Template-guided Clarifying Question Generation for Web Search Clarification

Jian Wang and Wenjie Li
Department of Computing,
The Hong Kong Polytechnic University
Web Search

(Adapted from https://www.seofuelmarketing.com/basic-differences-google-bing/)

Problems?
Problems?

• Search queries are often short, the underlying user intents are often ambiguous.
• It’s challenging for Web search engines to return the appropriate results that pertain to the users’ actual information needs.

(Adapted from https://www.seofuelmarketing.com/basic-differences-google-bing/)
Web Search Clarification

User Query: convert string to int

Clarifying Question: What programming language are you looking for?

Search Engine Result Page (SERP):

- Java Convert String to int - javatpoint
  https://www.javatpoint.com/java-string-to-int
  Java Convert String to int. We can convert String to an int in java using Integer.parseInt() method. To convert String into Integer, we can use Integer.valueOf( )..
  Java Char to Int - Java String valueOf

- How to convert string to int in C#? - TutorialsTeacher
  https://www.tutorialsteacher.com/articles/convert-string-to-int
  Sep 25, 2019 - The TryParse() methods are available for all the primitive types to convert string to the calling data type. It is the recommended way to convert string to an...

- How to convert a string to a number - C# Programming Guide ...
  https://docs.microsoft.com/en-us/dotnet/csharp/
  Feb 16, 2021 - The following example calls the Convert.ToInt32(String) method to convert an Input string to an Int. The example catches the two most common exceptions that can be...
  decimal: ToDecimal (String) double: ToDouble (String)
  float: ToSingle (String) short: ToInt16 (String)

- Java String to Int - How to Convert a String to an Integer
  https://www.freecodecamp.org/news/java-string-to...
  Nov 23, 2020 - Use Integer.parseInt() to Convert a String to an Integer. This method returns the string as a primitive type int. If the string does not contain a valid...
  Estimated Reading Time: 4 mins

Answer Options:

- Java
- Python
- C++
- C
- Javascript
- Swift
- VBA
- Matlab

Videos of Convert String to Int

- Bing.com/videos
  "17" - 17 Ways to Convert String to an Int...
  21K views - Jan 4, 2021
  YouTube - TalentLo
Web Search Clarification

Problem Formulation:

Let $Q = \{q_1, q_2, \ldots, q_n\}$ be the set of user queries, for each query $q_i (1 \leq i \leq n)$, let $S_{q_i} = \{s^1_{q_i}, s^2_{q_i}, \ldots, s^m_{q_i}\}$ denote top-$m$ search engine result pages (SERP) in response to $q_i$, where the content of each $s^j_{q_i} (1 \leq j \leq m)$ is the snippet of the Web page returned by the search engine.

Given a user query $q_i$ and SERP snippets $S_{q_i}$, the task of Web search clarification is to automatically ask a clarifying question $c_i$ with the intention of clarifying the user’s ambiguous information need.
Web Search Clarification

Challenges?

• **Generative method.** For sequence-to-sequence methods (Sutskever et al., 2014; Bahdanau et al., 2015) to generate clarifying questions directly, they can hardly well capture the **intra-semantics** of each SERP and the **inter-patterns** between different SERPs, which are crucial for what is to be clarified.
Web Search Clarification

Challenges?

• **Generative method.** For sequence-to-sequence methods (Sutskever et al., 2014; Bahdanau et al., 2015) to generate clarifying questions directly, they can hardly well capture the *intra-semantics* of each SERP and the *inter-patterns* between different SERPs, which are crucial for what is to be clarified.

• **Retrieval method.** The bottleneck is to select the most appropriate one clarifying question from a *large pool of question candidates* with high *efficiency*. 
Our Motivation

- Clarifying questions often follow a few types of templates according to their purposes like disambiguation, comparison, asking for preference, or asking for sub-topic information.

What (.+) are you looking for?

What *programming language* are you looking for?

*programming language*
Our Motivation

• Clarifying questions often follow a few types of templates according to their purposes like disambiguation, comparison, asking for preference, or asking for sub-topic information.

• Preliminary statistical analysis reveals that common question templates can match over 95% of the clarifying questions.
Our Method

- A simple yet effective template-guided clarifying question (TG-ClariQ) generation model, which employs Transformer (Vaswani et al., 2017) to enable deep interactions between user queries and SERP contents.

- Jointly learning to select the question template from a list of template candidates and fill in the question slot from a slot vocabulary.
Our Method

Template ranking score

Feed Forward

Slot distribution

Softmax

Mean Pooling & Concatenate

Add & Norm

Point-wise Feed Forward

Add & Norm

Point-wise Feed Forward

Add & Norm

Cross Attention

Add & Norm

Self Attention

Add & Norm

Self Attention

Text Encoder

Text Encoder

user query + SERP snippets

candidate question template

$N \times$
Datasets

We use the MIMICS (Zamani et al., 2020) data collection.

• We extract <query, clarifying question> pairs, each pair is associated with at most top-10 SERP snippets returned by the Bing’s search API.

• Training/validation/testing: 38,508/1000/1000 samples

• We obtain 8 question templates in total, which cover all clarifying questions in the samples.

<table>
<thead>
<tr>
<th>ID</th>
<th>Clarifying question template</th>
<th>#train</th>
<th>#dev</th>
<th>#test</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1</td>
<td>select one to refine your search</td>
<td>12,000</td>
<td>325</td>
<td>308</td>
</tr>
<tr>
<td>T2</td>
<td>what (do you want</td>
<td>would you like) to know about (.+)?</td>
<td>10,662</td>
<td>230</td>
</tr>
<tr>
<td>T3</td>
<td>(which</td>
<td>what) (.+) do you mean?</td>
<td>8,607</td>
<td>147</td>
</tr>
<tr>
<td>T4</td>
<td>(what</td>
<td>which) (.+) are you looking for?</td>
<td>4,645</td>
<td>130</td>
</tr>
<tr>
<td>T5</td>
<td>what (do you want</td>
<td>would you like) to do with (.+)?</td>
<td>1,988</td>
<td>89</td>
</tr>
<tr>
<td>T6</td>
<td>who are you shopping for?</td>
<td>300</td>
<td>40</td>
<td>37</td>
</tr>
<tr>
<td>T7</td>
<td>what are you trying to do?</td>
<td>227</td>
<td>30</td>
<td>29</td>
</tr>
<tr>
<td>T8</td>
<td>do you have any (specific</td>
<td>particular) (.+) in mind?</td>
<td>79</td>
<td>9</td>
</tr>
</tbody>
</table>
Baseline Methods

• **Clarifying question selection (CQS):**
  Extract clarifying questions from a large pool of candidate questions

• **Clarifying template selection (CTS):**
  Directly select clarifying question templates

• **Clarifying question generation (CQG):**
  Generate clarifying questions in an end-to-end manner
## Experimental Results

<table>
<thead>
<tr>
<th>Methods</th>
<th>Accuracy</th>
<th>MRR@3</th>
<th>BLEU</th>
<th>Entity F1</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CQS</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BM25</td>
<td>0.355</td>
<td>0.399</td>
<td>n.a.</td>
<td>0.414</td>
</tr>
<tr>
<td>RankNet</td>
<td>0.308</td>
<td>0.384</td>
<td>n.a.</td>
<td>0.203</td>
</tr>
<tr>
<td>LambdaMART</td>
<td>0.490</td>
<td>0.564</td>
<td>n.a.</td>
<td>0.214</td>
</tr>
<tr>
<td>BERT</td>
<td>0.394</td>
<td>0.440</td>
<td>n.a.</td>
<td>0.356</td>
</tr>
<tr>
<td><strong>CTS</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BM25</td>
<td>0.095</td>
<td>0.191</td>
<td>n.a.</td>
<td>n.a.</td>
</tr>
<tr>
<td>RankNet</td>
<td>0.323</td>
<td>0.455</td>
<td>n.a.</td>
<td>n.a.</td>
</tr>
<tr>
<td>LambdaMART</td>
<td>0.564</td>
<td>0.621</td>
<td>n.a.</td>
<td>n.a.</td>
</tr>
<tr>
<td>BERT</td>
<td>0.676</td>
<td>0.794</td>
<td>n.a.</td>
<td>n.a.</td>
</tr>
<tr>
<td><strong>CQG</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seq2Seq-LSTM</td>
<td>n.a.</td>
<td>n.a.</td>
<td>45.30</td>
<td>0.166</td>
</tr>
<tr>
<td>Seq2Seq-LSTM+Copy</td>
<td>n.a.</td>
<td>n.a.</td>
<td>52.64</td>
<td>0.495</td>
</tr>
<tr>
<td>Seq2Seq-Transformer+Copy</td>
<td>n.a.</td>
<td>n.a.</td>
<td>55.37</td>
<td>0.546</td>
</tr>
<tr>
<td><strong>TG-ClariQ</strong>-LSTM</td>
<td>0.659</td>
<td>0.791</td>
<td>55.05</td>
<td>0.682</td>
</tr>
<tr>
<td>TG-ClariQ-BERT</td>
<td>*<em>0.722</em></td>
<td>*<em>0.827</em></td>
<td>*<em>60.49</em></td>
<td>*<em>0.788</em></td>
</tr>
</tbody>
</table>
Discussions

• If there are two or more slots that need to be filled in a question template, our model can be extended by adding additional slot generation layers and designing extra strategies to determine the order of slot filling. (Due to the single slot nature of the dataset, we leave this as a direction for future investigation.)
Discussions

• If there are two or more slots that need to be filled in a question template, our model can be extended by adding additional slot generation layers and designing extra strategies to determine the order of slot filling. (Due to the single slot nature of the dataset, we leave this as a direction for future investigation.)

• Asking clarifying questions is an essential step for Web search clarification. We intend to further explore how to generate the answer options that are paired with the clarifying questions.
Conclusion

• We explore an interesting but under-explored task, which aims to automatically ask clarifying questions with the intention of clarifying the user’s ambiguous information needs in web search scenarios.

• We propose a simple yet effective model to solve potential challenges of this task, with the main idea of jointly learning to select the question template and fill in the question slot.

Our code is available at: [https://github.com/iwangjian/TG-ClariQ](https://github.com/iwangjian/TG-ClariQ)
References


Thank you!

Q & A