# Glasgow Representation and Information Learning Lab (GRILL) at the Conversational Assistance Track 2020

**Carlos Gemmell**
University of Glasgow
carlos.gemmell@gla.ac.uk

**Jeffrey Dalton**
University of Glasgow
jeff.dalton@glasgow.ac.uk

## Abstract

In this paper we present our methods, experimental setup and results for the Conversational Assistance Track (CAsT) at TREC 2020. We present a novel neural query re-writing objective for conversational disambiguation that maximises semantic and grammatical knowledge transfer by aligning pre-training and fine tuning objectives. When resolving queries, our model regenerates previous context staying true to original infilling objective. Our re-writer assimilates query and context to auto-regressively resolve queries from previous utterances resulting performance approaching that of manual results. When used as part of a multi-stage retrieval pipeline leveraging point-wise and pair-wise scores, our system allows for robust conversational information seeking. We demonstrate our system by significantly outperforming median results in both manual and automatic runs from the track and show generalisation of our system with qualitative examples.

## 1 Introduction

Conversation is the primary way we communicate with each other. However, this is not the case with our technology. Search engines, for instance, are still constrained to a single query that needs to contain all relevant information to be answered. Conversational search is a challenge for current systems since the information required to answer the question or retrieve relevant information is spread among multiple turns of interaction. This year, TREC CAsT formalises this challenge into an actionable environment to test systems.

In this work we present our proposed method bridging the gap between conversational modelling and multi-stage information retrieval. We introduce a new model to re-write unresolved queries based on previous context using deep bidirectional Seq2Seq networks, and show its effectiveness when compared to similar systems from CAsT 2020. We especially show the significance of aligning the training objectives used during fine tuning to maximise knowledge transfer form the original weights.

## 2 Task Description

Similar to traditional document retrieval, CAsT is framed as an information seeking task where the objective is to return a relevant document $d*$ from a large corpus $D$ given a query $q$. However, the conversational nature implies that at certain turns not all required information to satisfy the user's query will be contained in $q$. We formalise the sequence of raw turns as:

$$Q_n = q_1, q_2, ..., q_n$$

Similar to human-human interaction, not all information is provided during a single utterance and is spread through the preceding turns of the conversation. This in turn leads to an information dependency for any $q_i$ to any number of preceding turns. We denote $q*$ as a fully resolved query containing all information necessary to obtain $d*$ without the need for additional context. We define an aggregation function $f$ such that all information contained in the sequence $Q_n$ results in $q*$. We note that while there is much information in $Q_n$, not all of which is relevant at turn $n$, the aggregation function $f$ specifically condenses relevant information for the last turn $q_n$.

$$q* = f(Q_n) \tag{1}$$

This spread of information across queries leads to a breakdown of traditional search systems reliant on a concise query $q*$.

In comparison to CAsT 2019, the 2020 version of the task increases the information dependency on the corpus $D$ by segmenting the information

needs to obtain $q*$ in the results of previous turns. Thus the previous results denoted as $d*_{1,\ldots,n-1}$ are for part of the inputs for $f$

$$q* = f(Q_n, d*_{1,\ldots,n-1}) \quad (2)$$

This increases the fidelity of the task w.r.t real world conversations, and increases the difficulty significantly since a theoretical marginalisation over $D$ is required, yet $D$ is often many orders of magnitude larger than $Q_n$.

Due to the low frequency of situations where $q*$ is dependant on $d*_{1,\ldots,n-1}$ we opt to model $f$ as we describe in Equation:1 by distilling all relevant information from $Q_n$ to approximate $q_*$.

## 3 Methods

Challenges are divided into two categories: manual runs and automatic runs. Manual runs evaluate a system's performance and retrieving $d*$ from $D$ when $q*$ is given, thus testing raw retrieval capacity. Automatic runs play out the full conversational scenario by providing a sequence of unresolved queries $Q_n$. To achieve the best performance for the task we optimise every section of the conversational retrieval pipeline. Our final system consists of multiple stages of retrieval and re-ranking to compliment our initial re-writing strategy.

### 3.1 Run Description

**Official runs**

- **grill_duoBART**: Automatic run with BART re-writer using full context (FC) decoding to generate the resolved query, then used in a BM25 first pass, monoBERT point-wise re-tanking, and duoBERT pairwise re-ranking without normalisation into the point-wise list.
- **grill_bmDuo**: Manual run leveraging the same setup as grill_duoBART without a the BART-FC re-writer.
- **grill_ctxDuo**: Manual run with an SDM first pass retrieval, then sent to mono & duoBERT without normalisation into the point-wise list.

**Unofficial runs**

- **org_manual_bertbase**: Manual baseline provided by the organisers. BM25 and BERT re-ranker similar to monoBERT.
- **org_auto_bertbase**: Baseline provided by the organisers using a GPT-2 (Radford et al.) decoder to generate only the resolved version of a query. BM25 and BERT re-ranker similar to monoBERT.
- **GPT2_duo_norm**: GPT-2 re-writer with a BM25, mono & duoBERT ranking pipeline. Pair-wise scores are normalised in the original list.
- **grill_BART_LT_bm_duo_norm**: An automatic run with a BART-LT re-writer generating only the resolved query, BM25, mono & duoBERT ranking pipeline.
- **grill_BART_FC_bm_duo_norm**: An automatic run with a BART-FC re-writer, BM25, mono & duoBERT ranking pipeline.
- **grill_BART_FC_SDM_mono:** An automatic run with a BART-FC re-writer, SDM, monoBERT ranking pipeline.
- **grill_SDM_Duo_norm**: A manual run with (BM25/SDM), mono & duoBERT ranking pipeline.
- **grill_fuseDuo**: An automatic run using BART-FC, mono & duo BERT. A bug was found in it as the source of the pair-wise normalisation and has been corrected.

### 3.2 Query Re-writing

In the following section we outline our proposed method $f'$ to aggregate information from $Q_n$ to approximate $q*$.

Since we hope to obtain a new query $q'$ from $Q_n$, an ordered set of unresolved queries, we opt for a Seq2Seq model as a re-writer. We use the Transformer (Vaswani et al., 2017) architecture to benefit from increased parallelism during training. CAsT is a challenging environment for these types of architectures given the limited dataset size. Pre-trained architectures (Devlin et al., 2018) have shown to remedy this to an extent by leveraging general text features, as such we initialise our re-writer with pre-trained weights from BART (Lewis et al., 2019) trained on text from Wikipedia and fine tuned on a summarisation task for CNN & Daily Mail. While weights from Wiki BART could already encode meaningful features, the training is framed as a span infilling task. In order to increase the information extraction and synthesis capabilities of our model we use the summarisation weights. We keep the same vocabulary to ensure feature transfer and use conditional

sequence modelling objective with cross entropy loss as gradient signal on 500 context and query samples from CAsT 2019 & 2020.

We keep the reconstruction objective by tasking the model to reproduce the original sequence of context queries as well as the re-written final query. We use the name BART-FC (BART Full Context) to refer to this form of inference.

To evaluate the effectiveness of BART-FC against a more traditional Seq2Seq formulation we create BART-LT (BART Last Turn) where the only generated output is the target resolved query.

### 3.3 First Pass Retrieval

Given either a manual query $q*$ or a re-written query $q'$, the objective of the first pass of our multi-stage retrieval pipeline is to efficiently obtain a pool of potentially relevant documents, as such, the initial pass is crucial to perform well in later stages. We set a cutoff at 1000 documents meaning order among these is irrelevant, while all other documents will be ignored for the ranking.

Since our corpus $D$ is large we investigate the use of a tuned BM25 model, and a Sequential Dependency Model (SDM) (Metzler and Croft, 2005) for efficient initial document recall.

### 3.4 Second Pass Re-ranking

Given a pool of retrieved documents with respect to $q$ we use multiple re-rankers to re-order documents according to relevance. In the following section we outline our approach.

#### 3.4.1 Mono BERT

In order to capture semantic relationships between a query and relevant documents we perform a point-wise re-ranking using monoBERT (Nogueira and Cho, 2020). MonoBERT performs a bi-encoding of the query and document and uses a custom classification head on the [CLS] token to score $s$ the relevance of the document to the query. Given the similarity of the training and target domain on MS Marco we do not perform any further fine-tuning to the model for CAsT.

$$s = monoBERT(q, d) \tag{3}$$

The new point-wise scores are then used to re-order the documents.

#### 3.4.2 Duo BERT

DuoBERT from Nogeira (Nogueira et al., 2019) is a pair-wise comparison model that evaluates the relevance of two documents $d_a$ & $d_b$ against a query $q$. All three sequences form part of the input to BERT with a specific head on the [CLS] token indicating a weighting to either document. This process allows a grounded comparison rather than a floating score given by monoBERT.

Since duoBERT is a pairwise model, scores are only relative and thus scores for a document are obtained by summing it's given score when compared to all other documents.This is an expensive process with $O(l^2)$ time complexity with respect to the pool length $l$ and as such is only suited as the last stage of a re-ranking pipeline. In our experiments we find a good trade-of between pool length and time by taking the top 10 documents from the monoBERT scores.

These re-ranking stages ensure that the highest quality documents are pushed to the top of the ranking.

Since scores given by duoBERT are only relative umong documents, the overall computed document relevance score is not comparable to that of monoBERT directly. Reintegrating the un-normalised pair-wise ranked documents into the original list can have a negative effect since there is no guarantee the relative scores will be higher than the point-wise scores. Normalising is achieved by taking the maximum and minimum of the original top scores and linearly interpolating the pair-wise scores between them. We discuss the effects of normalisation in the results section.

## 4 Results

We observe significant improvements for our developed multi-stage pipeline over the median scores and baseline systems in all categories. Statistical significance is computed by a standard pared t-test and stated within a 95% confidence interval by '*'against the organisers baselines.

Table 1 indicates the need for re-writing from raw queries. Improvements in re-writing quality are highlighted with BART-LT significantly outperforming GPT-2 given the same training objective of generating solely the resolved query. Both models during fine tuning diverge from the original pre-training objective, however, BART-LT benefits from a bi-direrctional encoder allowing for more fine grained information flow.

However, remaining aligned with the original pre-training objective during fine tuning and inference yields best results as BART-FC shows both

| Model | NDCG@3 | NDCG@5 | NDCG | MAP | R@500 | R |
|---|---|---|---|---|---|---|
| best | 0.733 | 0.687 | 0.710 | 0.492 | — | — |
| median | 0.280 | 0.274 | 0.375 | 0.180 | — | — |
| Raw + BM25 + BERT | 0.170 | 0.159 | 0.144 | 0.078 | 0.160 | 0.160 |
| org_auto_bertbase | 0.300 | 0.287 | 0.284 | 0.159 | 0.333 | 0.333 |
| GPT2_duo_norm | 0.313 | 0.300* | 0.289 | 0.164 | 0.333 | 0.333 |
| grill_BART_LT_bm_duo_norm | 0.336* | 0.324* | 0.389* | 0.219* | 0.504* | 0.516* |
| grill_duoBART | **0.398*** | **0.380*** | **0.403*** | **0.215*** | **0.476*** | **0.526*** |
| grill_BART_FC_bm_duo_norm | 0.398* | 0.379* | 0.417* | 0.239* | 0.515* | 0.526* |
| grill_BART_FC_SDM_mono | 0.386* | 0.368* | 0.479* | 0.262* | 0.642* | 0.688* |
| grill_fuseDuo | **0.416*** | **0.395*** | **0.486*** | **0.265*** | **0.642*** | **0.688*** |

Table 1: Performance comparison for CAsT 2020 automatic runs. Metrics without depth indication are taken at 1k. R indicates recall.

| Model | NDCG@3 | NDCG@5 | NDCG | MAP | R@500 | R |
|---|---|---|---|---|---|---|
| best | 0.724 | 0.683 | 0.698 | 0.478 | — | — |
| median | 0.414 | 0.398 | 0.489 | 0.263 | — | — |
| BM25 | 0.284 | 0.279 | 0.461 | 0.214 | 0.685 | 0.767 |
| org_manual_bertbase | 0.479 | 0.461 | 0.451 | 0.272 | 0.508 | 0.508 |
| BM25 + mono BERT | 0.503* | 0.498* | 0.591* | 0.374* | 0.755* | 0.767* |
| grill_bmDuo | **0.530*** | **0.511*** | 0.571* | 0.324* | 0.689* | 0.767* |
| grill_bm_Duo_norm | 0.531* | **0.519*** | 0.598* | 0.378* | 0.755* | 0.767* |
| grill_ctxDuo | 0.507* | 0.500* | **0.607*** | **0.383*** | **0.782*** | **0.799*** |
| grill_SDM_Duo_norm | **0.531*** | 0.518* | **0.614*** | **0.389*** | **0.782*** | **0.799*** |

Table 2: Performance comparison for CAsT 2020 manual runs. Metrics without depth indication are at 1000.

qualitative and empirical improvements when using the same re-ranking pipeline.

MonoBERT and duoBERT makeup our last stages of our conversational document retrieval pipeline. As we see in Table 2 MonoBERT provides a point-wise score and results in the most performance gain given an original ranking from either BM25 or an SDM. However, duoBERT is very effective at the top of the list with improvements of 3% absolute in NDCG@3 over the monoBERT runs. Given the quadratic cost of increasing the list for duoBERT, we investigate the optimal list size for ranking time and effectiveness and find 10 through validation on CAsT 2019 data to strike a good balance. Increasing further to 30 documents yields insignificant improvements.

Using un-normalised pair-wise scores in combination with an original ranking has the effect, in our experiments, to send the lowest ranking document from the pair-wise process to the bottom of the overall ranking since scores are often negative due to the relative nature of the comparison. This has a sharp effect on the top results and can be seen by the reduction in recall@500 while recall@1k remains constant.

## 5 Conclusion

In this work we present our submission to TREC CAsT 2020. We propose a multi-stage approach for effective conversational document retrieval leveraging fine tuned language models for both contextual query re-writing, point-wise and pair-wise re-ranking. We introduce a novel fine tuning objective in BART-FC for query re-writing that maximises knowledge transfer from a pre-trained language model by aligning the fine tuning task to the original reconstruction objective of BART. This ensures minimal loss in semantic knowledge and garmatical structure when applied to new tasks. The combination of our contextual re-writer with effective re-rankers significantly outperform median results for the track and strong baseline systems.

## References


Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805 [cs]*. ArXiv: 1810.04805.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. *arXiv:1910.13461 [cs, stat]*. ArXiv: 1910.13461.

Donald Metzler and W. Bruce Croft. 2005. A Markov random field model for term dependencies. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval - SIGIR '05*, page 472, Salvador, Brazil. ACM Press.

Rodrigo Nogueira and Kyunghyun Cho. 2020. Passage Re-ranking with BERT. *arXiv:1901.04085 [cs]*. ArXiv: 1901.04085.

Rodrigo Nogueira, Wei Yang, Kyunghyun Cho, and Jimmy Lin. 2019. Multi-Stage Document Ranking with BERT. *arXiv:1910.14424 [cs]*. ArXiv: 1910.14424.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language Models are Unsupervised Multitask Learners. page 24.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. *arXiv:1706.03762 [cs]*. ArXiv: 1706.03762.