Conversational Search

Adapted from: TREC CAST, original available here

Motivation

Conversational Information Seeking (CIS) is highlighted as an important emerging research area in the SWIRL 2018 workshop report on future trends in Information Retrieval. CIS is timely and important with increased adoption of a new generation of conversational 'assistant' systems, including Amazon Alexa, Cortana, Bixby, Google Assistant, and many others. Voice-based assistant interactions are now common, with a recent Comscore report showing that over 20% of homes in America own a smart speaker and over 500M devices with the Google Assistant worldwide. However, despite current assistants' ability to perform simple well-defined actions, their ability to support conversational information seeking is still very limited.

The goal of this TREC track is to pursue CIS research and create a large-scale reusable test collections for open-domain conversational search systems. The primary initial focus will be on system understanding of information needs in a conversational format and finding relevant responses using contextual information. In particular, we are motivated by long-running and complex tasks requiring multiple turns (possibly multiple sessions). The figure to the right shows a typical pipeline for a conversational agent system. The vision is to address all of these aspects. We first focus on retrieval of relevant result content in context.

We define several key properties of topics for CAsT:

- Complexity - Requires multiple turns to address the different aspects
- Diversity - Cut across all domains of information topics (news, travel, health, politics, history, science, etc…)
- Answerable - Most turns should be answerable with relevant content
- ~~Multi-source - Requires content from multiple information sources (not a single article)~~
- Varied discourse - Varying types of conversational structural patterns

Example

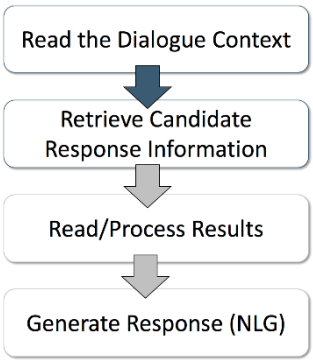
Task Track topic 22: flowering plants for cold climates

[You would like to buy and take care of flowering plants for cold climates]

TopicID.TurnID	User	System answer candidates
1.1	What flowering plants work for cold climates?	Answer passage 1, 2 and 3
1.2	How much cold can pansies tolerate?	Answer passage 1, 2 and 3
1.3	Does it have different varieties?	Answer passage 1, 2 and 3
1.4	Can it survive frost?	Answer passage 1, 2 and 3
1.5	How do I protect my plants from cold weather?	Answer passage 1, 2 and 3
1.6	How do plants adapt to cold temperature?	Answer passage 1, 2 and 3
1.7	What is the UK hardiness rating for plants?	Answer passage 1, 2 and 3
1.8	How does it compare to the US rating?	Answer passage 1, 2 and 3
1.9	What's the rating for pansies?	Answer passage 1, 2 and 3
1.10	What about petunias?	Answer passage 1, 2 and 3

Task

CAsT defines conversational search as a retrieval task in the conversational context. The year 1 task is candidate response ranking in context. The goal of the task is still to satisfy a user's information need, which is expressed through a sequence of conversational turns. The response from the retrieval system is a ranking of short text responses suitable for voice-interface or a mobile screen (e.g. roughly 1-3 sentences in length). As pictured above, for year 1 the conversational topics and turns will be fixed trajectories pre-defined in advance.



Primary task

Retrieval-based "candidate response ranking" in context (INFORM dialogue acts)

- Read the current dialogue turns up to the given turn (context)
  - For Year 1 the context provided is a fixed set of previous series of raw utterances in the preceding turns up to current step.
- Retrieve candidate response (text passages) from a fixed text collection for the current turn (every turn in the predefined set of topics).

Dataset

Text Collection

The goal is to retrieve passages from target open-domain text collections. We use a combination of three different text collections (that mirrors major verticals for conversational agents). Passages must be retrieved from one of the three following collections:

~~**English Wikipedia**~~

- ~~TREC Complex Answer Retrieval v2.1 (may increase to 3.0 pending availability)~~
- ~~Article paragraph content from Wikipedia (Wikipedia dump from December 20, 2016)~~
- ~~Approximately 5 Million articles~~
- ~~This data is publicly available~~

MS MARCO web passage data

- MS MARCO Passage Reranking data
- 10 million answer candidates from Bing search
- This data is publicly available

~~**Washington Post news collection**~~

- ~~TREC Washington Post Corpus used by the TREC News Track~~
- ~~608,180 news articles and blog posts from 2012 through 2017~~
- ~~NOTE: This collection requires a data license agreement.~~

Training topics

We provide sample training topics from two sources. In the project you will use the *manually* constructed dialogues (the second is derived from *MARCO web search session* data).

Manual dialogues

- Manually created dialogues by the track coordinators
- Limited relevance assessments from a baseline algorithm will be provided.

We provide the conversation topics in a standard protocol buffer (and JSON) format.

Example:

```
  {
    "title": "Goat breeds"
    "number": 2,
    "description": "Interested in buying goats that implies interest in
different breeds of goats and their use (milk, meat, and fur).",
    "turn": [
      {
        "number": 1,
        "raw_utterance": "What are the main breeds of goat?"
      },
      {
        "number": 2,
        "raw_utterance": "Tell me about boer goats."
      },
      {
        "number": 3,
        "raw_utterance": "What breed is good for meat?"
      },
     ...
      {
        "number": 11,
        "raw_utterance": "Are they profitable?"
      }
    ],
  }
```

Each *topic* has a sequence of *turns*. Information should be returned for each turn.

Evaluation

Results

Sample results are provided on the website for reference. This is the standard TREC run format for convenience:

```
TOPICID_TURNID Q0 CAR_9918559420932915201 1 34.4 samplerun
TOPICID_TURNID Q0 MARCO_12491099185594209 2 31.2 samplerun
TOPICID_TURNID   Q0   WAPO_b2e89334-33f9-11e1-825f-dabc29fd7071-1   3   30.2
samplerun
```

- The first field is the **turn identifier**, consisting of the **topic id** and **turn id concatenated** with an **underscore**, e.g. 31_1 for topic 31, turn 1)
- The second is a **literal** Q0 that is a placeholder
- The **passage identifier** (collection+passage id) separated by an underscore, e.g. (CAR_991855942093291520)
- The **rank** per turn in the conversation
- The **score** of the passage (in descending order)
- The **run identifier**, this should be descriptive and unique to your team and institution

Assessment

The relevance standard for a [turn, passage] pair is intended to represent how a person would feel if she asked the question to her favorite conversational assistant (Siri, Cortana, Alexa, Google Assistant, etc…) and it responded with the text in the passage. A five-point relevance scale is used, as follows:

- **4 Fully Meets -** The passage is the 'perfect' single response to the utterance. The passage is a perfect answer for the turn. It includes all of the information needed to fully answer the turn in the conversation context. It focuses only on the subject and contains little extra information.
- **3 Highly meets** - The passage answers the utterance and is focused on the answer (i.e., what a voice assistant should deliver). The passage answers the question and is focused on the turn. It would be a satisfactory answer if Google Assistant or Alexa returned this passage in response to the query. It may contain limited extraneous information.
- **2 Moderately Meets** - The passage answers the utterance, but is focused on something related (i.e., it might initially be clear why a voice assistant picked this passage). The passage may contain the answer, but users will need extra effort to pick the correct portion. The passage may be relevant, but it may only partially answer the turn, missing a small aspect of the context.
- **1 Slightly meets** - The passage includes some information about the turn, but does not directly answer it. Users will find some useful information in the passage that may lead to the correct answer, perhaps after additional rounds of conversation (better than nothing).
- **0 Fails to meet** - The passage is not relevant to the question. The passage is unrelated to the target query.

Metrics

The ranking depth is the same as for adhoc search, but we focus on the earlier positions (1, 3, 5) for the conversational scenario. Standard ranking metrics such as P@1, P@3 and MAP will be calculated using the judgments.

The turn depth evaluates the system performance at the n-th conversational turn in the topic. Performing well on deeper rounds (larger n) indicates better ability to understand context. We will define and experiment with variations on graded relevance measures that takes the overall conversation sequence into account.

RI Project Implementation

The goal of the Information Retrieval course is to implement a solution to solve the above task.

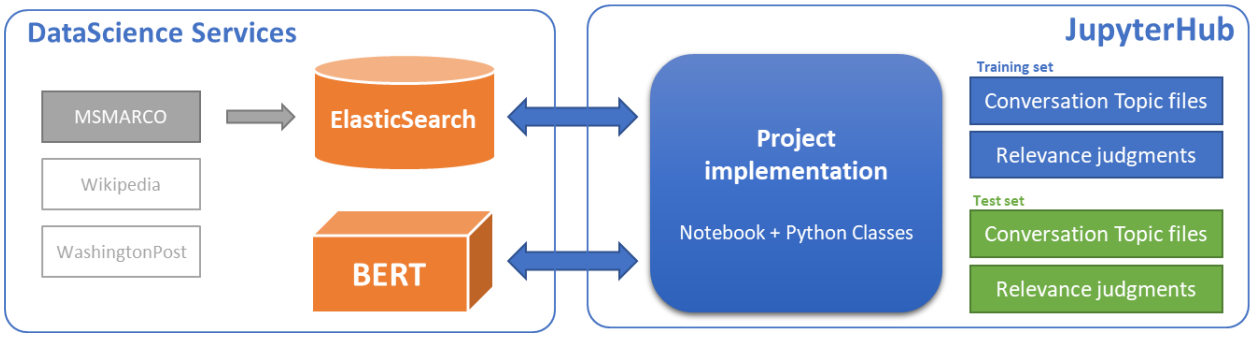
- Only the MSMARCO dataset will be used.
- You should not worry about indexing: MSMARCO will be pre-indexed and available through an Elastic Search service.
- Each group will have one username only to access the NOVASearch cluster. You must create **sessions with 2GB of memory, without GPU**.

Architecture

You will use the JupyterHub environment where critical data processing components are available through REST services:

- an ElasticSearch service will be available to **search the corpus**;
- a BERT service with Google's pre-trained model will be available to **create sentence embeddings**.

Your implementation will be divided in 4 incremental steps: (1) retrieval, (2) embeddings, (3) ranking and (4) conversation tracking.



Deadlines

Phase 1: 20 November (30%)

- **Grading**: 10% implementation + 20% report (use the notebook for the report)
- **Retrieval step:** Use the LMD model to select the top 100/1000 passages for each conversation turn.
- **Embedding**: Use sentence embedding methods to compute the similarity between passages and conversation turn.
- **Experimental metrics**: P@10, Recall and NDCG

Phase 2: 20 December (70%)

- **Grading**: 20% implementation + 50% report
- **Re-ranking step**: tune a learning to rank method with the embeddings.
- **Dialog answer calculation**: Use the conversation state to select the agent's best answer candidates (5 per turn).
- **Experimental metrics**: P@10, Recall and NDCG

Phase 1a: Retrieval step

- **Reading the Conversation Topics**

  Analyse and understand the methods that read the conversation topics.

- **Using Elastic Search in the NOVASearch Cluster**

  Analyse and understand the methods that search Elastic Search.

- **Evaluating the Conversation Topics Results**

  Analyse and understand the methods that evaluate conversation answer results.

  Implement a search-based conversation framework evaluation framework to evaluate conversation topics made up of conversation turns.

  Evaluate only until the 8th conversation turn (unfortunately the TREC CAST organizers ran out of budget).

Phase 1b: Embeddings

- **Compute the BERT embeddings of the corpus passages and conversation turns**

  Analyse and understand the implementation provided for computing sentence embeddings with BERT.

  Explore this embedding space using the sci-kit learn *k-nn* algorithm.

  Implement a embedding-based ranking conversation framework evaluation framework to evaluate conversation topics made up of conversation turns.

Phase 2a: Passage Re-Ranking

- **Learning to Rank**

  Analyse and understand the implementation provided for the ranking with coordinate ascent algorithm.

  Implement a ranking-based conversation framework evaluation framework to evaluate conversation topics made up of conversation turns.

- ~~**(optional) Learning to Rank with Gradient Boosted Trees**~~

  ~~Analyse and understand the implementation provided for the ranking with GBT algorithm.~~

Phase 2b: Conversation state-tracking

- **Conversation state-tracking**

  Implement a conversation state tracking method.

  You may use the conversations with the manual co-references resolution.

  Implement a conversation state tracking and evaluate the conversation topics.