Sentence Retrieval for Open-Ended Dialogue using Dual Contextual Modeling

Itay Harel[1⋆], Hagai Taitelbaum[2], Idan Szpektor[2], and Oren Kurland[3]

[1] TSG IT Advanced Systems Ltd., Tel Aviv, Israel
itay.harel91@gmail.com
[2] Google Research, Tel Aviv, Israel
{hagait,szpektor}@google.com
[3] Technion — Israel institute of technology, Haifa, Israel
kurland@technion.ac.il

Abstract. We address the task of retrieving sentences for an open domain dialogue that contain information useful for generating the next turn. We propose several novel neural retrieval architectures based on dual contextual modeling: the dialogue context and the context of the sentence in its ambient document. The architectures utilize contextualized language models (BERT), fine-tuned on a large-scale dataset constructed from Reddit. We evaluate the models using a recently published dataset. The performance of our most effective model is substantially superior to that of strong baselines.

Keywords: open domain dialogue · dialogue retrieval · sentence retrieval

1 Introduction

Throughout the last few years there has been a rapid increase in various tasks related to dialogue (conversational) systems [14, 12, 37, 7, 15, 47]. Our work focuses on responses in an open-dialogue setup: two parties converse in turns on any number of topics with no restrictions to the topic shifts and type of discussion on each topic. In addition, the dialogue is not grounded to a specific document, in contrast to the setting used in some previous work (e.g., [28]). The task we pursue is to retrieve passages — specifically, sentences — from some document corpus that would be useful for generating the next response in a given dialogue; the response can be written either by humans or by conditional generative language models [12, 17, 34].

There has been much research effort on utilizing information induced from the context of the last turn in the dialogue — henceforth referred to as *dialogue context* — so as to retrieve a response from a corpus of available responses [45, 4, 35, 40, 38, 46, 48, 20]. However, these models address complete responses as the retrieved items. In our setting, the retrieved items are sentences from documents,

⋆ Work done while at the Technion.

which may aid in writing a complete response. Unlike full responses, sentences usually do not contain all the information needed to effectively estimate their relevance to an information need. Indeed, there is a line of work on ad hoc sentence retrieval [30, 13] and question answering [41, 18, 24] that demonstrated the clear merits of using information induced from the document containing the sentence, henceforth referred to as *sentence context*.

To address sentence retrieval in a dialogue setting, we present a suite of novel approaches that employ dual contextual modeling: they utilize information not only from the dialogue context but also from the context of candidate sentences to be retrieved; specifically, from the documents that contain them. We are not aware of previous work on conversational search that utilizes the context of retrieved sentences. Using the context of the dialogue is important for modeling latent information needs that were explicitly or implicitly mentioned only in previous dialogue turns. Using the context of the sentence in its ambient document is important for inducing an enriched representation of the sentence which can help, for example, to fill in topical and referential information missing from the sentence.

Our sentence retrieval approaches employ the BERT [10] language model, fine-tuned for simultaneous modeling of the context of the last turn in the dialogue and the context of a candidate sentence for retrieval. We propose three different BERT-based architectures that differ in the way context in the dialogue and in the document are modeled and interact with each other. While our main architectural novelty lies in the study of the dialogue/sentence context interaction, some of the dialogue context modeling techniques we employ are also novel to this task.

We evaluated our models using a recently published dataset for sentence retrieval for open-domain dialogues [19]. The dataset was created from Reddit. It includes human generated relevance labels for sentences with respect to dialogues. As in [19], we used weakly supervised (pseudo) relevance labels for training our models.

We contrast the performance of our models with that of a wide suite of strong reference comparisons. The retrieval performance of our best performing approach is substantially better than that of the baselines. We also show that while using only the dialogue context results in performance superior to that of using only the sentence context, using them both is of clear merit. In addition, we study the performance effect of the type of document context used for sentences and the length of the context used from the dialogue.

To summarize, we address a research challenge that has not attracted much research attention thus far: retrieving sentences that contain information useful for generating the next turn in an open-domain dialogue. This is in contrast to retrieving responses and/or generating them, and to conversational retrieval where the information need is explicitly expressed via queries or questions. On the model side, our work is the first, to the best of our knowledge, to model both the dialogue context and the context of candidate sentences to be retrieved; specifically, using neural architectures that utilize a pre-trained language model.

2 Related Work

The two main lines of work on open domain dialogue-based systems [21] are response generation [12, 17, 49, 34] and response selection [45, 4, 35, 40, 38, 46, 48, 20]; some of these methods are hybrid selection/generation approaches. Some recent generative dialogue systems include a retrieval component that generates a standalone query from the dialogue context against a fixed search engine (e.g. Bing or Google) [23, 42, 16]. Response selection is a retrieval task of ranking candidate full responses from a given pool. In contrast, our task is to retrieve from a document corpus sentences that could serve as a basis for generating a response, optimizing the retrieval model.

Related to our task is conversational search [36, 7, 15, 47]. The goal of this task is to retrieve answers to questions posted in the conversation or to retrieve passages/documents that pertain to the information needs expressed in it. In our setting, we do not make any assumptions on the type of response to be generated from retrieved sentences. It could be an answer, an argument, or even a question. Consequently, the types of information needs our retrieval models have to satisfy are not necessarily explicitly mentioned in the dialogues (e.g., in the form of a question); they could be quite evolved, and should be inferred from the dialogue.

The last turn in a dialogue is often used as the basis for response selection or passage/answer retrieval. A large body of work utilized information induced from the dialogue *context* – the turns preceding the last one – to improve ranking. There are approaches that reformulate the last turn [26], expand it using terms selected from the context [27, 43], expand it using entire previous turns [29, 37, 34], or use the context for cross referencing the last turn [40]. Yan et al. [45] proposed to create multiple queries from the context, assign a retrieval score to a candidate response w.r.t. each query, and fuse the retrieved scores. We demonstrate the merits of our joint representation approach w.r.t. a representative turn expansion method [43] and to Yan et al.'s [45] fusion approach.

Other models for dialogue-based retrieval include the dialogue context as part of a retrieval neural network. Several works [38, 46, 48] use a hierarchical architecture for propagating information from previous turns to the last turn representation. Qu et al. [35] embed selected previous answers in the conversation as an additional layer in BERT. In contrast, we focus on early cross-attention of all context in the input, simultaneously modeling the dialogue context and the context of the sentences to be retrieved.

Passage context from the ambient document was used for non-conversational passage retrieval [30, 13] and question answering [41, 18, 24], but there was no dialogue context to utilize in contrast to our work. As already mentioned, there is much work on utilizing dialogue context for dialogue-based retrieval [29, 37, 34, 40, 27, 26, 43], and we use some of these methods as reference comparisons, but the passage (response) context was not utilized in this line of work.

3 Retrieval Framework for Open Dialogues

Suppose that two parties are holding a dialogue g which is an open-ended conversation. Open domain means that there are no restrictions about the topics discussed in g and their shifts.

The parties converse in turns: $g \stackrel{def}{=} < t_1, \ldots, t_n >$; t_1 is the first turn and t_n is the current (last) turn. Herein, we refer to a turn and the text it contains interchangeably. Our goal is to retrieve *relevant* sentences from some document corpus $\mathcal{C}$, where *relevance* means that the sentence contains information that can be used to generate the next turn, t_{n+1}. We address this sentence retrieval task via a two-stage ranking approach: an initial ranker (see Section 4 for details) followed by a more computationally-intensive ranker that reranks the top-k retrieved sentences and is our focus.

In open-ended dialogues there is often no explicit expression of an information need; e.g., a query or a question. We assume that the current turn, t_n, expresses to some extent the information need since t_{n+1} is a response to t_n. Because turns can be short, preceding turns in g are often used as the context of t_n in prior work on conversational search and dialogue-based retrieval [45, 46, 48, 20, 35, 40]. Accordingly, we define the sequence $CX(t_n) \stackrel{def}{=} t_{n-h}, \ldots, t_{n-1}$ to be the dialogue (historical) context, where h is a free parameter. We treat the sequence $t_{n-h}, \ldots, t_n$ as a *pseudo query*, Q, which is used to express the presumed information need[4].

Standard ad hoc sentence retrieval based on query-sentence similarities is prone to vocabulary mismatch since both the queries and sentences are relatively short. We use the BERT [10] language model to compare the pseudo query, Q, with a sentence, which should ameliorate the effects of token-level mismatch. We note that the pseudo query Q is not as short as queries in ad hoc retrieval. Nevertheless, we hypothesize that utilizing information from the document containing the sentence, which was found useful in ad hoc sentence retrieval [30, 13] and question answering [41, 18, 24], will also be of merit in sentence retrieval for open dialogue.

Specifically, we define the context of sentence s in its ambient document as a sequence of m sentences selected from the document: $CX(s) \stackrel{def}{=} s_1, \ldots, s_m$; m is a free parameter. These sentences are ordered by their relative ordering in the document, but they need not be adjacent[5] to each other in the document. We treat the ordered sequence composed of s and its context as the *pseudo sentence*: $S \stackrel{def}{=} s_1, \ldots, s, \ldots, s_m$ (s may appear before s_1 or after s_m).

In what follows, we describe estimates for the relevance of sentence s with respect to dialogue g where relevance means, as noted above, inclusion of information useful for generating the next turn in the dialogue. The estimates are based on modeling the relation between the pseudo sentence S and the pseudo query Q.

[4] If $n - 1 < h$, we set $Q \stackrel{def}{=} t_1, \ldots, t_n$. If h=0, $Q \stackrel{def}{=} t_n$, $CX(t_n) \stackrel{def}{=} \{\}$.

[5] We evaluate a few approaches for selecting the sentences in $CX(s)$ in Section 5.

3.1 Sentence Retrieval Methods

We present three architectures for sentence retrieval in dialogue context that are based on matching between Q and S. These architectures utilize BERT for embedding texts, taking its pooler output vector as the single output, denoted by v^{BERT}.

From Dense Vectors to a Sentence Relevance Score The architectures we present below utilize a generic neural component that induces a sentence relevance score from a set of k dense vectors. The vectors, denoted $v_1, \ldots, v_k$, are the outcome of representing parts of Q and S and/or their matches using the architectures.

Following work on estimating document relevance for ad hoc document retrieval using passage-level representations [25], we train an encoder-only Transformer [10] whose output is fed into a softmax function that induces a relevance score for sentence s with respect to pseudo query Q:

$$Score(s|v_1, \ldots, v_k) = Softmax(W_{score}\ v_{out} + b_{score}), \tag{1}$$

$$v_{out} = TRANSFORMER_{enc}(v_0, \ldots, v_k)[0], \tag{2}$$

where: a) W_{score} and b_{score} are learned parameters; b) v_0 is the word embedding of the [CLS] token in BERT, which is prepended to the vector sequence $v_1, \ldots, v_k$; and c) v_{out} is the contextual embedding of the [CLS] token v_0 in the topmost Transformer layer[6]. Figures 1, 2 and 3 depict three models with this high-level architecture.

Architectures
We next present three BERT-based architectures. The output of each architecture is a sequence of dense vectors which is the input to $Score(s|v_1, \ldots, v_k)$ in Eq. 1 to compute the final sentence score.

The Tower architecture (Fig. 1). Two instances of BERT with shared weights produce two output vectors to compute $Score(s|v_1^{BERT}, v_2^{BERT})$ in Eq. 1. The input to the first BERT is the pseudo query Q with separating tokens between the turns: "[CLS] t_{n-h} [SEP] ... [SEP] t_n [SEP]"[7]. The second BERT is fed with the pseudo sentence S to be scored: "[CLS] s_1 [SEP] ... s [SEP] ... s_m [SEP]". This architecture is similar to Dense Passage Retrieval (DPR) [23].

The Hierarchical architecture (Fig. 2). A potential drawback of the Tower architecture (cf., [33]) is that matching the pseudo query and the pseudo sentence is performed after their dense representations were independently induced.

[6] We tried replacing the Transformer-based embedding in Eq. 2 with a feed-forward network with the same number of parameters, but this resulted in inferior performance.

[7] We also tested a simpler scoring approach (without fine tuning), $Cosine(v_1, v_2)$, which performed significantly worse.

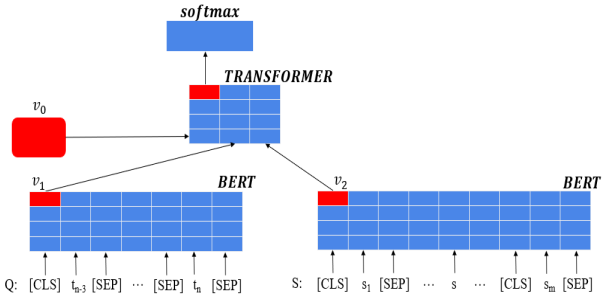

Fig. 1: The Tower model with dialog history $h = 3$.

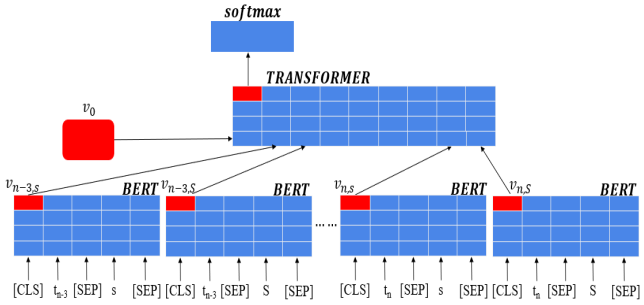

Fig. 2: The Hierarchical model with dialog history $h = 3$.

To address this, we present the Hierarchical architecture, which uses BERT to induce joint representations of each turn in the pseudo query with parts of the pseudo sentence.

This model uses $2*(h+1)$ instances of BERT with shared weights, constructed as follows. For each turn $i \in \{n-h, \ldots, n\}$ in Q, the model computes output $v_{i,s}^{BERT}$ by feeding BERT with turn i and the sentence s: "[CLS] t_i [SEP] s". Similarly, the output $v_{i,S}^{BERT}$ is computed by feeding BERT turn i and pseudo sentence S: "[CLS] t_i [SEP] s_1 [SEP] $\ldots s$ [SEP] $\ldots s_m$[SEP]". The output vectors $\{v_{n,s}^{BERT}, v_{n,S}^{BERT} \ldots v_{n-h,s}^{BERT}, v_{n-h,S}^{BERT}\}$ are fed as input to Eq. 1. This architecture is inspired by the PARADE ad hoc document retrieval model [25]. Unlike PARADE, we enrich all embeddings with positional encoding. The goal is to model sequential information, e.g., the order of turns in the dialogue.

The QJoint architecture (Fig. 3). The Hierarchical architecture enables early joint representations for each turn in the pseudo query Q and the pseudo sentence S. Still, turns are used independently to derive intermediate representations. We next present the QJoint architecture that represents jointly all turns in Q. The goal is to cross-attend the inter-relations between the turns in Q and their relations with S as early as possible.

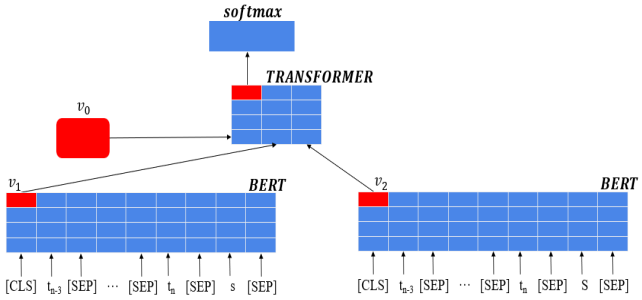

Fig. 3: The QJoint model with dialog history $h = 3$.

We use two instances of BERT with shared weights. The first jointly represents Q and s, with input "[CLS] t_{n-h} [SEP] ... t_n [SEP] s [SEP]". The second instance jointly represents Q and S, with input "[CLS] t_{n-h} [SEP] ... t_n [SEP] s_1 [SEP] ... s [SEP] ... s_m [SEP]". The two output BERT vectors serve as input to Eq. 2.

QJoint is conceptually reminiscent of early passage-based document retrieval methods where document and passage query similarity scores were interpolated [2]. Here, the pseudo sentence S is viewed as the document and the sentence s as the passage and their query matching induced using the BERT models is interpolated via a Transformer. The sentence context serves as a means to disambiguate, resolve references and offer global document topicality that may be missing from a single sentence. Yet, the BERT model that focuses on matching only the single sentence with the pseudo query offers a differentiator for ranking two consecutive sentences, which would share much of their pseudo sentence content.

Neural Reference Comparisons As noted in Section 1, previous work on conversational search and dialogue-based retrieval did not utilize the context of candidate passages in contrast to our architectures. We use several such BERT-based sentence retrieval methods as reference comparisons.

RANK$_{\text{BERT}}$. This method, which was the best performing in [19] for sentence retrieval for open-domain dialogues, takes BERT with input "[CLS] q [SEP] s [SEP]" and uses its output as input to Eq. 1 (which includes a Transformer layer). In this method, q is set to be turn t_n and no context for the sentence s is utilized; see QuReTeC and CONCAT next for different q settings.

QuReTeC. The Query Resolution by Term Classification method [43] (QuReTeC in short) is a representative of methods (e.g., [27]) that use explicit term-based expansion, based on the dialog history, to enrich the current (last) turn t_n. Specifically, it applies a token-level classifier, utilizing turns in $CX(t_n)$ (with a [SEP] token separating between turns), to select a few tokens that will be added to t_n. The resultant text is provided as input q to RANK$_{\text{BERT}}$.

CONCAT. As an alternative to the term selection approach, represented by QuReTeC, we consider the CONCAT method [27] which uses all $CX(t_n)$ when constructing the input q to $\text{RANK}_{\text{BERT}}$: "[CLS] t_{n-h} [SEP] ... [SEP] t_n [SEP] s [SEP]" . We note that CONCAT is essentially a special case of QJoint (Section 3.1) where no sentence context is used.

External Fusion. The Hierarchical architecture (Section 3.1) fuses information induced from the turns in the pseudo query by aggregating turn/sentence representations using a Transformer. In contrast, QJoint, and its special case CONCAT, perform the fusion via a joint representation of all the turns in Q and the sentence.

We also consider an external fusion approach which fuses information induced from the turns in Q at the retrieval-score level. We employ $\text{RANK}_{\text{BERT}}$ to assign a score to each sentence s in an initially retrieved sentence list with respect to each turn in Q. Hence, each turn t_i induces a ranked list of sentences L_i. Let $rank_{L_i}(s)$ be s's rank in L_i; the highest rank is 1. We use reciprocal rank fusion (RRF) [6] to fuse the lists $\{L_i\}$: $ExtFuse(s) \stackrel{def}{=} \sum_{L_i} \mu_i \frac{1}{\nu + rank_{L_i}(s)}$, where ν is a free parameter and μ_i is a uniform weight; linear and exponential decay weights did not yield improvements. Fusion was also used in [45] at the retrieval score level for conversational search, but contextualized language models were not used.

4 Experimental Setting

Dataset and Evaluation Measures. We use a recent dataset of sentence retrieval for open-ended dialogues [19][8]. Dialogues were extracted from Reddit, and sentences were retrieved from Wikipedia. The test set contains 846 diagloues, each accompanied with an initially retrieved list of 50 sentences judged by crowd workers for relevance. The initial ranker, henceforth *Initial Ranker*, is based on unigram language models and utilizes the dialogue context [19]. All sentence retrieval methods that we evaluate re-rank the initial sentence list provided in the dataset. We use Harel et al.'s [19] 50 random equal-sized splits of the test dialogues to validation and test sets; the validation set is used for hyperparamter tuning. We report the average and standard deviation over the 50 test sets of mean average precision (MAP), NDCG of the 5 highest ranked sentences (NDCG@5) and mean reciprocal rank (MRR). The two tailed permutation (randomization) test with $10,000$ random permutations and $p \leq 0.05$ was used to determine statistical significance. Bonferroni correction is applied for multiple hypothesize testing.

We followed the guidelines on the weakly-supervised training data collection from [19], which showed much merit. Specifically, the sentences in the initial list retrieved for a dialogue were assigned pseudo relevance labels using a fusion

[8] `https://github.com/SIGIR-2022/A-Dataset-for-Sentence-Retrieval-for-Open-Ended-Dialogues.git`.

approach. Following these guidelines, we obtained ∼73,000 dialogues with weakly annotated sentences, which were used to fine tune a basic BERT-based sentence retrieval model.

Other datasets are not a good fit for training and evaluating our models since (i) they do not include open-domain dialogues (e.g., TREC's CAsT datasets [7–9][9], CoQA [37], DoQA [3] and QuAC [4]) and/or (ii) they do not include the document context for training a dual context model.

Model and Training Settings. All neural models (Section 3.1) were fine-tuned end2end on the weakly supervised training set, unless stated otherwise. For a single text embedding in the Tower architecture, pre-trained BERT-Large [10] is used as a starting point. To embed a pair of texts, e.g., in $\text{RANK}_{\text{BERT}}$ and QJoint, as starting point we used a pre-trained BERT that was fine-tuned for ad hoc passage retrieval on the MS MARCO dataset [31]. We fine-tuned it using the $\text{RANK}_{\text{BERT}}$ architecture[10]; q and s were set to a query and a passage in MS MARCO, respectively, and trained with pointwise classification loss[11] [32].

We implemented and trained the QuReTeC model using the hyperparameter values detailed in [43] for 10 epochs. When generating the resolved queries, we applied the constraints mentioned below to the dialogue context; i.e., we set $h = 3$ with maximum of 70 tokens per turn. Then, $\text{RANK}_{\text{BERT}}$ was utilized for inference on the resolved queries. We tested all QuReTeC variants (different batch sizes, learning rates and number of epochs), each with the $\text{RANK}_{\text{BERT}}$ variant that was the best performing model in most of the validation splits.

Modeling the Pseudo Sentence Context. We tested three alternatives for modeling $CX(s)$, the context of sentence s in its ambient document: (i) *LocalSurround.* the sentence that precedes and the sentence that follows s, (ii) *LocalPrev.* the two sentences that precede s; and (iii) *Global.* the two sentences in the document whose TF-IDF vectors are most similar (via Cosine similarity) to the TF-IDF vector of s. Sentences in (i) and (ii) were limited to passage boundaries.

Unless stated otherwise, we used *LocalSurround* in our models, since it performed best in terms of MAP. We use only two sentences as context due to BERT's input-length limitation. If the input to BERT should still be truncated, first the sentences in the context are truncated, and only then the sentence s.

Bag-Of-Terms Reference Comparisons. In addition to the neural reference comparisons described in Section 3.1, and the unigram language-model-based Initial Ranker, we applied **Okapi BM25** [39] on the last turn t_n as a reference comparison.

Hyperparameter Tuning. The values of the following hyperparameters were optimized for MAP over the validation sets. Okapi BM25's $k_1 \in \{1.2, 2, 4, 8, 12\}$

[9] In addition, these datasets include a too small number of dialogues which does not allow for effective training of the proposed architectures, even when used for weak supervision.

[10] Without the additional transformer in Eq. 2.

[11] Training with pairwise loss showed no improvement.

Table 1: Architectures which utilize both the dialogue and the sentence context. Statistical significance w.r.t. Tower and Hierarchical is marked with 't' and 'h', respectively.

	MAP	NDCG@5	MRR
Tower	$.212^{\pm.007}$	$.298^{\pm.015}$	$.291^{\pm.014}$
Hierarchical	$.451^{\pm.010}_{t}$	$.611^{\pm.012}_{t}$	$.588^{\pm.013}_{t}$
QJoint	$\mathbf{.477}^{\pm.010}_{th}$	$\mathbf{.644}^{\pm.012}_{th}$	$\mathbf{.609}^{\pm.013}_{th}$

and $b \in \{0.25, 0.5, 0.75, 1\}$. RRF's (external fusion) $\nu \in \{0, 10, 60, 100\}$. All BERT-based models were trained using the Adam optimizer with learning rate $\in \{3e-6, 3e-8\}$ and batch size $\in \{8, 16\}$. $\text{RANK}_{\text{BERT}}$ that is the starting point of all these models was fine tuned as in [32].

All models were trained for 10 epochs on Google's TPU[12] v3-8 and the best model snapshot was selected based on the validation set. The number of Transformer layers (Section 3.1) was set to 2 in all related architectures following [25]. The maximum sequence length for all BERT-based models is 512. The dialogue context length h is set to 3. (We analyze the effect of h in Section 5.)

5 Results

Main Result. Table 1 compares our architectures, which utilize both the dialogue context and the sentence context. We see that Hierarchical outperforms Tower. This shows that jointly modeling matched texts, in our case the pseudo query and the sentence, is superior to modeling the interaction between texts only at the top-most neural layer. This finding is aligned with previous reports on semantic matching [11, 22]. We also see that QJoint is the best performing model. This attests to the downside of "breaking up" the pseudo query at the lower representation levels of the network while early cross-representation between the pseudo query and the pseudo sentence results in higher-quality semantic retrieval modeling. One potential benefit of Hierarchical is increased input capacity, since concatenating both the query and its context and the sentence and its context in QJoint may exceed the input size limit and incur penalty due to truncation.

Table 2 compares our most effective architecture, QJoint, with the neural and bag-of-terms baselines. The main difference between QJoint and the other models is that QJoint utilizes both the dialogue context and the sentence context, while the other methods utilize only the dialogue context, with the exception of BM25, as is the case in all prior work as noted in Section 1.

We see in Table 2 that all trained neural methods significantly outperform the Initial Ranker and Okapi BM25. The superiority of ExtFuse (external fusion) to

[12] `https://cloud.google.com/tpu/`

Table 2: The best performing QJoint compared to reference models. Statistical significance w.r.t. Initial Ranker and QJoint is marked with 'i' and 'q' respectively.

	MAP	NDCG@5	MRR
Okapi BM25	$.185^{\pm.006}_{iq}$	$.259^{\pm.010}_{iq}$	$.258^{\pm.009}_{iq}$
Initial Ranker	$.238^{\pm.007}$	$.355^{\pm.012}$	$.353^{\pm.012}$
QuReTeC	$.375^{\pm.009}_{iq}$	$.543^{\pm.014}_{iq}$	$.517^{\pm.013}_{iq}$
ExtFuse	$.436^{\pm.011}_{iq}$	$.606^{\pm.013}_{iq}$	$.582^{\pm.014}_{iq}$
CONCAT	$.470^{\pm.009}_{iq}$	$.635^{\pm.012}_{iq}$	$.607^{\pm.012}_{i}$
QJoint	$\mathbf{.477}^{\pm.010}$	$\mathbf{.644}^{\pm.012}$	$\mathbf{.609}^{\pm.013}$

Table 3: QJoint with no context, only with sentence context, only with dialogue context, and both. The corresponding statistical significance marks are 'n', 's' and 'h', respectively.

Context Used	MAP	NDCG@5	MRR
None	$.354^{\pm.009}$	$.481^{\pm.014}$	$.468^{\pm.013}$
Sentence context only	$.351^{\pm.008}$	$.478^{\pm.014}$	$.463^{\pm.013}$
Dialogue context only	$.470^{\pm.009}_{ns}$	$.635^{\pm.012}_{ns}$	$.607^{\pm.012}_{ns}$
Both	$\mathbf{.477}^{\pm.010}_{nsh}$	$\mathbf{.644}^{\pm.012}_{nsh}$	$\mathbf{.609}^{\pm.013}_{ns}$

QuReTeC can potentially be attributed to the fact that it compares all the turns in the dialogue context with the sentence rather than "fuses" several selected terms with the last turn to yield a single query compared with the sentence. It is also clear that CONCAT outperforms ExtFuse, which attests to the merit of using a joint representation for the entire pseudo query and the sentence compared to fusing retrieval scores attained from matching parts of the pseudo query with the sentence. We also point that CONCAT improves over Hierarchical (see Table 1), which does utilize the sentence context. This indicates that Hierarchical's use of the sentence context does not compensate for the performance drop due to breaking up the pseudo query in the model's lower layers. Finally, Table 2 shows that QJoint consistently and statistically significantly outperforms all other methods. The improvement over CONCAT shows the merit of utilizing the sentence context, since CONCAT is a special case of QJoint that does not use it.

Analysis of Retrieval Contexts. To further study the merit of using both the dialogue and sentence contexts, we trained the QJoint model (i) with no context, (ii) only with sentence context, (iii) only with dialogue context and (iv) with both (the full model).

Table 3 shows that using only the sentence context without the dialogue context yields performance that is statistically significantly indistiguishable from that of not using context at all, and statistically significantly inferior to using

Table 4: Sentence context in QJoint. 's' and 'p': statistical significance w.r.t. *LocalSurround* and *LocalPrev*, respectively.

	MAP	NDCG@5	MRR
QJoint $_{LocalSurround}$	$\mathbf{.477}^{\pm.010}$	$\mathbf{.644}^{\pm.012}$	$.609^{\pm.013}$
QJoint $_{LocalPrev}$	$.476^{\pm.010}$	$.639^{\pm.013}$	$\mathbf{.612}^{\pm.015}$
QJoint $_{Global}$	$.467^{\pm.011}_{sp}$	$.623^{\pm.013}_{sp}$	$.601^{\pm.015}_{sp}$

Table 5: The effect of the number of past turns (h) used as dialogue context on QJoint's performance. Statistical significance w.r.t. $h = 0, 1, 2$ is marked with 0, 1 and 2, respectively.

	MAP	NDCG@5	MRR
QJoint ($h = 0$)	$.351^{\pm.008}$	$.478^{\pm.014}$	$.463^{\pm.013}$
QJoint ($h = 1$)	$.430^{\pm.009}_{0}$	$.589^{\pm.014}_{0}$	$.566^{\pm.014}_{0}$
QJoint ($h = 2$)	$.472^{\pm.010}_{01}$	$.638^{\pm.012}_{01}$	$.604^{\pm.012}_{01}$
QJoint ($h = 3$)	$\mathbf{.477}^{\pm.010}_{01}$	$\mathbf{.644}^{\pm.012}_{01}$	$\mathbf{.609}^{\pm.013}_{012}$

only the dialogue context. Yet, using both dialogue and sentence contexts yields statistically significant improvements over using just the dialogue context. This result indicates that while the sentence context does not help by itself, it is beneficial when used together with the dialogue context.

Thusfar, the context of sentence s, $CX(s)$, was the two sentences that surround it in the document. Table 4 presents the performance of our best method, QJoint, with the alternative sentence contexts described in Section 4. We see that both *LocalSurround* and *LocalPrev* statistically significantly outperform *Global*, which is aligned with findings in some work on question answering [18]. This attests to the merits of using the "local context"; i.e., the sentences around the candidate sentence. There are no statistically significant differences between *LocalSurround* and *LocalPrev*, providing flexibility to choose local context based on other constraints; e.g., input size.

Heretofore, we used the $h = 3$ turns that precede the current (last) turn in the dialogue as the dialogue context. Table 5 shows that reducing the number of previous turns for dialogue context results in decreasing performance. The smaller difference between $h = 2$ and $h = 3$ is due to relatively few dialogues in the test set with history longer than 2 turns (about 15%). For these dialogues, the difference in performance between $h = 2$ and $h = 3$ is similar to that between $h = 1$ and $h = 2$.

6 Conclusions and Future Work

We addressed the task of retrieving sentences that contain information useful for generating the next turn in an open-ended dialogue. Our approaches utilize both

the dialogue context and the context of candidate sentences in their ambient documents. Specifically, we presented architectures that utilize various hierarchies of the match between a sentence, its context and the dialogue context. Empirical evaluation demonstrated the merits of our best performing approach.

We intend to explore the use of transformers for long texts [5, 1, 44] to overcome the input size limitation. We also plan to ground generative language models with our retrieval models and study the conversations that emerge from such grounding.

Acknowledgements. We thank the reviewers for their comments. This work was supported in part by a grant from Google.

References

1. Beltagy, I., Peters, M.E., Cohan, A.: Longformer: The long-document transformer. CoRR **abs/2004.05150** (2020), `https://arxiv.org/abs/2004.05150`
2. Callan, J.P.: Passage-level evidence in document retrieval. In: Proceedings of SIGIR. pp. 302–310 (1994)
3. Campos, J.A., Otegi, A., Soroa, A., Deriu, J., Cieliebak, M., Agirre, E.: DoQA - accessing domain-specific FAQs via conversational QA. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 7302–7314 (2020)
4. Choi, E., He, H., Iyyer, M., Yatskar, M., Yih, W.t., Choi, Y., Liang, P., Zettlemoyer, L.: QuAC: Question answering in context. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. pp. 2174–2184 (2018)
5. Choromanski, K., Likhosherstov, V., Dohan, D., Song, X., Gane, A., Sarlós, T., Hawkins, P., Davis, J., Mohiuddin, A., Kaiser, L., Belanger, D., Colwell, L., Weller, A.: Rethinking attention with performers. CoRR **abs/2009.14794** (2020), `https://arxiv.org/abs/2009.14794`
6. Cormack, G.V., Clarke, C.L., Buettcher, S.: Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval. pp. 758–759 (2009)
7. Dalton, J., Xiong, C., Callan, J.: TREC cast 2019: The conversational assistance track overview (2020)
8. Dalton, J., Xiong, C., Callan, J.: Cast 2020: The conversational assistance track overview. In: Proceedings of TREC (2021)
9. Dalton, J., Xiong, C., Callan, J.: Trec cast 2021: The conversational assistance track overview. In: Proceedings of TREC (2022)
10. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. CoRR **abs/1810.04805** (2018)
11. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). pp. 4171–4186 (2019)

12. Dinan, E., Roller, S., Shuster, K., Fan, A., Auli, M., Weston, J.: Wizard of wikipedia: Knowledge-powered conversational agents. In: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019 (2019)
13. Fernández, R.T., Losada, D.E., Azzopardi, L.: Extending the language modeling framework for sentence retrieval to include local context. Inf. Retr. **14**(4), 355–389 (2011)
14. Gao, J., Galley, M., Li, L.: Neural approaches to conversational AI. CoRR (2018)
15. Gao, J., Xiong, C., Bennett, P., Craswell, N.: Neural approaches to conversational information retrieval. CoRR **abs/2201.05176** (2022)
16. Glaese, A., McAleese, N., Trebacz, M., Aslanides, J., Firoiu, V., Ewalds, T., Rauh, M., Weidinger, L., Chadwick, M., Thacker, P., Campbell-Gillingham, L., Uesato, J., Huang, P.S., Comanescu, R., Yang, F., See, A., Dathathri, S., Greig, R., Chen, C., Fritz, D., Elias, J.S., Green, R., Mokrá, S., Fernando, N., Wu, B., Foley, R., Young, S., Gabriel, I., Isaac, W., Mellor, J., Hassabis, D., Kavukcuoglu, K., Hendricks, L.A., Irving, G.: Improving alignment of dialogue agents via targeted human judgements (2022)
17. Gopalakrishnan, K., Hedayatnia, B., Chen, Q., Gottardi, A., Kwatra, S., Venkatesh, A., Gabriel, R., Hakkani-Tür, D.: Topical-Chat: Towards Knowledge-Grounded Open-Domain Conversations. In: Proc. Interspeech 2019. pp. 1891–1895 (2019). https://doi.org/10.21437/Interspeech.2019-3079, `http://dx.doi.org/10.21437/Interspeech.2019-3079`
18. Han, R., Soldaini, L., Moschitti, A.: Modeling context in answer sentence selection systems on a latency budget. CoRR **abs/2101.12093** (2021)
19. Harel, I., Taitelbaum, H., Szpektor, I., Kurland, O.: A dataset for sentence retrieval for open-ended dialogues. CoRR (2022)
20. Huang, H., Choi, E., Yih, W.: Flowqa: Grasping flow in history for conversational machine comprehension. In: Proceedings of ICLR (2019)
21. Huang, M., Zhu, X., Gao, J.: Challenges in building intelligent open-domain dialog systems. ACM Trans. Inf. Syst. **38**(3), 21:1–21:32 (2020)
22. Humeau, S., Shuster, K., Lachaux, M., Weston, J.: Poly-encoders: Architectures and pre-training strategies for fast and accurate multi-sentence scoring. In: 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020 (2020)
23. Komeili, M., Shuster, K., Weston, J.: Internet-augmented dialogue generation. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 8460–8478 (2022)
24. Lauriola, I., Moschitti, A.: Answer sentence selection using local and global context in transformer models. In: Proceedings of ECIR. pp. 298–312 (2021)
25. Li, C., Yates, A., MacAvaney, S., He, B., Sun, Y.: Parade: Passage representation aggregation for document reranking. arXiv preprint arXiv:2008.09093 (2020)
26. Lin, S., Yang, J., Nogueira, R., Tsai, M., Wang, C., Lin, J.: Conversational question reformulation via sequence-to-sequence architectures and pretrained language models (2020)
27. Lin, S., Yang, J., Nogueira, R., Tsai, M., Wang, C., Lin, J.: Query reformulation using query history for passage retrieval in conversational search. CoRR (2020)
28. Ma, L., Zhang, W., Li, M., Liu, T.: A survey of document grounded dialogue systems (DGDS). CoRR **abs/2004.13818** (2020)
29. Mehrotra, S., Yates, A.: MPII at TREC cast 2019: Incoporating query context into a BERT re-ranker. In: Voorhees, E.M., Ellis, A. (eds.) Proceedings of TREC (2019)

30. Murdock, V., Croft, W.B.: A translation model for sentence retrieval. In: Proceedings of HLT/EMNLP. pp. 684–695 (2005)
31. Nguyen, T., Rosenberg, M., Song, X., Gao, J., Tiwary, S., Majumder, R., Deng, L.: Ms marco: A human-generated machine reading comprehension dataset (2016)
32. Nogueira, R., Cho, K.: Passage re-ranking with bert. arXiv preprint arXiv:1901.04085 (2019)
33. Qiao, Y., Xiong, C., Liu, Z., Liu, Z.: Understanding the behaviors of BERT in ranking. CoRR (2019)
34. Qin, L., Galley, M., Brockett, C., Liu, X., Gao, X., Dolan, B., Choi, Y., Gao, J.: Conversing by reading: Contentful neural conversation with on-demand machine reading. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 5427–5436. Association for Computational Linguistics, Florence, Italy (Jul 2019). https://doi.org/10.18653/v1/P19-1539, https://www.aclweb.org/anthology/P19-1539
35. Qu, C., Yang, L., Qiu, M., Croft, W.B., Zhang, Y., Iyyer, M.: BERT with history answer embedding for conversational question answering. In: Proceedings SIGIR. pp. 1133–1136
36. Radlinski, F., Craswell, N.: A theoretical framework for conversational search. In: Proceedings of CHIIR. pp. 117–126 (2017)
37. Reddy, S., Chen, D., Manning, C.D.: Coqa: A conversational question answering challenge. Trans. Assoc. Comput. Linguistics **7**, 249–266 (2019)
38. Ren, G., Ni, X., Malik, M., Ke, Q.: Conversational query understanding using sequence to sequence modeling. In: Proceedings of WWW. pp. 1715–1724 (2018)
39. Robertson, S., Walker, S., Jones, S., Hancock-Beaulieu, M., Gatford, M.: Okapi at trec-3. In: Proceedings of TREC-3. pp. 109–126 (1995)
40. Stamatis, V., Azzopardi, L., Wilson, A.: VES team at TREC conversational assistance track (cast) 2019. In: Proceedings of TREC (2019)
41. Tan, C., Wei, F., Zhou, Q., Yang, N., Du, B., Lv, W., Zhou, M.: Context-aware answer sentence selection with hierarchical gated recurrent neural networks. IEEE ACM Trans. Audio Speech Lang. Process. **26**(3), 540–549 (2018)
42. Thoppilan, R., De Freitas, D., Hall, J., Shazeer, N., Kulshreshtha, A., Cheng, H.T., Jin, A., Bos, T., Baker, L., Du, Y., et al.: Lamda: Language models for dialog applications. arXiv preprint arXiv:2201.08239 (2022)
43. Voskarides, N., Li, D., Ren, P., Kanoulas, E., de Rijke, M.: Query resolution for conversational search with limited supervision. In: Proceedings of SIGIR. pp. 921–930 (2020)
44. Wang, S., Li, B.Z., Khabsa, M., Fang, H., Ma, H.: Linformer: Self-attention with linear complexity. CoRR **abs/2006.04768** (2020), https://arxiv.org/abs/2006.04768
45. Yan, R., Song, Y., Wu, H.: Learning to respond with deep neural networks for retrieval-based human-computer conversation system. In: Proceedings of SIGIR. pp. 55–64 (2016)
46. Yan, R., Zhao, D.: Coupled context modeling for deep chit-chat: Towards conversations between human and computer. In: Guo, Y., Farooq, F. (eds.) Proceedings of SIGKDD. pp. 2574–2583 (2018)
47. Zamani, H., Trippas, J.R., Dalton, J., Radlinski, F.: Conversational information seeking. CoRR (2022)
48. Zhang, Z., Li, J., Zhu, P., Zhao, H., Liu, G.: Modeling multi-turn conversation with deep utterance aggregation. In: Proceedings of COLING. pp. 3740–3752
49. Zhu, C., Zeng, M., Huang, X.: Sdnet: Contextualized attention-based deep network for conversational question answering. CoRR **abs/1812.03593** (2018)