental Results

Table 3  MRR and MAP of various models under GT-A (best results are shown in bold)

<table>
<thead>
<tr>
<th>Method</th>
<th>MRR</th>
<th>MAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>QLLM</td>
<td>0.1460</td>
<td>0.1070</td>
</tr>
<tr>
<td>BCS-QLLM</td>
<td>0.1893 (↑29.66%)</td>
<td>0.1424 (↑33.08%)</td>
</tr>
<tr>
<td>TCS-QLLM</td>
<td><strong>0.1965 (↑34.59%)</strong></td>
<td><strong>0.1469 (↑37.29%)</strong></td>
</tr>
<tr>
<td>LDALM</td>
<td>0.1695 (↑16.10%)</td>
<td>0.1281 (↑19.72%)</td>
</tr>
<tr>
<td>CBLM</td>
<td>0.0031</td>
<td>0.0024</td>
</tr>
</tbody>
</table>

Table 4  Different methods’ MQRT in QR (in seconds)

<table>
<thead>
<tr>
<th>Method</th>
<th>MQRT (in seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>QLLM</td>
<td>10.4271</td>
</tr>
<tr>
<td>BCS-QLLM</td>
<td>5.5098</td>
</tr>
<tr>
<td>TCS-QLLM</td>
<td>8.9884</td>
</tr>
<tr>
<td>LDALM</td>
<td>16.7689</td>
</tr>
<tr>
<td>CBLM</td>
<td>4.2488</td>
</tr>
</tbody>
</table>

Table 5  Effects of using category-answerer indexes on answerer filtering

<table>
<thead>
<tr>
<th>Type</th>
<th>Avg. num of potential answerers Before filtering</th>
<th>After filtering</th>
<th>Loss of recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Severe</td>
<td>243,167</td>
<td>19,235 (↓92.09%)</td>
<td>0.24</td>
</tr>
<tr>
<td>Lenient</td>
<td>243,167</td>
<td>137,171 (↓43.59%)</td>
<td>0.14</td>
</tr>
</tbody>
</table>
Contribution of Chapter 5

• Propose a novel QR approach which utilizes category information
  – Category-answerer indexes
  – Basic and transferred category-sensitive language models

• Empirical results
  – Much shorter list of candidate answerers
  – More accurate expertise estimation
Agenda

• Introduction
• Background
• Question Quality Analysis and Prediction (Chapter 3)
• Question Routing
  – Quality and Availability (Chapter 4)
  – Category (Chapter 5)
• Question Structuralization (Chapter 6)
• Conclusion and Future Work
Motivation

List structure (with category hierarchy)  List structure (with social tags)

A Computational Framework for Question Processing in CQA Services
Example: Questions about Edinburgh

A Computational Framework for Question Processing in CQA Services
Question Structuralization

• Introduction to Cluster Entity Tree (CET)
• CET Construction
  – Entity extraction
  – Tree construction
  – Hierarchical entity clustering
• Evaluation
  – User study
  – CET-based question re-ranking
• Summary
Structuralize Questions: Cluster Entity Tree (CET)

1. Where can I buy a hamburger in Edinburgh?
2. Where can I get a shawarma in Edinburgh?
3. How long does it take to drive between Glasgow and Edinburgh?
4. What's the difference between Glasgow and Edinburgh?
5. Good hotels in London and Edinburgh?
6. Looking for nice, clean cheap hotel in Edinburgh?
7. Does anyone know of a reasonably cheap hotel in Edinburgh that is near to Niddry Street South?
8. Who can recommend an affordable hotel in Edinburgh City Center?
Challenges

- Question texts are usually ill-formed
- How to extract named entities with high precision and recall?
- How to efficiently cluster entities?
CET Construction

Entity Extraction

Tree Construction

Hierarchical Entity Clustering

A Computational Framework for Question Processing in CQA Services
Entity Extraction

- Candidate entity extraction
  - Parse each document to a parse tree
  - Extract all noun phrases, stem
  - Find the noun phrases included in our entity repository (NeedleSeek)

- Entropy-based filtering

\[
Entropy(e_i) = - \sum_{c=1}^{|C|} P_c(e_i) \log P_c(e_i)
\]

(total number of categories)

(number of \(e_i\) in category \(c\))

(all number of candidate entities in category \(c\))
Evaluation

- 520 randomly sampled questions, 20 from each top category of Yahoo! Answers

<table>
<thead>
<tr>
<th>Method</th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stanford NER</td>
<td>0.750</td>
<td>0.155</td>
<td>0.257</td>
</tr>
<tr>
<td>FIGER (Ling and Weld, 2012)</td>
<td>0.763</td>
<td>0.154</td>
<td>0.256</td>
</tr>
<tr>
<td>Freebase</td>
<td>0.644</td>
<td>0.595</td>
<td>0.619</td>
</tr>
<tr>
<td>Ours</td>
<td>0.647</td>
<td>0.809</td>
<td>0.719</td>
</tr>
</tbody>
</table>
Tree Construction

- Input: an entity and a set of documents
- Output: a hierarchical entity tree with the given entity as the root
- Method
  - Root node: the given entity + ids of documents containing the entity
  - Layer (1): entities that co-occur with the root entity + corresponding doc ids
  - ...
  - Layer (n): for each entity on layer (n-1) nodes, all entities that co-occur with it and all its superiors + corresponding doc ids
Hierarchical Entity Clustering

- An agglomerative clustering algorithm modified from (Hu et al., 2012)
  - Efficient
  - No need to set the number of clusters
  - Good performance in practice
User Study

- 24 CETs from 70,195 questions
- 12 knowledge-learning tasks and 12 question-search tasks
  - A knowledge-learning task asks for some knowledge about an entity from question texts
    - “find the games running on **macbook pro**”
  - A question-search task asks users to find similar questions
    - “questions about who will win the MVP in **NBA** this year”
User Study

- 16 participants
- List-based program and CET-based program
- A questionnaire after each task
  - Familiarity
  - Easiness
  - Satisfaction
  - Adequate time
  - Helpfulness
  - Comments
## User Study Results

<table>
<thead>
<tr>
<th></th>
<th>Knowledge-learning Tasks</th>
<th>Question-search Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CET-based</td>
<td>List-based</td>
</tr>
<tr>
<td># Queries</td>
<td>2.99</td>
<td>4.47</td>
</tr>
<tr>
<td># Answers</td>
<td>8.32</td>
<td>6.06</td>
</tr>
<tr>
<td>Precision</td>
<td>0.38</td>
<td>0.19</td>
</tr>
<tr>
<td>Time (secs)</td>
<td>136.44</td>
<td>121.87</td>
</tr>
</tbody>
</table>
## Questionnaire Results

<table>
<thead>
<tr>
<th></th>
<th>Knowledge-learning Tasks</th>
<th>Question-search Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CET-based</td>
<td>List-based</td>
</tr>
<tr>
<td>Familiarity</td>
<td>3.18</td>
<td>3.22</td>
</tr>
<tr>
<td>Easiness</td>
<td>3.64</td>
<td>3.66</td>
</tr>
<tr>
<td>Satisfaction</td>
<td>3.70</td>
<td>2.94</td>
</tr>
<tr>
<td>Enough Time</td>
<td>3.87</td>
<td>3.83</td>
</tr>
<tr>
<td>Helpfulness</td>
<td>4.16</td>
<td>3.03</td>
</tr>
</tbody>
</table>

A Computational Framework for Question Processing in CQA Services
CET-based Question Re-Ranking

• Idea
  – Questions sharing similar topics should be ranked similarly
  – Traditional question retrieval models (Cao et al., 2010) cannot capture key semantics
  – By utilizing CET
    • Entities are given more weight while trivial words are not
    • Questions which are ranked lower will be brought higher by their top-ranked neighbors in the same cluster
Problem

Query q: Any **hamburger** to recommend in **Edinburgh**?

**Relevant Questions (Q_q):**

q_1: Any to recommend in **Edinburgh**?
q_2: Can anyone tell me where to buy a **hamburger** in **Edinburgh**?
q_3. Where to get something to eat like **shawarma** in **Edinburgh**? Thank you very much!

A Computational Framework for Question Processing in CQA Services
Step 1: PageRank

Question Collection (Q):
1. Where can I buy a hamburger in Edinburgh?
2. Where can I get a shawarma in Edinburgh?
3. How long does it take to drive between Glasgow and Edinburgh?
4. What is the difference between Glasgow and Edinburgh?
5. Good hotels in London and Edinburgh?
6. Looking for nice, clean, cheap hotel in Edinburgh?
7. Does anyone know of a reasonably cheap hotel in Edinburgh that is near to Niddry Street South?
8. Who can recommend an affordable hotel in Edinburgh City Center?
Step 2: CET Construction

Query q: Any **hamburger** to recommend in **Edinburgh**?
Step 3: CET-based Question Clustering

Relevant Questions ($Q_q$):

$q_1$: Any to recommend in Edinburgh?
$q_2$: Can anyone tell me where to buy a hamburger in Edinburgh?
$q_3$: Where to get something to eat like shawarma in Edinburgh? Thank you very much!

Entity Chains:

- edinburgh
- hamburger
- shawarma

Cluster 1

$q_2$, $q_3$
Step 4: Question Re-ranking

Query $q$: Any hamburger to recommend in Edinburgh?

Relevant Questions ($Q_q$):
- $q_1$: Anything to recommend in Edinburgh?
- $q_2$: Can anyone tell me where to buy a hamburger in Edinburgh?
- $q_3$: Where to get something to eat like shawarma in Edinburgh? Thank you very much!

Re-ranking Results ($Q'_q$):
- $q_2$: Can anyone tell me where to buy a hamburger in Edinburgh? (↑)
- $q_3$: Where to get something to eat like shawarma in Edinburgh? Thank you very much! (↑)
- $q_1$: Anything to recommend in Edinburgh?
Re-ranking Results

(a) Computer & Internet

(b) Travel
Contribution of Chapter 6

- Propose a novel **hierarchical entity-based approach** to structuralize questions in CQA services
- Design a **three-step framework** to construct CETs and show its effectiveness from empirical results
- Demonstrate the great advantages of our approach in knowledge finding
  - User study (User aspect)
  - Question re-ranking (System aspect)
Agenda

• Introduction
• Background
• Question Quality Analysis and Prediction
• Question Routing
  – Quality and Availability
  – Category
• Question Structuralization
• Conclusion and Future Work
Conclusion

- A computational framework for question processing in CQA services
  - Facilitate answerers access to proper questions
  - Help askers obtain information more effectively
  - Improve system’s content organization & QA efficiency
Future Work

• Quality Analysis and Prediction
  – More salient features
  – Question search and recommendation

• Routing
  – Category hierarchy
  – Diversity

• Structuralization
  – Entity normalization
  – Document summarization
THANK YOU!
Questions and comments are welcome and appreciated.
A Computational Framework for Question Processing in CQA Services

Publications

- Question Quality Analysis and Predicting (CQA'12)
- Question Routing (CIKM'10; CIKM'11)
- Question Structuralization (EMNLP'13)
- Question Finding (CIKM'11)
- Expert Finding and Answer Quality Estimation (KAIS, To appear)
- Community Analysis (IJCNN'12)
BACKUP SLIDES (FAQ)

- Chapter 3
- Chapter 4
- Chapter 5
- Chapter 6
A Question’s Life in Yahoo! Answers

1. Post Question
2. Open
3. Get Answers
   - Yes: Select Best Answer
     - Yes: In Voting
       - No: Time Exceeded
         - Yes: Extend Time
           - No: Expired (Deleted)
     - No: Time Exceeded
6. Resolved

Question Analysis and Prediction

• How to set the ground truth of question quality?
• Features
• How to generate user similarity matrix $M$ and question similarity matrix $N$?
• Why MRLP performs better?
• Why using sensitivity/specificity instead of precision/recall?
• Why the performance of MRLP is still not satisfying? How to improve it in the future?
### Rules for the ground truth setting

<table>
<thead>
<tr>
<th>NTA</th>
<th>RM</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

NTA: number of tag-of-interests + number of answers
RM: reciprocal of the minutes for getting the best answer

### Summary of questions in four levels

<table>
<thead>
<tr>
<th>Level</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Count</td>
<td>53,806</td>
<td>62,192</td>
<td>69,836</td>
<td>52,715</td>
</tr>
</tbody>
</table>
Features

• Post-Solving features
  – Used for constructing the ground truth
    • Number of tag-of-interests
    • Number of answers
    • The minutes for getting the best answer

• Pre-Solving features
  – Used for predicting question quality
    • User related features: total points, number of questions asked, etc.
    • Question related features: text length, Wh-words, etc.
Suppose there are $m$ askers who ask $n$ questions in $t$ topics, let $U^1, U^2, ..., U^t$ denote the vectors $(m \times 1)$ of askers’ asking expertise in these topics, and $Q(n \times 1)$ denote the vector of question quality, we define a $m \times n$ matrix $E$, where $e_{ij} = 1(i \in [1, m], j \in [1, n])$ means $u_i$ asks $q_j$, otherwise $e_{ij} = 0$. From $E$ we get $E'$:

$$E'_{ij} = \frac{e_{ij}}{\sum_{k=1}^{n} e_{ik}}.$$ 

For the question part of the bipartite graph, we create edges between any two questions within same topics:

$$w(q_i, q_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{\lambda^2_q}\right)$$

$$N_{ij} = P(q_i \rightarrow q_j) = \frac{w(q_i, q_j)}{\sum_{k=1}^{n} w(q_i, q_k)}$$

For the asker part of the bipartite graph, we generate the probabilistic transition matrix $M$ similarly.
MRLP VS Others

• It models the interaction between askers and topics explicitly
• It captures the mutual reinforcement relationship between asking expertise and question quality
Sensitivity & Specificity

- Sensitivity measures the algorithm’s ability to identify **high-quality** questions (=recall)
- Specificity measures the algorithm’s ability to identify **low-quality** questions
- Precision and recall focus on positive instances

<table>
<thead>
<tr>
<th>Condition (as determined by &quot;Gold standard&quot;)</th>
<th>Test outcome positive</th>
<th>Test outcome negative</th>
</tr>
</thead>
<tbody>
<tr>
<td>True positive</td>
<td>False positive (Type I error)</td>
<td>True negative</td>
</tr>
<tr>
<td>False negative (Type II error)</td>
<td>True negative</td>
<td></td>
</tr>
</tbody>
</table>

- **Precision** = \( \frac{\Sigma \text{True positive}}{\Sigma \text{Test outcome positive}} \)
- **Negative predictive value** = \( \frac{\Sigma \text{True negative}}{\Sigma \text{Test outcome negative}} \)
- **Sensitivity** = \( \frac{\Sigma \text{True positive}}{\Sigma \text{Condition positive}} \)
- **Specificity** = \( \frac{\Sigma \text{True negative}}{\Sigma \text{Condition negative}} \)
- **Accuracy**
Discussion

• MRLP is more effective in distinguishing high quality questions from low quality ones than state-of-the-art methods

• At present, neither MRLP nor other methods achieves satisfactory performance due to the influence of features
Discussion

- **Salient features?**
  - User study via crowdsourcing systems

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>IG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sub_len</td>
<td>Number of words in question subject (title)</td>
<td>0.0115</td>
</tr>
<tr>
<td>Con_len</td>
<td>Number of words in question content</td>
<td>0.0029</td>
</tr>
<tr>
<td>Wh-type</td>
<td>Whether the question subject starts with Wh-word (e.g., “what”, “where”, etc.)</td>
<td>0.0001</td>
</tr>
<tr>
<td>Sub_punc_den</td>
<td>Number of question subject’s punctuation over length</td>
<td>0.0072</td>
</tr>
<tr>
<td>Sub_typo_den</td>
<td>Number of question subject’s typos over length</td>
<td>0.0021</td>
</tr>
<tr>
<td>Sub_space_den</td>
<td>Number of question subject’s spaces over length</td>
<td>0.0138</td>
</tr>
<tr>
<td>Con_punc_den</td>
<td>Number of question content’s punctuation over length</td>
<td>0.0096</td>
</tr>
<tr>
<td>Con_typo_den</td>
<td>Number of question content’s typos over length</td>
<td>0.0006</td>
</tr>
<tr>
<td>Con_space_den</td>
<td>Number of question content’s spaces over length</td>
<td>0.0113</td>
</tr>
<tr>
<td>Avg_word</td>
<td>Number of words per sentence in question’s subject and content</td>
<td>0.0048</td>
</tr>
<tr>
<td>Cap_error</td>
<td>The fraction of sentences which are started with a small letter</td>
<td>0.0064</td>
</tr>
<tr>
<td>POS_entropy</td>
<td>The entropy of the part-of-speech tags of the question</td>
<td>0.0004</td>
</tr>
<tr>
<td>NF_ratio</td>
<td>The fraction of words that are not the top 10 frequent words in the collection</td>
<td>0.0009</td>
</tr>
<tr>
<td>Total_points</td>
<td>Total points the asker earns</td>
<td>0.0339</td>
</tr>
<tr>
<td>TotalAnswers</td>
<td>Number of answers the asker provided</td>
<td>0.0436</td>
</tr>
<tr>
<td>Best_answers</td>
<td>Number of best answers the asker provided</td>
<td>0.0331</td>
</tr>
<tr>
<td>Total_questions</td>
<td>Number of questions the asker provided</td>
<td>0.0339</td>
</tr>
<tr>
<td>Resolved_questions</td>
<td>Number of resolved questions asked by the asker</td>
<td>0.0357</td>
</tr>
<tr>
<td>Star_received</td>
<td>Number of stars received for all questions</td>
<td>0.0367</td>
</tr>
</tbody>
</table>
Question Routing

• Statistic of tracked data
• Details of the Basic Model and the Smoothed Model for expertise estimation
• Why integrate expertise score and availability score directly?
• Experimental setup
• Impact of $\beta$
Tracked Data

- Many askers cannot get satisfied answers in time

<table>
<thead>
<tr>
<th></th>
<th># resolved questions</th>
<th># unresolved questions with at least one answer</th>
<th># unresolved questions without answer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yahoo! Answers</td>
<td>527</td>
<td>1,820</td>
<td>442</td>
</tr>
<tr>
<td>Baidu Zhidao</td>
<td>682</td>
<td>1,325</td>
<td>993</td>
</tr>
</tbody>
</table>

- Answerers have to find questions manually
Expertise Estimation

• Basic Model

\[ Q_{BM}(u_i, q_r) = \frac{\sum_{q_j \sim u_i} Q(u_i, q_j) \cdot \text{sim}(q_j, q_r)}{\sum_{q_j \sim u_i} \text{sim}(q_j, q_r)} \]

\[ w_{qt} = t_{f_i,q} \times \log \frac{N}{df_t} \]

• Smoothed Model

\[ Q_{SM}(u_i, q_r) = \beta Q_{BM}(u_i, q_r) + (1 - \beta) \frac{\sum_{u_j \in U/u_i} \sum_{q_k \sim u_j} Q(u_j, q_k) \cdot \text{sim}(Q_{u_jq_k}, Q_{u_iq_r})}{\sum_{u_j \in U/u_i} \sum_{q_k \sim u_j} \text{sim}(Q_{u_jq_k}, Q_{u_iq_r})} \]

\[ \text{sim}(Q(u_j, q_k), Q(u_i, q_r)) = \frac{1}{\sqrt{\frac{1}{\text{sim}(u_i, u_j)^2} + \frac{1}{\text{sim}(q_k, q_r)^2}}} \]
## Example

<table>
<thead>
<tr>
<th></th>
<th>$q_1$</th>
<th>$q_2$</th>
<th>$q_3$</th>
<th>$q_4$</th>
<th>$q_{\text{new}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_1$</td>
<td></td>
<td>0.7</td>
<td></td>
<td></td>
<td>?</td>
</tr>
<tr>
<td>$u_2$</td>
<td></td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$u_3$</td>
<td>0.9</td>
<td></td>
<td></td>
<td>0.8</td>
<td></td>
</tr>
<tr>
<td>$u_4$</td>
<td></td>
<td></td>
<td>0.6</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Experimental Setup

• Data
  – Yahoo! Answers data (April 6, 2010 - May 14, 2010)
    • Objective: Predict the answerers of the questions posted after May 6, 2010
    • Training set: 17,182 questions, 48,663 answers and 16,298 answerers
    • Testing set: 1,713 questions, 5,403 answers and 2,891 answerers
    • Features: 7 answer-related and 5 user-related features

• Evaluation Metric
  – Mean Reciprocal Rank (MRR)
    \[ MRR = \frac{1}{|Q|} \sum_{i=1}^{Q} \frac{1}{\text{rank}_i} \]
Integration of Expertise Score and Availability Score

• High expertise score doesn’t mean high availability score
• An active answerers doesn’t necessary obtain high expertise score (when considering answer quality)
• Expertise and availability are not totally independent
Impact of $\beta$

The MRR value of Smoothed Q versus various $\beta$
Category-sensitive QR

- Importance of category: an example
- Difference between question routing and question retrieval
- An example of category-answerer indexes
- Impact of user prior \( P(u) \) in language models
- Transferred probabilities between leaf categories
- Impact of \( \delta \) on TCS-LM (Content VS User)
- LDA
- Data set statistics
- Definitions of evaluation metrics
  - Prec@K
  - MRR
  - MAP
One Example

• Alex, a senior Java programmer, is an active answer in Yahoo! Answers. He has answered more than 1,000 questions in terms of **Java programming** as well as 100 questions about **Java coffee**.

• Bob, a cafe manager, is also a frequent user of Yahoo! Answers. He answered around **300 questions about Java coffee**, but he knows little about Java programming.

• Carl, a college student, now asks a question “I met a problem in making Java, any ideas” in “**Food & Drink**” category.
Question Routing and Question Retrieval

- **Question routing**
  - Steps
    - User Profiling
    - Question Profiling
    - Matching
  - Models for user and question profiling
    - Topic Model based, Language Model based, Classification-based, Diversity and Freshness aided, etc.

- **Question retrieval**
  - Models
    - language model, Translation-based Language model, VSM, BN25, etc.
Category-Answerer Indexes

A Computational Framework for Question Processing in CQA Services
Impact of User Prior

- Uniform distribution (Liu et al., 2004)
- In-degree (Bouguessa et al., 2008)

Prec@K of LM (left) and BCS-LM (right) with different answerer priors
Latent Dirichlet Allocation

Diagram showing the model with variables α, θ, Z, W, β, φ, K, N, M.
# Dataset

<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of questions</td>
<td>433,072</td>
</tr>
<tr>
<td>Number of answers</td>
<td>1,510,531</td>
</tr>
<tr>
<td>Average number of answers for one question</td>
<td>3.49</td>
</tr>
<tr>
<td>Maximum number of answers for one question</td>
<td>50</td>
</tr>
<tr>
<td>Mean first reply duration (in minutes)</td>
<td>197.32</td>
</tr>
<tr>
<td>Average question length in words (both subject and content)</td>
<td>43.87</td>
</tr>
<tr>
<td>Average answer length in words</td>
<td>30.08</td>
</tr>
<tr>
<td>Number of askers</td>
<td>240,277</td>
</tr>
<tr>
<td>Number of answerers</td>
<td>270,043</td>
</tr>
<tr>
<td>Number of both askers and answerers</td>
<td>68,551</td>
</tr>
<tr>
<td>Number of askers only</td>
<td>171,726</td>
</tr>
<tr>
<td>Number of answerers only</td>
<td>201,492</td>
</tr>
</tbody>
</table>
Precision at K (Prec@K): For a set of new questions $Q_r$, $Prec@K$ reports the fraction of successful QR when top $K$ answerers of the ranking list are returned. The criteria of a successful QR, in the present study, is defined as at least one answerer in the top $K$ of the ranking list actually answered the routed question. In this metric, the position of these users is not considered. The only key factor is whether there is at least one user in these $K$ candidates who answered the routed question. $Prec@K$ is calculated as:

$$Prec@K = \frac{\sum_{q_r \in Q_r} S(q_r, K)}{|Q_r|},$$

$$S(q_r, K) = \begin{cases} 
1, & \text{if QR for } q_r \text{ is successful;} \\
0, & \text{otherwise.}
\end{cases}$$
Mean Reciprocal Rank (MRR): The reciprocal rank for an individual question $q_r$ is the reciprocal of the rank at which the first user in the ranking list who actually answered $q_r$, or 0 if none of the users in the list answered $q_r$. The MRR value for a set of questions is the mean of each question’s reciprocal rank. It is defined as:

$$MRR = \frac{1}{|Q_r|} \sum_{q_r \in Q_r} \frac{1}{Rank(q_r)},$$

where $Q_r$ is a set of questions to be routed, $Rank(q_r)$ is the rank of the first user who actually answered $q_r$ in the ranking list.
Mean Average Precision (MAP): For a set of new questions $Q_r$, MAP measures the mean of the average precision for each question $q_r$ in $QR$:

$$MAP = \frac{\sum_{q_r \in Q_r} AvgP(q_r)}{|Q_r|},$$

$$AvgP(q_r) = \frac{\sum_{k=1}^{N_r} (P_r(k) \cdot IsAns(k))}{NRA_r},$$

$$P_r(k) = \frac{NRA_r(k)}{k},$$

where $Q_r$ is a set of questions to be routed, $N_r$ is the number of potential answerers for $q_r$ generated from answerer filtering, $NRA_r$ is the number of real answerers for $q_r$. $IsAns(k)$ is a binary function to denote whether the $k_{th}$ answerer actually answered $q_r$, and $NRA_r(k)$ denote the number of real answerers in top $k$ ranked answerers for $q_r$. 
Transferred Probabilities (Example)

Table 2 Transferred probabilities between partial leaf categories (answerer-based method)

<table>
<thead>
<tr>
<th>From</th>
<th>To</th>
<th>Software</th>
<th>Printers</th>
<th>Comedy</th>
<th>Lyrics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programming &amp; Design</td>
<td></td>
<td>0.2975</td>
<td>0.0251</td>
<td>0.0026</td>
<td>0.0026</td>
</tr>
<tr>
<td>Scanners</td>
<td></td>
<td>0.1158</td>
<td>0.5604</td>
<td>0.0014</td>
<td>0.0008</td>
</tr>
<tr>
<td>Drama</td>
<td></td>
<td>0.0053</td>
<td>0.0006</td>
<td>0.2593</td>
<td>0.0137</td>
</tr>
<tr>
<td>Other - Music</td>
<td></td>
<td>0.0102</td>
<td>0.0019</td>
<td>0.0273</td>
<td>0.1683</td>
</tr>
</tbody>
</table>

Table 3 Transferred probabilities between partial leaf categories (content-based method)

<table>
<thead>
<tr>
<th>From</th>
<th>To</th>
<th>Software</th>
<th>Printers</th>
<th>Comedy</th>
<th>Lyrics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programming &amp; Design</td>
<td></td>
<td>0.2250</td>
<td>0.0236</td>
<td>0.0116</td>
<td>0.0116</td>
</tr>
<tr>
<td>Scanners</td>
<td></td>
<td>0.1676</td>
<td>0.2671</td>
<td>0.0049</td>
<td>0.0034</td>
</tr>
<tr>
<td>Drama</td>
<td></td>
<td>0.0136</td>
<td>0.0020</td>
<td>0.5481</td>
<td>0.0376</td>
</tr>
<tr>
<td>Other - Music</td>
<td></td>
<td>0.0443</td>
<td>0.0070</td>
<td>0.0748</td>
<td>0.2922</td>
</tr>
</tbody>
</table>
Impact of $\delta$

MRR for TCS-LM using answerer-based and content-based approaches to estimate transferring probability under GT-A

A Computational Framework for Question Processing in CQA Services
Question Structuralization

- Why adopt entity-based approach for question structuralization?
- Definitions of ER and CET
- Tree construction example
- Detail of clustering algorithm
- What is the similarity function for clustering?
- How to evaluate the clustering results?
- Detail of category mapping
- Definition of B-Cubed Metrics
- What is the usage of Set EC?
- Program interface
- User study tasks
Structuralize Questions: Review

- Predefined category hierarchy
  - Coarse grained
  - Hard to maintain
- Topic models
  - Not trivial to control the granularity of topics (Chen et al., 2011).
  - Interpretation problem
- Social tagging
  - Not widely applicable
  - Sparsity (Shepitsen et al., 2008)
Advantages of CET

- CET avoids the granularity, interpretation, and sparsity problems by utilizing a large-scale entity repository
  - Entity repository contains millions of named entities on various topics
  - Usually give descriptions of entities
- Automatically build semantic hierarchy
  - Flexible & easy to maintain
Definitions

- **Entity repository**
  - \( \text{ER} = \{ R, g \} \)
  - \( R \) is a set of named entities
  - \( g \) is a mapping function that defines the similarity of any two entities

- **Cluster Entity Tree (CET)**
  - \( \text{CET}_e = (v_e, V, E, C) \) is a tree structure
  - Each node \( v_s \in V \) on \( \text{CET}_e \) includes
    - An entity extracted from the set of documents \( D_e \in D \) containing \( e \)
    - A list \( L(s) \) which stores the indexes of documents containing entity \( s \) and its superior entities
  - If \( v_s \) is \( v_t \)'s parent node, entity \( t \) must co-occur with \( s \) and \( s \)'s all superior entities at least once
  - Each \( c \in C \) includes a set of similar nodes which share the same parent node
Tree Construction Example

1. Where can I buy a hamburger in Edinburgh?
2. Where can I get a shawarma in Edinburgh?
3. How long does it take to drive between Glasgow and Edinburgh?
4. What's the difference between Glasgow and Edinburgh?
5. Good hotels in London and Edinburgh?
6. Looking for nice, clean cheap hotel in Edinburgh?
7. Does anyone know of a reasonably cheap hotel in Edinburgh that is near to Niddry Street South?
8. Who can recommend an affordable hotel in Edinburgh City Center?

Root entity: edinburgh
**Modified Agglomerative Clustering**

**Input:** a set of entities with the same parent  
**Output:** clusters of entities

- Select one entity and create a new cluster which contains the entity
- Select the next entity $e_i$, calculate the similarity between the entity and all existing clusters
- Find $\arg\max \sim(e_i, c)$, s.t. $\sim(e_i, c) > \theta$; otherwise, create a new cluster with $e_i$ as the element
- Stop when all entities are clustered
Hierarchical Entity Clustering: Similarity Function

• Follow the approach in (Shi et al., 2010)
  – First-order co-occurrence: Pattern-based (PB)
  – Second-order co-occurrence: Distributional similarity (DS)

• PB
  – The set of terms extracted by applying a pattern one time is called a raw semantic class (RASC)
  – Given two entities a and b, calculate their similarity based on the number of RASCs containing both of them

• DS
  – Terms appearing in similar contexts tend to be similar
  – Given two entities a and b, calculate the similarity between their corresponding context feature vectors

If at least one entity is proper noun, PB is employed; otherwise DS is used.
PB

- Some well-designed patterns are leveraged to extract similar entities from a huge repository of webpages. The set of terms extracted by applying a pattern one time is called a raw semantic class (RASC).
- Given two entities $t_a$ and $t_b$, PB calculates their similarity based on the number of RASCs containing both of them (Zhang et al., 2009).

$$Sim(t_a, t_b) = \log(1 + \sum_{i=1}^{r_{ab}} P_{ab_i}) \cdot \sqrt{idf(t_a) \cdot idf(t_b)},$$

where $idf(t_a) = \log(1 + \frac{N}{C(t_a)})$, $P_{ab_i}$ is a pattern which can generate RASC(s) containing both term $t_a$ and term $t_b$, $r_{ab}$ is the total number of such patterns, $N$ is the total number of RASCs, and $C(t_a)$ is the number of RASCs containing $t_a$.

$$Sim_{PB}(t_a, t_b) = \frac{\log Sim(t_a, t_b)}{2 \log Sim(t_a, t_a)} + \frac{\log Sim(t_a, t_b)}{2 \log Sim(t_b, t_b)}$$
A term is represented by a feature vector, with each feature corresponding to a context in which the term appears.

The similarity between two terms is computed as the similarity between their corresponding feature vectors. Jaccard similarity is employed to estimate the similarity between two terms.

Suppose the feature vectors of \( t_a \) and \( t_b \) are \( x \) and \( y \) respectively:

\[
Sim_{DS}(t_a, t_b) = \frac{\sum_i \min(x_i, y_i)}{\sum_i (x_i) + \sum_i (y_i) - \sum_i \min(x_i, y_i)}
\]
Clustering Evaluation

• 8M questions from 4 top categories of Yahoo! Answers

• Ground truth setting
  – Map categories among YA and Freebase
  – Extract entities which appear exactly once in the corresponding Freebase categories
  – Attach each entity with a unique Freebase category label

• Three approaches
  – AC-MAX, AC-MIN, and AC-AVG
  – AC-MAX performs the best (F1 > 0.75)
Clustering Evaluation

Clustering results using AC-MAX ($\theta_{\text{max}}=0.1$)

<table>
<thead>
<tr>
<th>Level</th>
<th>Travel</th>
<th>Count</th>
<th>P</th>
<th>R</th>
<th>F1</th>
<th>Cars &amp; Transportation</th>
<th>Count</th>
<th>P</th>
<th>R</th>
<th>F1</th>
<th>Computer &amp; Internet</th>
<th>Count</th>
<th>P</th>
<th>R</th>
<th>F1</th>
<th>Sports</th>
<th>Count</th>
<th>P</th>
<th>R</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>748</td>
<td>0.972</td>
<td>0.653</td>
<td>0.743</td>
<td>1281</td>
<td>0.948</td>
<td>0.868</td>
<td>0.897</td>
<td>3064</td>
<td>0.913</td>
<td>0.664</td>
<td>0.743</td>
<td>890</td>
<td>0.941</td>
<td>0.883</td>
<td>0.901</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>200</td>
<td>0.974</td>
<td>0.730</td>
<td>0.798</td>
<td>1202</td>
<td>0.989</td>
<td>0.956</td>
<td>0.965</td>
<td>11344</td>
<td>0.961</td>
<td>0.842</td>
<td>0.879</td>
<td>636</td>
<td>0.978</td>
<td>0.964</td>
<td>0.963</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>120</td>
<td>1.000</td>
<td>0.833</td>
<td>0.890</td>
<td>858</td>
<td>1.000</td>
<td>0.981</td>
<td>0.988</td>
<td>8184</td>
<td>0.978</td>
<td>0.899</td>
<td>0.920</td>
<td>492</td>
<td>0.965</td>
<td>0.882</td>
<td>0.899</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>1776</td>
<td>1.000</td>
<td>0.980</td>
<td>0.986</td>
<td>3648</td>
<td>0.990</td>
<td>0.908</td>
<td>0.934</td>
<td>1080</td>
<td>0.978</td>
<td>0.844</td>
<td>0.881</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>2520</td>
<td>1.000</td>
<td>0.952</td>
<td>0.968</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>1068</td>
<td>0.976</td>
<td>0.688</td>
<td>0.770</td>
<td>5117</td>
<td>0.984</td>
<td>0.946</td>
<td>0.959</td>
<td>28760</td>
<td>0.968</td>
<td>0.857</td>
<td>0.891</td>
<td>3098</td>
<td>0.965</td>
<td>0.886</td>
<td>0.907</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Category Mapping

• Goal: automatically evaluate clustering
  – Each entity is attached with a unique Freebase category label

• Two experts are asked to conduct category mapping from Yahoo! Answers to Freebase

<table>
<thead>
<tr>
<th>Yahoo! Answers</th>
<th>FreeBase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cars &amp; Transportation</td>
<td>Aviation, Transportation, Boats</td>
</tr>
<tr>
<td></td>
<td>Spaceflight, Automotive, Bicycles, Rail</td>
</tr>
<tr>
<td>Computers &amp; Internet</td>
<td>Computer, Internet</td>
</tr>
<tr>
<td>Sports</td>
<td>Soccer, Olympics, Sports, American football,</td>
</tr>
<tr>
<td></td>
<td>Baseball, Basketball, Ice Hockey, Martial Arts,</td>
</tr>
<tr>
<td></td>
<td>Cricket, Tennis, Boxing, Skiing</td>
</tr>
<tr>
<td>Travel</td>
<td>Travel, Location, Transportation</td>
</tr>
</tbody>
</table>
## Set EC

<table>
<thead>
<tr>
<th>Category</th>
<th>Number of Questions</th>
<th>Number of Entities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cars &amp; Transportation</td>
<td>1,220,427</td>
<td>3,267,596</td>
</tr>
<tr>
<td>Computers &amp; Internet</td>
<td>2,912,280</td>
<td>7,324,655</td>
</tr>
<tr>
<td>Sports</td>
<td>2,363,758</td>
<td>6,230,868</td>
</tr>
<tr>
<td>Travel</td>
<td>1,347,801</td>
<td>3,728,286</td>
</tr>
</tbody>
</table>
B-Cubed Metrics

- B-Cubed precision of an item is the proportion of items in its cluster which have the item's category (including itself).
- The overall B-Cubed precision is the averaged precision of all items.

Figure: Example of computing the BCubed precision and recall for one item.
Interface

Figure: The interface of CET-based program
## User Study Tasks

<table>
<thead>
<tr>
<th>ID</th>
<th>Task Title</th>
<th>Category</th>
<th>Main Entity</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Find the names of games running on macbook pro</td>
<td>Computer &amp; Internet</td>
<td>Macbook Pro</td>
<td>E</td>
</tr>
<tr>
<td>2</td>
<td>Find which components of thinkpad notebooks are usually asked</td>
<td>Computer &amp; Internet</td>
<td>Thinkpad</td>
<td>E</td>
</tr>
<tr>
<td>3</td>
<td>How to ps body using photoshop cs2</td>
<td>Computer &amp; Internet</td>
<td>Photoshop CS2</td>
<td>E</td>
</tr>
<tr>
<td>4</td>
<td>Questions about the best canon laser printer for a mac</td>
<td>Computer &amp; Internet</td>
<td>Laser printer</td>
<td>S</td>
</tr>
<tr>
<td>5</td>
<td>Questions about how to connect Xbox 360 to Laptop or PC using a router</td>
<td>Computer &amp; Internet</td>
<td>Xbox 360</td>
<td>S</td>
</tr>
<tr>
<td>6</td>
<td>Questions about green screen problem of Windows Movie Maker</td>
<td>Computer &amp; Internet</td>
<td>Windows Movie Maker</td>
<td>S</td>
</tr>
<tr>
<td>7</td>
<td>Find the cities compared with Edinburgh</td>
<td>Travel</td>
<td>Edinburgh</td>
<td>E</td>
</tr>
<tr>
<td>8</td>
<td>Find the names of animals on myrtle beach</td>
<td>Travel</td>
<td>Myrtle Beach</td>
<td>E</td>
</tr>
<tr>
<td>9</td>
<td>Find the names of cities in Portugal</td>
<td>Travel</td>
<td>Portugal</td>
<td>E</td>
</tr>
<tr>
<td>10</td>
<td>Questions about looking for good hostels in Madrid</td>
<td>Travel</td>
<td>Madrid</td>
<td>S</td>
</tr>
<tr>
<td>11</td>
<td>Questions about how to get a low price ticket to Hong Kong Disneyland</td>
<td>Travel</td>
<td>Disneyland</td>
<td>S</td>
</tr>
<tr>
<td>12</td>
<td>Questions about how to go to Chinatown in Chicago</td>
<td>Travel</td>
<td>Chicago</td>
<td>S</td>
</tr>
<tr>
<td>13</td>
<td>Find the brand of running shoes that users have asked</td>
<td>Sports</td>
<td>Running shoes</td>
<td>E</td>
</tr>
<tr>
<td>14</td>
<td>Find football players that compared with messi</td>
<td>Sports</td>
<td>Messi</td>
<td>E</td>
</tr>
<tr>
<td>15</td>
<td>Find the names of skiing places that users have asked</td>
<td>Sports</td>
<td>Skiing</td>
<td>E</td>
</tr>
<tr>
<td>16</td>
<td>Questions asking horse racing website</td>
<td>Sports</td>
<td>Horse racing</td>
<td>S</td>
</tr>
<tr>
<td>17</td>
<td>Questions about who will win the MVP in NBA this year</td>
<td>Sports</td>
<td>NBA</td>
<td>S</td>
</tr>
<tr>
<td>18</td>
<td>Questions about when is the next match between Barcelona and Real Madrid</td>
<td>Sports</td>
<td>Real Madrid</td>
<td>S</td>
</tr>
<tr>
<td>19</td>
<td>Find the brand of cars that have been compared with Toyota</td>
<td>Cars &amp; Transportation</td>
<td>Toyota</td>
<td>E</td>
</tr>
<tr>
<td>20</td>
<td>Which aspects of Jeep Wrangler have been asked</td>
<td>Cars &amp; Transportation</td>
<td>Jeep Wrangler</td>
<td>E</td>
</tr>
<tr>
<td>21</td>
<td>Finding the names of sports cars being asked</td>
<td>Cars &amp; Transportation</td>
<td>Sports cars</td>
<td>E</td>
</tr>
<tr>
<td>22</td>
<td>Questions which compare Mercedes Benz and BMW</td>
<td>Cars &amp; Transportation</td>
<td>Mercedes Benz</td>
<td>S</td>
</tr>
<tr>
<td>23</td>
<td>Questions about the price to tow a suv from Newark to Florida</td>
<td>Cars &amp; Transportation</td>
<td>SUV</td>
<td>S</td>
</tr>
<tr>
<td>24</td>
<td>Questions about How to reset the oil light for a 95 Honda civic</td>
<td>Cars &amp; Transportation</td>
<td>Honda Civic</td>
<td>S</td>
</tr>
</tbody>
</table>