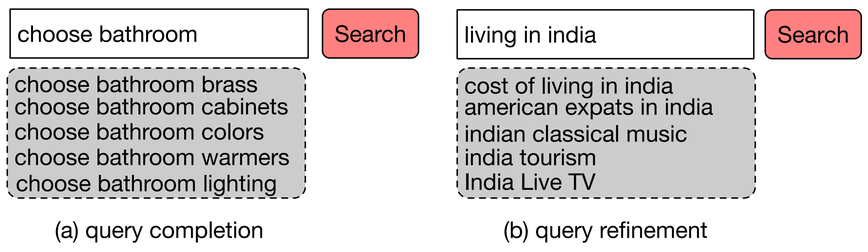
GENERATING HIGH-QUALITY QUERY SUGGESTION CANDIDATES FOR TASK-BASED SEARCH

Heng Ding[1,2], Shuo Zhang[2], Darío Garigliotti[2], and **Krisztian Balog[2]**
[1] Wuhan University, Wuhan, China [2] University of Stavanger, Stavanger, Norway

MOTIVATION

- Given an initial query, we want to get **a ranked list of query suggestions** that cover all the possible subtasks related to the task the user is trying to achieve.
- Challenge 1: Can a unified method produce both query completions (QC) and query refinements (QR)?
- Challenge 2: Can suggestions be obtained without relying on candidates from a major web search engine, or even a query log?



METHODS FOR QUERY SUGGESTION GENERATION

- In our two-stage approach (*suggestion generation* and *suggestion ranking),* we focus on the first component.
- $P(q|q_0)$ denotes the probability of a suggestion candidate q given a task-related initial query q_0

1. Popular Suffix Model
uses frequent suffixes mined from a query log

$$P(q|q_0) = popularity(s)$$

2. Neural Language Model
extends the input query character by character

$$P(q|q_0) = \prod_{j=n}^{m-1} P(c_{j+1}|c_1, \ldots, c_j)$$

3. Sequence-to-sequence Model
translates a source query into a target suggestion

$$P(q|q_0) = \prod_{j=1}^{m-1} P(w'_{j+1}|w'_1, \ldots, w'_j, q_0)$$

INFORMATION SOURCES

- **AOL query log**. We pair queries in the same session
- **KnowHow**: a knowledge base of *(task, predicate, subtask)* triples. We collect all *task-subtask* pairs
- **WikiAnswers**: a collection of questions scraped from WikiAnswers.com. We get task-related queries by removing "how do you" and "how to" prefixes

TEST COLLECTION

- We consider all 100 queries from the TREC 2015 and 2016 Tasks tracks
- We produce suggestion candidates combining the methods with the information sources
- We annotate 12K+ QC and 9K+ QR suggestions with relevance assessments via crowdsourcing

RESULTS

Method	QC				QR	
	P@10	P@20	R	CR	P@10	P@20
AOL-PopSuffix	0.257	0.245	0.168	0.168	-	-
KnowHow-PopSuffix	0.195	0.170	0.102	0.256	-	-
WikiAnswers-PopSuffix	0.181	0.167	0.101	0.333	-	-
AOL-NLM	0.256	0.241	0.170	0.474	-	-
KnowHow-NLM	0.166	0.147	0.108	0.575	-	-
WikiAnswers-NLM	0.163	0.121	0.088	0.650	-	-
AOL-Seq2Seq	0.283	0.181	0.156	0.765	0.043	0.031
KnowHow-Seq2Seq	0.158	0.111	0.079	0.813	0.206	0.148
Keyphrase-based [1]	0.321	0.239	0.130	-	0.575	0.504
Google API	0.267	0.134	0.078	-	0.289	0.145

Table. Precision for candidate suggestions generated by different *source-model* configurations. For QC methods, we also report on recall (R) and cumulative recall (CR).

- **RQ1:** *Can existing query suggestion methods generate high-quality query suggestions for task-based search?*
  - Yes, and we are able to produce more suggestions than what the (public) Google API provides
  - Keyphrased-based method [1] is still the best
- **RQ2:** *What are useful information sources per method?*
  - AOL query log is the best source for QC suggestions
  - For QR, KnowHow works well; AOL lacks QR-based pairs
  - CR improvements show they generate **unique candidates**

For example, given q_0 = *choose bathroom*, we get *choose bathroom marks* (WikiAnswers-NLM), *choose bathroom supply* (AOL-NLM), *choose bathroom for your children* (KnowHow-NLM), *choose bathroom appliances* (KnowHow-Seq2Seq), all beyond Google API and keyphrase-based systems

References:
[1] D. Garigliotti and K. Balog. Generating Query Suggestions to Support Task-Based Search. In Proc. of SIGIR '17.

University of Stavanger
Information Access & Interaction

Get the paper
http://bit.ly/2BnSjhR