Benchmark and Neural Architecture for Conversational Entity Retrieval from a Knowledge Graph

Mona Zamiri
mona.zamiri@wayne.edu
Department of Computer Science
Wayne State University
Detroit, USA

Yao Qiang
yao@wayne.edu
Department of Computer Science
Wayne State University
Detroit, USA

Fedor Nikolaev
fedor@wayne.edu
Department of Computer Science
Wayne State University
Detroit, USA

Dongxiao Zhu
dzhu@wayne.edu
Department of Computer Science
Wayne State University
Detroit, USA

Alexander Kotov
kotov@wayne.edu
Department of Computer Science
Wayne State University
Detroit, USA

ABSTRACT

This paper introduces a novel information retrieval (IR) task of Conversational Entity Retrieval from a Knowledge Graph (CER-KG), which extends non-conversational entity retrieval from a knowledge graph (KG) to the conversational scenario. The user queries in CER-KG dialog turns may rely on the results of the preceding turns, which are KG entities. Similar to the conversational document IR, CER-KG can be viewed as a sequence of interrelated ranking tasks. To enable future research on CER-KG, we created QBLink-KG, a publicly available benchmark that was adapted from QBLink, a benchmark for text-based conversational reading comprehension of Wikipedia. As an initial approach to CER-KG, we experimented with Transformer- and LSTM-based query encoders in combination with the Neural Architecture for Conversational Entity Retrieval (NACER), our proposed feature-based neural architecture for entity ranking in CER-KG. NACER computes the ranking score of a candidate KG entity by taking into account diverse lexical and semantic matching signals between various KG components in its neighborhood, such as entities, categories, and literals, as well as entities in the results of the preceding turns in dialog history. The reported experimental results reveal the key challenges of CER-KG along with the possible directions for new approaches to this task.

CCS CONCEPTS

• **Information systems → Retrieval models and ranking**.

KEYWORDS

Conversational IR, Entity Retrieval, Knowledge Graphs, Deep Learning, IR Benchmarks

ACM Reference Format:
Mona Zamiri, Yao Qiang, Fedor Nikolaev, Dongxiao Zhu, and Alexander Kotov. 2024. Benchmark and Neural Architecture for Conversational Entity Retrieval from a Knowledge Graph. In *Proceedings of the ACM Web Conference 2024 (WWW '24), May 13–17, 2024, Singapore, Singapore.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3589334.3645676

WWW '24, May 13–17, 2024, Singapore, Singapore

ACM ISBN 979-8-4007-0171-9/24/05
https://doi.org/10.1145/3589334.3645676

1 INTRODUCTION

The recent advances in deep learning have propelled human-machine dialog from the narrow confines of scripted task completion into everyone's daily life. With the growing popularity of mobile devices and digital personal assistants, the human-machine dialog is well-poised to soon become the primary modality for information seeking. In conversational information seeking [11], users engage in a dialog with a search system to address their information needs. Producing a search system's response for user utterances in information-seeking dialogues requires leveraging a wide variety of sources (text collections, knowledge graphs, tables, and databases) and an even wider variety of approaches that can utilize these sources along with the dialog context in the form of the preceding dialog turns.

Prior research on conversational information seeking focused on two major directions - conversational question answering (QA) and conversational information retrieval (IR). Conversational QA has been well-studied in the scenarios that involve documents [25, 41–44, 53], knowledge graphs [8, 17, 23, 24, 34, 46, 48], tables [22] and their combinations, such as KG and documents [49, 50] or KG, documents and tables [9]. Conversational IR research, however, has so far only focused on documents [18, 29, 54], whereas **entity retrieval from a KG has not yet been studied in a conversational setting.** To address this oversight, we introduce a novel task of **C**onversational **E**ntity **R**etrieval from a **K**nowledge **G**raph (CER-KG) summarized in Figure 1 and defined as follows:

Definition 1. ***Conversational Entity Retrieval from a Knowledge Graph*** *is an IR task that focuses on retrieving a KG entity in response to a free-form query that may explicitly or implicitly rely on the dialog context.*

This definition leads to several important differences between CER-KG and Conversational QA from a KG (CQA-KG). From a conceptual perspective, CER-KG extends entity retrieval from a KG to a dialog setting. Similar to conversational document IR [12], CER-KG can thus be viewed as a sequence of interrelated rounds

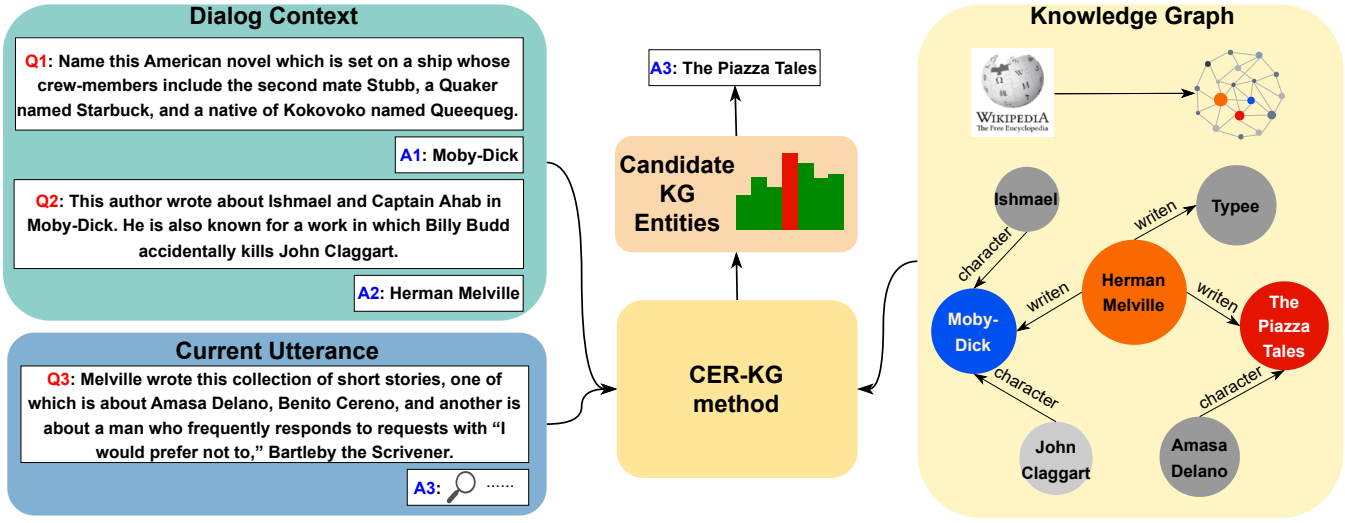

Figure 1: Overview of the proposed task of Conversational Entity Retrieval from a KG (CER-KG).

of candidate KG entity retrieval and ranking. Correspondingly, the key challenges of CER-KG are the identification of a comprehensive set of candidate answer entities in a KG and the effective relevance signals and methods to translate those signals into the accurate ranking of candidate entities. On the other hand, CQA-KG and QA from a KG, which it extends, can be viewed as a sequence of interrelated inference and reasoning procedures over a KG subset. The key challenges of those procedures are the discovery of methods that can simultaneously perform logical, comparative, quantitative and verification reasoning, and infer the answers that may not be present in a KG.

There are also notable differences in the benchmarks proposed for these tasks. First, unlike short automatically constructed questions with a single focal entity typical of the datasets for CQA-KG, such as CSQA [46] or ConvQuestions [8], QBLink-KG, our benchmark for CER-KG, makes less strict assumptions about the structure of the queries (as follows from Figure 1, the manually written queries in QBLink-KG can be arbitrarily long and include multiple entity mentions) or the nature of the answer entity (unlike the answer entities to simple questions in CSQA, which are restricted only to the object position of KG triplets, the answer entities in CER-KG can be in the subject or object position of KG triplets). Questions in CSQA, on the other hand, can have other answer types besides KG entities (e.g. numbers, dates, yes/no) that may not exist in the KG or have no answer at all. Overall, CER-KG complements CQA-KG in the ecosystem of methods for different types of information needs that may arise in real-life conversational information-seeking interactions.

As the first approach to CER-KG, we propose a **N**eural **A**rchitecture for **C**onversational **E**ntity **R**etrieval (NACER), a feature-based neural architecture to point-wise ranking of candidate KG entities for each dialog turn. Rather than taking distributed representations of the current dialog turn, dialog context, and a candidate KG entity to assess relevance internally, NACER directly utilizes diverse relevance signals in the form of input features that capture semantic and lexical similarities between a current dialog turn, preceding answer(s) and candidate entity's neighboring KG components, such as entities, categories, and literals. The candidate KG entities are then ranked according to their relevance scores computed by NACER. *In principal, NACER can be used along with CQA-KG methods to produce responses at appropriate turns of the same information-seeking dialog.*

To evaluate NACER[1] and enable future research on CER-KG, we adapted QBLink [15], an existing benchmark for conversational reading comprehension of Wikipedia, to construct QBLink-KG[2], a CER-KG benchmark for DBpedia [28].

2 RELATED WORK

2.1 Non-conversational entity retrieval from a KG

Benchmarks for non-conversational entity retrieval from a KG, such as DBPedia-Entity v2 [19], aim at finding an entity, an attribute of an entity, or a list of entities in response to a keyword query or a question. Traditional IR methods proposed for this task [7, 38, 59] construct structured documents for each KG entity and aim to correctly weigh and aggregate lexical matches of the key query concepts in different fields of structured entity documents to obtain the entity ranking score. The neural architectures proposed for this task range from feed-forward neural networks with attention [2] to transformers [6, 13, 16, 57] and aim to match dense representations of textual queries and KG entities.

2.2 QA and CQA from a KG

Prior research on QA from a KG independently studied simple and complex questions. Simple questions, such as those in the SimpleQuestions benchmark [3], correspond to a single KG triplet, in

[1] source code available at https://doi.org/10.5281/zenodo.10685904
[2] available at https://doi.org/10.6084/m9.figshare.25256290

which the entity in the subject position is mentioned in a question and the entity in the object position is the answer. Existing approaches for simple QA from a KG can be grouped into two categories: end-to-end neural networks [20, 32] and pipelined approaches [31, 36, 39, 52, 58].

Property	SQA	QA	CQA	ER	CER
Involves a multi-turn dialog	✗	✗	✓	✗	✓
Answer is present in a KG	✓	✗	✗	✓	✓
Answer is a KG entity	✓	✗	✗	✓	✓
Multiple types of answers or no answer	✗	✓	✓	✗	✗
Answer requires reasoning and/or inference	✗	✓	✓	✗	✗
Anaphoras, co-references and ellipses	✗	✗	✓	✗	✗

Table 1: Summary of the key properties of Simple Question Answering (SQA), Complex Question Answering (QA), Conversational Question Answering (CQA), Entity Retrieval (ER) and Conversational Entity Retrieval (CER) from a KG.

Complex QA from a KG has been well-studied in both non-conversational [5, 21, 30, 40, 47] and conversational [8, 17, 23, 24, 34, 48] settings. The major challenge of complex questions is that answering them requires multi-hop traversal of a KG, performing reasoning, comparison, counting or set operations over a subset of a KG to discover the facts that may not be explicitly present in a KG. These challenges have been addressed with heuristic approaches [8], multi-hop inference [30, 47], reinforcement learning [24] and semantic parsing into an executable logical form [17, 21, 23, 34, 40, 48] or a specialized language to represent the reasoning process [5]. Conversational setting introduces additional challenges of resolving anaphoras, co-references, and ellipses.

The key properties of CER-KG and the related tasks are summarized in Table 1, from which it follows that CER-KG methods cannot be evaluated on CQA-KG benchmarks and vice versa.

3 QBLINK-KG

QBLink-KG, our proposed benchmark for CER-KG, was adapted from QBLink [15], a benchmark for conversational reading comprehension over Wikipedia. QBLink consists of a short lead and a series of up to three queries (all are hand-crafted), the answers to which are single named entities corresponding to the titles of Wikipedia articles. Formally, the task of CER-KG is to retrieve the correct answer entity a_k from a KG in response to a query q_k in the kth dialog turn given the dialogue context, which includes all preceding queries $q_1, \ldots, q_{k-1}$ and answers $a_1, \ldots, a_{k-1}$ to them.

We used the English subset of the September 2021 DBpedia snapshot[3] as the target KG for QBLink-KG. Since DBpedia is constructed through information extraction from Wikipedia infoboxes [28], QBLink answers provided as the titles of Wikipedia articles can be easily converted to DBpedia entity URIs, if the corresponding entities exist in DBpedia.

QBLink cannot be utilized for CER-KG in its original form since knowledge graphs (even those derived from Wikipedia) contain significantly less information than Wikipedia. Specifically, a named entity that is the answer to a QBLink question may not exist as an entity in DBpedia. To adapt QBLink to CER from DBpedia, we performed two necessary filtering steps described below. The total number of queries in each split of the benchmark after each filtering step are summarized in Table 2.

Filtering step	Train	Valid	Test
No filtering	68,454	5,451	9,597
`wiki_page` $\neq \emptyset$	59,796	4,772	8,436
Target entity $\in \mathcal{Y}$	14,586	1,100	1,682

Table 2: Total number of queries in each split of the benchmark after each filtering step.

First, we filtered out all QBLink queries that are unusable for the benchmark regardless of entity linking and candidate selection methods (i.e. all queries with an empty `wiki_page` field or those queries for which the answer does not correspond to a Wikipedia page or cannot be mapped to a DBpedia entity). For the evaluation of NACER and the baselines with specific entity linking and candidate selection methods used in this work, we then filtered out the queries with the answers that do not belong to the set of candidate entities $\mathcal{Y}$ obtained with these methods.[4] The final statistics of QBLink-KG are shown in Table 3.

Statistic	Train	Valid	Test
Total words	388,900	30,397	53,025
Distinct words	37,722	8,261	11,897
Avg. words per query	26.66	27.36	26.25

Table 3: Statistics of QBLink-KG.

As follows from Table 3, the queries in QBLink-KG are verbose, with over 20 words per query on average.

3.1 Entity linking and selection of candidate entities

Both NACER and the baselines utilize the same set of candidate entities $\mathcal{Y}$ generated based on the set of entities $\mathcal{E} = \{e_l^1, \ldots, e_l^r\}$ linked from q_k, as shown in Figure 3. The entities linked to q_k were obtained using the method proposed in [32][5], which proved to be effective for non-conversational simple QA from a KG. A set of candidate answer entities $\mathcal{Y}$ was obtained by including all other entities in the same triplets with the entities in $\mathcal{E}$. To prevent an explosion of the set of candidate entities, we did not consider linked entities in q_k with a degree greater than 100.

4 NACER

To identify the most effective types of relevance signals for CER-KG, we propose NACER, a feature-based neural architecture for KG entity ranking. As shown in Figure 2, NACER has a modular architecture consisting of three main components: the encoding layer,

[3] https://databus.dbpedia.org/dbpedia/collections/dbpedia-snapshot-2021-09

[4] to enable experiments with other entity linking and candidate entity selection methods, we release both filtered and unfiltered versions of QBLink-KG

[5] with the only difference that the linked entities can be subjects or objects of KG triplets

the matching feature aggregation layers, and the entity relevance score computation layer.

4.1 Encoding Layer

Features. NACER computes the score of each candidate KG entity $y_i \in \mathcal{Y}$ based on the feature vector $\bar{y}_i$ constructed based on q_k, a_{k-1}[6] and $\mathcal{T}_i$, a set of all KG triplets that include y_i, as detailed in Table 4. The feature vector $\bar{y}_i$ for y_i consists of the features derived using either semantic similarity function $f_e(\mathbf{a}, \mathbf{b})$ or lexical similarity function $f_w(a, b)$ based on: (1) lexical and distributed representations of KG structural components (entities, predicates, literals and categories) in $\mathcal{T}_i$; (2) lexical and distributed representations of q_k; (3) lexical and distributed representations of a_{k-1}:

$$\begin{aligned}\bar{y}_i = [&\texttt{ent}_e, \texttt{pred}_e, \texttt{lit}_e, \texttt{cat}_e, \texttt{ans-1}_e,\\ &\texttt{ent}_w, \texttt{pred}_w, \texttt{lit}_w, \texttt{cat}_w, \texttt{ans-1}_w].\end{aligned} \tag{1}$$

The first five features are calculated using f_e, while the last five features are calculated using f_w, as detailed in Table 4.

We experiment with three parametric and non-parametric variants of $f_e(\mathbf{a}, \mathbf{b})$ to determine the degree of similarity between the distributed representations of $\mathbf{a}$ and $\mathbf{b}$: (1) dot product $f_{e\text{-dot}}(\mathbf{a}, \mathbf{b}) = \mathbf{a}^\top \mathbf{b}$; (2) multiplicative interaction function $f_{e\text{-mult}}(\mathbf{a}, \mathbf{b}) = \mathbf{a}^\top \mathbf{W} \mathbf{b}$ with trainable parameter $\mathbf{W}$; (3) additive interaction function $f_{e\text{-add}}(\mathbf{a}, \mathbf{b}) = \mathbf{v}^\top \tanh(\mathbf{W}_a \mathbf{a} + \mathbf{W}_b \mathbf{b})$ with trainable parameters $\mathbf{v}$, $\mathbf{W}_a$ and $\mathbf{W}_b$. The parameters $\mathbf{W}$ for the multiplicative interaction function, and $\mathbf{v}, \mathbf{W}_a, \mathbf{W}_b$ for the additive interaction function can be either shared between $\texttt{ent}_e$, $\texttt{pred}_e$, $\texttt{lit}_e$, $\texttt{cat}_e$, $\texttt{ans-1}_e$ features or trained independently for each feature (column **par. sharing** in Table 5).

$f_w(a, b)$ utilizes the bag-of-words representations of $a = \{a_1, \ldots, a_n\}$ and $b = \{b_1, \ldots, b_m\}$ to quantify the lexical similarity as a sum of *smooth inverse frequencies* [1] of their overlapping terms:

$$f_w(a, b) = \sum_{w \in a \cap b} \frac{\lambda}{\lambda + n(w)}, \tag{2}$$

where λ is a hyper-parameter and $n(w)$ is KG frequency of term w.

Embeddings. We used the publicly available[7] embeddings of words and KG structural components (entities, predicates, categories, and literals) obtained using KEWER method [37] in the encoding layer of NACER and for feature computation.

Turn encoding methods. $\mathbf{a}_{k-1}$, a distributed representation of the preceding answer in the dialog, and $\mathbf{q}_k$, a distributed representation of the kth query in a CER-KG information-seeking dialog, are created in the encoding layer. We consider four options for dialog turn encoding: (1) **KEWER**: calculating the weighted mean of KEWER embeddings of the words and entities in q_k; (2) **BiLSTM**: embedding q_k using a pre-trained BiLSTM with max-pooling [10]; (3) **BERT**: embedding q_k with a pre-trained BERT [14]; (4) **BERT+KEWER**: embedding q_k with the K-Adapter [55], a framework enabling to inject KG-specific information encoded in KEWER embeddings into the distributed representation of $\mathbf{q}_k$ created with pre-trained BERT.

[6] without the loss of generality, we limit the discussion to only the answer to the previous turn a_{k-1}. However, features based on $a_1, \ldots, a_{k-2}$ can easily be added to $\bar{y}_i$ (see results and analysis in Section 6.2)

[7] https://academictorrents.com/details/4778f904ca10f059eaaf27bdd61f7f7fc93abc6e

4.2 Feature aggregation and score computation layers

Each candidate answer entity y_i for the kth turn is then ranked based on its logit score:

$$\begin{aligned}p_{\text{logit}}(y_i | q_k, a_{k-1}, \mathcal{T}_i) = \\ \mathbf{w}_s^\top \sigma(\mathbf{W}_{a_2}^\top \sigma(\mathbf{W}_{a_1}^\top \bar{y}_i + \mathbf{b}_{a_1}) + \mathbf{b}_{a_2}) + b_s,\end{aligned} \tag{3}$$

where $\mathbf{W}_{\{a_1,a_2\}}$ and $\mathbf{b}_{\{a_1,a_2\}}$ are the weights and biases in the matching feature aggregation layers (we use two in Eq. 3, but the number can vary); $\mathbf{w}_s$ is a weight vector of the size determined by the number of neurons in the final matching feature aggregation layer; b_s is a scalar bias of the entity score computation layer, and p_{logit} denotes a non-normalized logit probability, which is passed through softmax during calculation of the loss function.

4.3 Loss function

Cross-entropy between one-hot distribution for the target entity y_t and the entity logit score from Eq. (3) was used as the loss function.

5 EXPERIMENTAL SETUP

5.1 Baselines

BM25F. As an established baseline using only lexical matching, we utilized BM25F [45], an extension of the popular BM25 retrieval model to structured (i.e. multi-field) documents. To adapt BM25F to the conversational retrieval scenario, we included a_{k-1} into q_k. DBpedia entities were converted into 4-field (entity names, attributes, categories and related entity names) entity documents using the method from [37][8]. We experimented with BM25F using the BM25 parameter settings recommended in the literature (b_f = 0.75, w_f = 1.0 set uniformly for each field and k_1 = 1.2) [33, p. 233] (BM25F$_{orig}$), and optimized the model using coordinate ascent based on 100 queries and answers randomly selected from the training set (BM25F$_{CA}$).

GENRE. We utilized GENRE [13], a Transformer-based model proposed for non-conversational entity retrieval, as a task-specific neural generative baseline. Instead of retrieving answer entities, GENRE directly generates their surface forms token-by-token in an auto-regressive manner. As a model fine-tuning BART for entity retrieval from Wikipedia and employing a constrained decoding strategy that forces generated text to be entities relevant to a query, GENRE is a strong baseline, which was shown to be superior to purely semantic matching-based entity retrieval methods using maximum-inner-product search over distributed representations of queries and entities. To adapt GENRE to the conversational retrieval scenario, we supply a_{k-1} and q_k into GENRE's encoder and map the generated surface forms of answer entities to DBpedia URIs.

LLaMa. We utilized LLaMa 2 [51] (specifically `llama-2-7b`) as a foundation large language model (LLM) baseline. The prompt for this model included a detailed description of the task along with 10 examples from the training set of QBLink-KG. Each example included the preceding answer, the current query, and the 10 ranked candidate entities from the set of candidate entities utilized by NACER and the memory network baselines with the correct answer entity ranked at the top.

[8] source code available at https://github.com/teanalab/dbpedia2fields

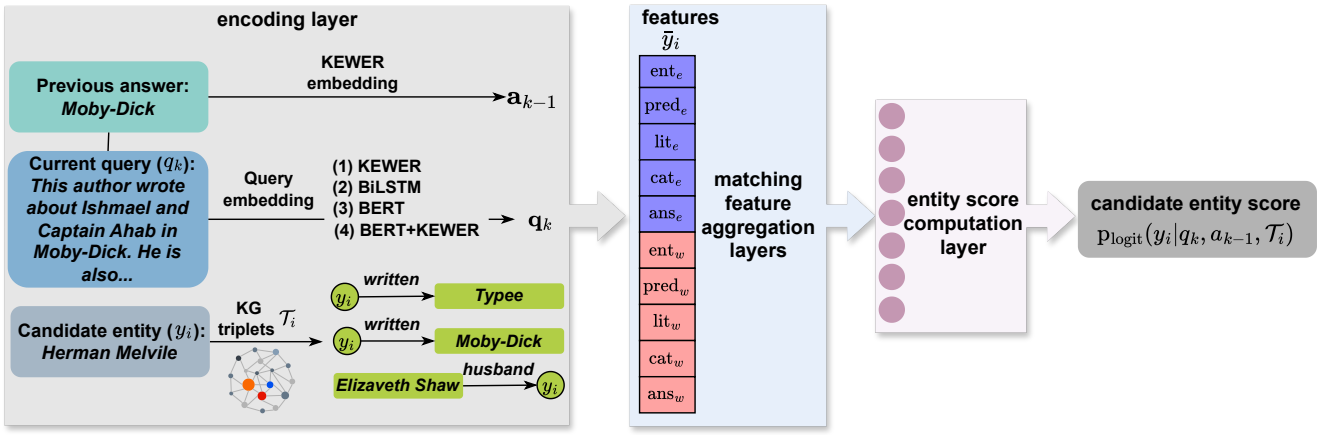

Figure 2: Neural Architecture for Conversational Entity Retrieval from a Knowledge Graph.

Feature	Feature value	Feature description
ent_e	$f_e\left(\mathbf{q}_k, \frac{\sum_{(y_i,p_o,e_o)\in\mathcal{T}_i}\mathbf{e}_o+\sum_{(e_s,p_s,y_i)\in\mathcal{T}_i}\mathbf{e}_s}{\lvert(y_i,p_o,e_o)\in\mathcal{T}_i\rvert+\lvert(e_s,p_s,y_i)\in\mathcal{T}_i\rvert}\right)$	semantic similarity between $\mathbf{q}_k$ and the mean of KEWER embeddings of KG entities that are either subject ($\mathbf{e}_s$) or object ($\mathbf{e}_o$) in the same triplet as y_i
pred_e	$f_e\left(\mathbf{q}_k, \frac{\sum_{(s_j,p_j,o_j)\in\mathcal{T}_i}\mathbf{p}_j}{\lvert(s_j,p_j,o_j)\in\mathcal{T}_i\rvert}\right)$	semantic similarity between $\mathbf{q}_k$ and the mean of KEWER embeddings of predicates $\mathbf{p}_j$ from the triplets in $\mathcal{T}_i$
lit_e	$f_e\left(\mathbf{q}_k, \frac{\sum_{(y_i,p_j,l_j)\in\mathcal{T}_i}\mathbf{l}_j}{\lvert(y_i,p_j,l_j)\in\mathcal{T}_i\rvert}\right)$	semantic similarity between $\mathbf{q}_k$ and the mean of embeddings $\mathbf{l}_j$ of literals from $\mathcal{T}_i$. $\mathbf{l}_j$ is calculated as the mean of KEWER embeddings of tokens in l_j
cat_e	$f_e\left(\mathbf{q}_k, \frac{\sum_{(y_i,c_j)\in\mathcal{T}_i}\mathbf{c}_j}{\lvert(y_i,c_j)\in\mathcal{T}_i\rvert}\right)$	semantic similarity between $\mathbf{q}_k$ and the mean of KEWER embeddings of categories $\mathbf{c}_j$ that y_i belongs to
ans-1_e	$f_e\left(\mathbf{a}_{k-1}, \frac{\sum_{(y_i,p_j,o_j)\in\mathcal{T}_i}\mathbf{o}_j+\sum_{(e_s,p_s,y_i)\in\mathcal{T}_i}\mathbf{e}_s}{\lvert(y_i,p_j,o_j)\in\mathcal{T}_i\rvert+\lvert(e_s,p_s,y_i)\in\mathcal{T}_i\rvert}\right)$	semantic similarity between $\mathbf{a}_{k-1}$ and the mean of KEWER embeddings of objects ($\mathbf{o}_j$) or subjects ($\mathbf{e}_s$) in the same triplets as y_i (o_j can be an entity, literal, or category)
ent_w	$\frac{\sum_{(y_i,p_o,e_o)\in\mathcal{T}_i} f_w(q_k,e_o)+\sum_{(e_s,p_s,y_i)\in\mathcal{T}_i} f_w(q_k,e_s)}{\lvert(y_i,p_o,e_o)\in\mathcal{T}_i\rvert+\lvert(e_s,p_s,y_i)\in\mathcal{T}_i\rvert}$	average lexical similarity between q_k and the labels of KG entities that are either a subject (e_s) or an object (e_o) in the same triplet with y_i
pred_w	$\frac{\sum_{(s_j,p_j,o_j)\in\mathcal{T}_i} f_w(q_k,p_j)}{\lvert(s_j,p_j,o_j)\in\mathcal{T}_i\rvert}$	average lexical similarity between q_k and the labels of predicates p_j from the triplets in $\mathcal{T}_i$
lit_w	$\frac{\sum_{(y_i,p_j,l_j)\in\mathcal{T}_i} f_w(q_k,l_j)}{\lvert(y_i,p_j,l_j)\in\mathcal{T}_i\rvert}$	average lexical similarity between q_k and literals l_j from $\mathcal{T}_i$
cat_w	$\frac{\sum_{(y_i,c_j)\in\mathcal{T}_i} f_w(q_k,c_j)}{\lvert(y_i,c_j)\in\mathcal{T}_i\rvert}$	average lexical similarity between q_k and the labels of all categories c_j that y_i belongs to
ans-1_w	$\frac{\sum_{(y_i,p_j,o_j)\in\mathcal{T}_i} f_w(a_{k-1},o_j)+\sum_{(e_s,p_s,y_i)\in\mathcal{T}_i} f_w(a_{k-1},e_s)}{\lvert(y_i,p_j,o_j)\in\mathcal{T}_i\rvert+\lvert(e_s,p_s,y_i)\in\mathcal{T}_i\rvert}$	average lexical similarity between a_{k-1} and objects (o_j) or subjects (e_s) in the same triplets as y_i (o_j can be an entity, literal, or category)

Table 4: Semantic and lexical similarity features utilized by NACER for scoring candidate answer entities.

KV-MemNN. Memory networks (MemNNs) [56] are a class of differentiable models, which can perform simple inference over structured or unstructured knowledge. Key-value MemNNs [35], in which the memories are indexed by the keys, were shown to be effective at retrieving answers in text-based QA [35], non-conversational simple QA from a KG [3] and conversational QA from a KG [46]. We used the following two adaptations of the Key-Value Memory Network (KV-MemNN) [35] to CER-KG as the baselines. These adaptations differ in the approaches used to fill M key-value memory slots $(k_1, v_1), \ldots, (k_M, v_M)$.

The first approach (named KV-MemNN$_{\text{in}}$) uses a_{k-1} and $e_l^1, \ldots, e_l^r$ and the entities linked from q_k as the keys $k_1, \ldots, k_M$ and entities in the same KG triplets as the values $v_1, \ldots, v_M$. This way, each key-value pair (k_i, v_i) can be constructed from a single KG triplet, in which the subject or object k_i is from the *in-key* set $\{a_{k-1}, e_l^1, \ldots, e_l^r\}$ and the object or subject in the same triplet is used as a value v_i. Key-value memories are represented using the KEWER entity embeddings as $(\mathbf{k}_1, \mathbf{v}_1), \ldots, (\mathbf{k}_M, \mathbf{v}_M)$. The set of entities used as values $\{v_1, \ldots, v_M\}$ is considered as the candidate entities $y_1, \ldots, y_C$. Each candidate entity y_i is scored using $\mathbf{q}_{H+1}$,

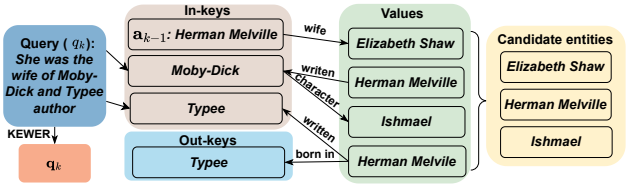

Figure 3: Construction of the key-value memory slot pairs and candidate entities for the KV-MemNN baselines.

the distributed representation of q after H hops over key-value memories and $\mathbf{y}_i$, the KEWER embedding of y_i, as $p_{\text{logit}}(y_i) = \mathbf{q}_H^\top \mathbf{y}_i$.

The second approach (named KV-MemNN$_{\text{out}}$) is identical in all aspects to KV-MemNN$_{\text{in}}$, except that the set of key-value memory slots $(k_1, v_1), \ldots, (k_M, v_M)$ are supplemented with the pairs (k_i, v_i), where the value v_i belongs to the set of candidate entities $\mathcal{Y} = \{y_1, \ldots, y_C\}$ as before, but the *out-key* k_i is not necessarily from the set $\{a_{k-1}, e_l^1, \ldots, e_l^r\}$ and can be any neighbor of the candidate entity y_i (i.e. either a subject or an object in the triplet that contains y_i as an object or a subject). Thus, the construction of memory slots is modified as follows. First, we consider a KG as an undirected graph G, where each subject-predicate-object triplet (s, p, o) corresponds to the undirected edge between the subject s and object o. Second, an additional hop in G is performed starting from the previously obtained value entities v_i to obtain the *out-keys*.

Figure 3 illustrates the KV-MemNN$_{\text{in}}$ and KV-MemNN$_{\text{out}}$ approaches to filling the memory slots. Note that the set of candidate entities $\mathcal{Y}$ in both KV-MemNN$_{\text{in}}$ and KV-MemNN$_{\text{out}}$ is identical to the set of candidate entities used by all variants of NACER, which allows for a fair comparison of NACER with KV-MemNN$_{\{\text{in,out}\}}$.

6 RESULTS

6.1 Retrieval accuracy

To examine different aspects of CER-KG and identify the types of methods that can be employed by effective solutions to it, we experimented with various variants of NACER and different types of baselines on the test set of QBLink-KG. The results of these experiments are presented in Table 5. Several main conclusions can be drawn from the analysis of these results.

First, the retrieval accuracy of NACER and KV-MemNN-based baselines varies significantly depending on the encoder for q_k and the type of matching function used. Although most combinations of NACER with BERT- or BiLSTM-based encoders generally outperformed all lexical, generative and LLM-based baselines, the competitive performance of GENRE and, more surprisingly, optimized BM25F with simple adaptations to the retrieval scenario are notable. The superior performance of NACER over both GENRE and BM25F can be attributed to the need to take into account both semantic and lexical matching signals when quantifying the relevance of the answer entities, possibly due to the verbosity of queries in QBLink-KG. LLaMa 2 performance indicates that answering verbose trivia-style queries in a conversational setting is a challenging task for in-context learning with foundation LLMs.

Second, among all compared models, the NACER with the query encoder using BERT and the KEWER-based K-Adapter, additive interaction function and no parameter sharing resulted in the highest retrieval accuracy. We believe there are two major reasons behind this result. First, as a pre-trained language model, BERT already possesses rich knowledge acquired in an unsupervised manner from Wikipedia. This knowledge allows it to perform slightly better than BiLSTM as a turn encoder when most interaction functions are used to calculate the features capturing semantic similarity between distributed representations of the current turn and components of the KG surrounding the candidate entities. Second, the K-Adapter efficiently injects the KG-specific information captured by KEWER embeddings into BERT allowing it to better capture KG structure when creating a distributed representation of the current query. This ultimately improves the effectiveness of the features capturing semantic similarity of the current query with the candidate entities, which translates into additional performance gains over the pre-trained BERT across most metrics.

Third, the dot product interaction function consistently resulted in the lowest accuracy among all semantic similarity functions utilized by NACER. On the other hand, parametric multiplicative and additive interaction functions increase the capacity of NACER, which translates into improvement in its accuracy. Furthermore, parameter sharing of multiplicative and additive interaction functions has a consistently negative effect on the accuracy across all metrics. NACER paired with different types of turn encoders generally demonstrates better performance without parameter sharing.

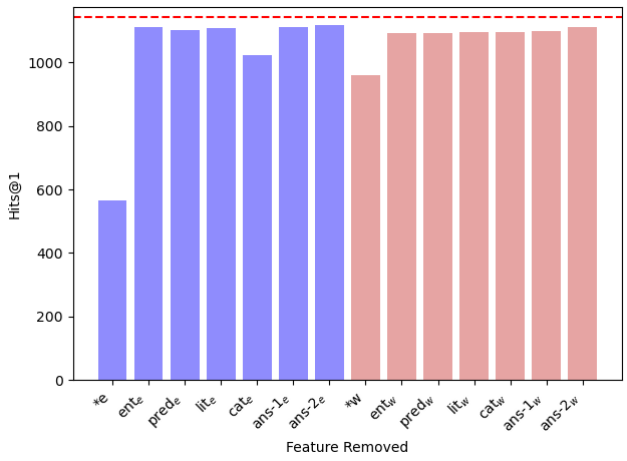

Figure 4: Retrieval accuracy of NACER, when individual, all semantic and all lexical similarity features are removed. The red dotted line is the accuracy of NACER with all features.

Lastly, NACER outperforms KV-MemNN-based baselines across all metrics in combination with any query encoder. The margin of the difference between the best configurations of NACER and KV-MemNN$_{\text{in}}$ ranges from 7% to 13 % for different metrics. This result indicates that, in QBLink-KG, the relevance signals pointing to the correct answer entity are mainly localized within a small neighborhood around that entity in a KG, hence finding the correct answer

Method	q_k encoding	$f_e(\mathrm{a},\mathrm{b})$	par. sharing	Hits@1	R@1	Hits@10	R@10	MRR
BM25F$_{orig}$	-	-	-	373	0.2218	1125	0.6688	0.3639
BM25F$_{CA}$	-	-	-	717*	0.4263*	1481*	0.8810*	0.5877*
GENRE	-	-	-	855*	0.5083*	1045*	0.6213*	0.5495*
LLaMa	-	-	-	383	0.3610	427	0.4030	0.3779
KV-MemNN$_{\mathrm{in}}$	KEWER	-	-	991*	0.5892*	1496*	0.8894*	0.6905*
KV-MemNN$_{\mathrm{in}}$	BiLSTM	-	-	854	0.5077	1449	0.8615	0.6269
KV-MemNN$_{\mathrm{in}}$	BERT	-	-	779	0.4631	1148	0.6825	0.5613
KV-MemNN$_{\mathrm{in}}$	BERT+KEWER	-	-	811	0.4822	1154	0.6861	0.6125
KV-MemNN$_{\mathrm{out}}$	KEWER	-	-	983	0.5844	1431	0.8507	0.6758
KV-MemNN$_{\mathrm{out}}$	BiLSTM	-	-	847	0.5035	1389	0.8258	0.6007
KV-MemNN$_{\mathrm{out}}$	BERT	-	-	765	0.4548	1131	0.6724	0.5512
KV-MemNN$_{\mathrm{out}}$	BERT+KEWER	-	-	802	0.4768	1143	0.6795	0.5587
NACER	KEWER	dot	-	648	0.3853	1314	0.7812	0.5172
NACER	KEWER	mult	Y	782	0.4649	1399	0.8317	0.5824
NACER	KEWER	mult	N	1016*‡	0.6040*‡	1567*‡	0.9316*‡	0.7164*‡
NACER	KEWER	add	Y	865	0.5143	1480	0.8799	0.6361
NACER	KEWER	add	N	977	0.5809	1533‡	0.9114‡	0.6967‡
NACER	BiLSTM	mult	Y	931	0.5535	1531‡	0.9102‡	0.6765
NACER	BiLSTM	mult	N	979	0.5820	1555‡	0.9245‡	0.7029‡
NACER	BiLSTM	add	Y	919	0.5464	1497‡	0.8900‡	0.6613
NACER	BiLSTM	add	N	1053*‡	0.6260*‡	1592*‡	0.9465*‡	0.7389*‡
NACER	BERT	mult	Y	807	0.4798	1439	0.8555	0.6067
NACER	BERT	mult	N	1016‡	0.6064‡	1573‡	0.9352‡	0.7178‡
NACER	BERT	add	Y	938	0.5577	1522‡	0.9049‡	0.6758
NACER	BERT	add	N	1095*‡	0.6510*‡	1600*‡	0.9512*‡	**0.7658***‡
NACER	BERT+KEWER	mult	Y	979	0.5820	1553‡	0.9233‡	0.6993‡
NACER	BERT+KEWER	mult	N	1030‡	0.6124‡	1559‡	0.9269‡	0.7239‡
NACER	BERT+KEWER	add	Y	1048‡	0.6231‡	1569‡	0.9328‡	0.7297‡
NACER	BERT+KEWER	add	N	**1121***‡	**0.6665***‡	**1602***‡	**0.9524***‡	0.7575*‡

Table 5: Accuracy of BM25F, GENRE, LLaMa and different variants of NACER and KV-MemNN on the test set of QBLink-KG. The largest value for each metric is boldfaced. The best performance by each model type is indicated by *. Statistical significance of the difference with KV-MemNN$_{\mathrm{in}}$ and KEWER encoder for q_k based on the two-tailed paired Student's t-test with $p = 0.05$ is indicated by ‡.

entity does not require the multi-hop inference capabilities of key-value memory networks needed to effectively address CQA-KG. Instead, effective methods for CER-KG should focus on identifying, capturing, and combining lexical and semantic matching signals in the immediate KG neighborhood of the answer entity.

6.2 Experiments with features

Feature ablation. To assess the relative importance of NACER features on its performance, we conducted a feature ablation study. In this study, we removed one feature or a set of features at a time and retrained the best-performing configuration of NACER (BERT with KEWER-based K-Adapter as the turn encoder, additive interaction function, and no parameter sharing). We also experimented with two additional configurations, in which all semantic similarity features ($*_e$) and all lexical similarity features ($*_w$) were removed. The resulting Hits@1 values are shown in Figure 4.

As follows from Figure 4, the performance drops significantly when either all semantic or all similarity features are removed, which indicates that both feature types are critical to NACER's performance, with the semantic similarity features playing a more important role than the lexical ones. Removal of most individual features (with a notable exception of $\mathtt{cat}_e$ and $\mathtt{ent}_w$) had a smaller but consistently negative impact on the accuracy of NACER, which indicates that NACER effectively aggregates lexical and semantic matching features into the answer entity score.

Features based on preceding answers. To assess the impact of the dialog context, we measured the retrieval accuracy of NACER when the features based on the preceding dialog turn answer ($\mathtt{ans\text{-}1}_e$ and $\mathtt{ans\text{-}1}_w$) were removed from and the features based on the answer to the two dialog turns prior to the current one ($\mathtt{ans\text{-}2}_e$ and $\mathtt{ans\text{-}2}_w$) were added to $\bar{y}_i$. The results of these experiments in Table 7 highlight the importance of accounting for the dialog context in the form of the answers to preceding queries in CER-KG.

6.3 Success and failure analysis

The top 3 entities ranked by NACER and KV-MemNN$_{\mathrm{in}}$ in combination with different query encoders are shown in Table 6. Examination of the results in this table reveals the qualitative superiority of answers obtained with NACER. Specifically, regardless of the query encoder, NACER was able to rank the correct entity as the top result for 2 out of 3 queries in the example dialog. KV-MemNN$_{in}$, on the other hand, was able to rank the correct entity in the top position only for 1 query and only with 1 query encoder. Regardless of the query encoder, NACER preserved the typical coherence of the

Method	Dialog turn	Top-3 answers and position of the correct answer			
		KEWER	BiLSTM	BERT	BERT+ KEWER
NACER	1. Name this English author of novels like "The Passion of New Eve" and "Nights at the Circus", known especially for feminist reinterpretations of other works	**Angela Carter** Sabine Huynh Janez Menart	**Angela Carter** Sabine Huynh Janez Menart	**Angela Carter** Sabine Huynh Janez Menart	**Angela Carter** Sabine Huynh Peter Russell
		1	1	1	1
	2. Carter wrote a libretto based on this Virginia Woolf novel, whose protagonist has affairs with Queen Elizabeth I and the princess Sasha and is mentored by Nicholas Greene while writing a long poem called "The Oak Tree"	Freshwater (play) The Waves Vanessa Bell	The Waves Nights at the Circus Wise Children	The Waves **Orlando: A Biography** Mrs. Dalloway	The Waves **Orlando: A Biography** The Magic Toyshop
		8	4	2	2
	3. At her death, Carter left incomplete a sequel to this Charlotte Bronte novel. Carter's sequel would've been about Adele Varens, the adopted daughter of Mr. Rochester and this novel's title character	**Jane Eyre** Villette (novel) Wise Children	**Jane Eyre** Jane Eyre (character) Edward Rochester	**Jane Eyre** Villette (novel) The Professor (novel)	**Jane Eyre** Villette (novel) The Professor (novel)
		1	1	1	1
KV-MemNN$_{in}$	1. Name this English author of novels like "The Passion of New Eve" and "Nights at the Circus", known especially for feminist reinterpretations of other works	Alamgir Hashmi **Angela Carter** Peter Russell	Illusion and Reality Sabine Huynh Janez Menart	Post- feminism Janez Menart Peter Russell	Magic realism Sabine Huynh Janez Menart
		1	6	9	9
	2. Carter wrote a libretto based on this Virginia Woolf novel, whose protagonist has affairs with Queen Elizabeth I and the princess Sasha and is mentored by Nicholas Greene while writing a long poem called "The Oak Tree"	Mrs. Dalloway Night and Day (novel) Jacob's Room	Hamza Alt code The Passion of New Eve	Mrs. Dalloway Nights at the Circus Between the Acts	Mrs. Dalloway The Waves Jacob's Room
		5	10+	10+	5
	3. At her death, Carter left incomplete a sequel to this Charlotte Bronte novel. Carter's sequel would've been about Adele Varens, the adopted daughter of Mr. Rochester and this novel's title character	**Jane Eyre** The Professor (novel) Villette (novel)	Alt code The Passion of New Eve Hamza	Shirley (novel) The Professor (novel) Villette (novel)	Shirley (novel) The Professor (novel) Villette (novel)
		1	10+	10+	10+

Table 6: Top-3 entities returned by NACER and KV-MemNN$_{in}$ baselines in combination with KEWER, BiLSTM, BERT and BERT with KEWER K-Adapter query encoders along with the rank of the correct entity for queries in the same QBLink-KG information seeking dialog. The correct answer entity is highlighted in boldface, if present in the top 3 results.

NACER with	Hits@1	R@1	Hits@10	R@10	MRR
no prec. answer	880	0.5232	1492	0.8871	0.6338
1 prec. answer	1121	0.6665	1602	0.9524	0.7575
2 prec. answers	1159	0.6891	1611	0.9578	0.7810

Table 7: Impact of the features based on the answers to preceding dialog turns on the retrieval accuracy of NACER.

top-ranked entities. Specifically, all entities top-ranked by NACER regardless of the context encoder for the first query in the dialog (Angela Carter, Sabine Huynh, Janez Menart and Peter Russell) are poets. All entities top ranked by both NACER in combination with BERT query encoder for the second query (The Waves, Orlando: A Biography and Mrs. Dalloway) and by KV-MemNN$_{in}$ in combination with BERT+KEWER (Mrs. Dalloway, The Waves and Jacob's Room) are Virginia Wolf's novels, however, NACER was more precise in ranking the correct answer. Similar observations can be made about the entities top-ranked by NACER and KV-MemNN$_{in}$ in combination with BERT. Jane Eyre, Villette, The Professor and Shirley are all Bronte's novels, however only NACER was able to correctly rank Jane Ayre as the top answer. Consistent with the results in Table 5, using a weighted mean of KEWER embeddings as the query encoder produces the most accurate results for KV-MemNN$_{in}$. The top results for this configuration are typically consistent, unlike the combination of KV-MemNN$_{in}$ with BiLSTM, but KV-MemNN$_{in}$ lacks precision. Overall ineffectiveness of the query encoder based on the aggregation of KEWER embeddings can be attributed to the fact that KEWER embeddings capture topical rather than typical similarity (e.g. Vanessa Bell is a sister of Virginia Woolf and Wise Children is a novel by Angela Carter).

7 CONCLUSION

In this paper, we introduce a novel task of CER-KG; QBLink-KG, the first benchmark for CER-KG; and NACER, a feature-based neural architecture for CER-KG. Experiments with NACER in combination with different types of query encoders reveal that neural architecture aggregating lexical and semantic matching features from the immediate KG neighborhood of candidate answer entities is a more effective solution for CER-KG than multi-hop inference, answer generation or in-context learning with LLMs.

In conclusion, we outline possible avenues for future work. First, the accuracy of NACER and the baselines is equally affected by the methods utilized for entity linking and candidate entity selection steps, even though these steps are external to NACER and the baselines. Alternative approaches to these steps may improve the reported results and warrant further investigation. No aspects of NACER and the employed methods for entity linking and candidate entity selection are specific to DBpedia, however, adapting QBLink-KG to other KGs (e.g. Wikidata) is another possible avenue.

ACKNOWLEDGMENTS

This work was supported by the National Institutes of Health under the award #1R21NR020388-01A1 and by the National Science Foundation under the award #2211897.

REFERENCES

[1] Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2016. A simple but tough-to-beat baseline for sentence embeddings. In *Proceedings of the 4th International Conference on Learning Representations (ICLR)*.

[2] Saeid Balaneshinkordan, Alexander Kotov, and Fedor Nikolaev. 2018. Attentive Neural Architecture for Ad-hoc Structured Document Retrieval. In *Proceedings of the 27th ACM International on Conference on Information and Knowledge Management (CIKM)*. 1173–1182.

[3] Antoine Bordes, Nicolas Usunier, Sumit Chopra, and Jason Weston. 2015. Large-scale simple question answering with memory networks. *arXiv preprint arXiv:1506.02075*.

[4] Denny Britz, Anna Goldie, Minh-Thang Luong, and Quoc Le. 2017. Massive Exploration of Neural Machine Translation Architectures. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 1442–1451.

[5] Shulin Cao, Jiaxin Shi, Liangming Pan, Lunyiu Nie, Yutong Xiang, Lei Hou, Juanzi Li, Bin He, and Hanwang Zhang. 2022. KQA Pro: A Dataset with Explicit Compositional Programs for Complex Question Answering over Knowledge Base. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*. 6101–6119.

[6] Shubham Chatterjee and Laura Dietz. 2022. BERT-ER: Query-specific BERT Entity Representations for Entity Ranking. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 1466–1477.

[7] Jing Chen, Chenyan Xiong, and Jamie Callan. 2016. An empirical study of learning to rank for entity search. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR)*. 737–740.

[8] Philipp Christmann, Rishiraj Saha Roy, Abdalghani Abujabal, Jyotsna Singh, and Gerhard Weikum. 2019. Look before you hop: Conversational question answering over knowledge graphs using judicious context expansion. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM)*. 729–738.

[9] Philipp Christmann, Rishiraj Saha Roy, and Gerhard Weikum. 2022. Conversational question answering on heterogeneous sources. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 144–154.

[10] Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised Learning of Universal Sentence Representations from Natural Language Inference Data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 670–680.

[11] Jeffrey Dalton, Sophie Fischer, Paul Owoicho, Filip Radlinski, Federico Rossetto, Johanne R Trippas, and Hamed Zamani. 2022. Conversational Information Seeking: Theory and Application. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 3455–3458.

[12] Jeffrey Dalton, Chenyan Xiong, Vaibhav Kumar, and Jamie Callan. 2020. CAsT-19: A dataset for conversational information seeking. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 1985–1988.

[13] Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2021. Autoregressive entity retrieval. *Proceedings of the 9th International Conference on Learning Representations (ICLR)*.

[14] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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 (NAACL-HLT)*. 4171–4186.

[15] Ahmed Elgohary, Chen Zhao, and Jordan Boyd-Graber. 2018. Dataset and baselines for sequential open-domain question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 1077–1083.

[16] Emma J Gerritse, Faegheh Hasibi, and Arjen P de Vries. 2022. Entity-aware Transformers for Entity Search. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 1455–1465.

[17] Daya Guo, Duyu Tang, Nan Duan, Ming Zhou, and Jian Yin. 2018. Dialog-to-action: Conversational question answering over a large-scale knowledge base. In *Proceedings of the 32nd Annual Conference on Neural Information Processing Systems (NeurIPS)*. 2942–2951.

[18] Nam Hai Le, Thomas Gerald, Thibault Formal, Jian-Yun Nie, Benjamin Piwowarski, and Laure Soulier. 2023. CoSPLADE: Contextualizing SPLADE for Conversational Information Retrieval. In *Proceedings of the 45th European Conference on Information Retrieval (ECIR)*. 537–552.

[19] Faegheh Hasibi, Fedor Nikolaev, Chenyan Xiong, Krisztian Balog, Svein Erik Bratsberg, Alexander Kotov, and Jamie Callan. 2017. DBpedia-entity v2: a test collection for entity search. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 1265–1268.

[20] Xiaodong He and David Golub. 2016. Character-level question answering with attention. In *Proceedings of the 2016 conference on empirical methods in natural language processing (EMNLP)*. 1598–1607.

[21] Xin Huang, Jung-Jae Kim, and Bowei Zou. 2021. Unseen Entity Handling in Complex Question Answering over Knowledge Base via Language Generation. In *Findings of the Association for Computational Linguistics: EMNLP*. 547–557.

[22] Mohit Iyyer, Wen-tau Yih, and Ming-Wei Chang. 2017. Search-based neural structured learning for sequential question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*. 1821–1831.

[23] Endri Kacupaj, Joan Plepi, Kuldeep Singh, Harsh Thakkar, Jens Lehmann, and Maria Maleshkova. 2021. Conversational Question Answering over Knowledge Graphs with Transformer and Graph Attention Networks. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*. 850–862.

[24] Magdalena Kaiser, Rishiraj Saha Roy, and Gerhard Weikum. 2021. Reinforcement learning from reformulations in conversational question answering over knowledge graphs. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 459–469.

[25] Gangwoo Kim, Hyunjae Kim, Jungsoo Park, and Jaewoo Kang. 2021. Learn to resolve conversational dependency: A consistency training framework for conversational question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*. 6130–6141.

[26] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *Proceesings of the 3rd International Conference on Learning Representations (ICLR)*.

[27] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. *Proceedings of the 26th Annual Conference on Neural Information Processing Systems (NeurIPS)* 25, 1097–1105.

[28] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. 2015. DBpedia–a large-scale, multilingual knowledge base extracted from Wikipedia. *Semantic Web* 6, 2, 167–195.

[29] Sheng-Chieh Lin, Jheng-Hong Yang, and Jimmy Lin. 2021. Contextualized Query Embeddings for Conversational Search. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 1004–1015.

[30] Xi Victoria Lin, Richard Socher, and Caiming Xiong. 2018. Multi-Hop Knowledge Graph Reasoning with Reward Shaping. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 3243–3253.

[31] Denis Lukovnikov, Asja Fischer, and Jens Lehmann. 2019. Pretrained transformers for simple question answering over knowledge graphs. In *Proceedings of the 2019 International Semantic Web Conference (ISWC)*. 470–486.

[32] Denis Lukovnikov, Asja Fischer, Jens Lehmann, and Sören Auer. 2017. Neural network-based question answering over knowledge graphs on word and character level. In *Proceedings of the 26th international conference on World Wide Web (WWW)*. 1211–1220.

[33] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press.

[34] Pierre Marion, Pawel Nowak, and Francesco Piccinno. 2021. Structured Context and High-Coverage Grammar for Conversational Question Answering over Knowledge Graphs. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 8813–8829.

[35] Alexander H. Miller, Adam Fisch, Jesse Dodge, Amir-Hossein Karimi, Antoine Bordes, and Jason Weston. 2016. Key-Value Memory Networks for Directly Reading Documents. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, (EMNLP)*. 1400–1409.

[36] Salman Mohammed, Peng Shi, and Jimmy Lin. 2018. Strong Baselines for Simple Question Answering over Knowledge Graphs with and without Neural Networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*. 291–296.

[37] Fedor Nikolaev and Alexander Kotov. 2020. Joint Word and Entity Embeddings for Entity Retrieval from a Knowledge Graph. In *Proceedings of the 42nd European Conference on Information Retrieval (ECIR)*. 141–155.

[38] Fedor Nikolaev, Alexander Kotov, and Nikita Zhiltsov. 2016. Parameterized fielded term dependence models for ad-hoc entity retrieval from knowledge graph. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR)*. 435–444.

[39] Michael Petrochuk and Luke Zettlemoyer. 2018. SimpleQuestions Nearly Solved: A New Upperbound and Baseline Approach. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 554–558.

[40] Kechen Qin, Cheng Li, Virgil Pavlu, and Javed Aslam. 2021. Improving Query Graph Generation for Complex Question Answering over Knowledge Base. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 4201–4207.

[41] Minghui Qiu, Xinjing Huang, Cen Chen, Feng Ji, Chen Qu, Wei Wei, Jun Huang, and Yin Zhang. 2021. Reinforced history backtracking for conversational question

answering. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*. 13718–13726.
[42] Chen Qu, Liu Yang, Cen Chen, Minghui Qiu, W Bruce Croft, and Mohit Iyyer. 2020. Open-retrieval conversational question answering. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval (SIGIR)*. 539–548.
[43] Chen Qu, Liu Yang, Minghui Qiu, W Bruce Croft, Yongfeng Zhang, and Mohit Iyyer. 2019. BERT with history answer embedding for conversational question answering. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval (SIGIR)*. 1133–1136.
[44] Chen Qu, Liu Yang, Minghui Qiu, Yongfeng Zhang, Cen Chen, W Bruce Croft, and Mohit Iyyer. 2019. Attentive history selection for conversational question answering. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM)*. 1391–1400.
[45] Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends® in Information Retrieval* 3, 4 (2009), 333–389.
[46] Amrita Saha, Vardaan Pahuja, Mitesh M Khapra, Karthik Sankaranarayanan, and Sarath Chandar. 2018. Complex sequential question answering: Towards learning to converse over linked question answer pairs with a knowledge graph. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI)*. 705–713.
[47] Apoorv Saxena, Aditay Tripathi, and Partha Talukdar. 2020. Improving multi-hop question answering over knowledge graphs using knowledge base embeddings. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*. 4498–4507.
[48] Tao Shen, Xiubo Geng, Tao Qin, Daya Guo, Duyu Tang, Nan Duan, Guodong Long, and Daxin Jiang. 2019. Multi-Task Learning for Conversational Question Answering over a Large-Scale Knowledge Base. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 2442–2451.
[49] Haitian Sun, Tania Bedrax-Weiss, and William Cohen. 2019. PullNet: Open Domain Question Answering with Iterative Retrieval on Knowledge Bases and Text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 2380–2390.
[50] Haitian Sun, Bhuwan Dhingra, Manzil Zaheer, Kathryn Mazaitis, Ruslan Salakhutdinov, and William Cohen. 2018. Open Domain Question Answering Using Early Fusion of Knowledge Bases and Text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 4231–4242.
[51] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971* (2023).
[52] Ferhan Ture and Oliver Jojic. 2017. No Need to Pay Attention: Simple Recurrent Neural Networks Work!. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2866–2872.
[53] Svitlana Vakulenko, Shayne Longpre, Zhucheng Tu, and Raviteja Anantha. 2021. Question rewriting for conversational question answering. In *Proceedings of the 14th ACM International Conference on Web search and Data Mining (WSDM)*. 355–363.
[54] Nikos Voskarides, Li Dan, Ren Pengjie, Kanoulas Evangelos, and Maarten de Rijke. 2020. Query resolution for conversational search with limited supervision. In *In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 921–930.
[55] Ruize Wang, Duyu Tang, Nan Duan, Zhongyu Wei, Xuanjing Huang, Guihong Cao, Daxin Jiang, Ming Zhou, et al. 2021. K-adapter: Infusing knowledge into pre-trained models with adapters. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP*. 1405–1418.
[56] Jason Weston, Sumit Chopra, and Antoine Bordes. 2015. Memory Networks. In *Proceedings of the 3rd International Conference on Learning Representations, (ICLR)*.
[57] Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. 2020. Scalable Zero-shot Entity Linking with Dense Entity Retrieval. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 6397–6407.
[58] Wenpeng Yin, Mo Yu, Bing Xiang, Bowen Zhou, and Hinrich Schütze. 2016. Simple Question Answering by Attentive Convolutional Neural Network. In *Proceedings of the 2016 International Conference on Computational Linguistics (COLING)*. 1746–1756.
[59] Nikita Zhiltsov, Alexander Kotov, and Fedor Nikolaev. 2015. Fielded sequential dependence model for ad-hoc entity retrieval in the web of data. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 253–262.

A EXTRACTING FEATURES FROM DBPEDIA SNAPSHOT

The feature vector $\bar{y_i}$ for a candidate KG entity y_i was derived from $\mathcal{T}_{\bar{i}}$, the set of KG triplets extracted from the *Mappingbased Objects* subset and the URI object triplets of the *Infobox Properties* in the DBpedia snapshot. $\texttt{ent}_e$ and $\texttt{ent}_w$ features were derived from the *Mappingbased Objects* subset and URI object triplets of the *Infobox Properties* included in $\mathcal{T}_{\bar{i}}$. Triplets from the *Mappingbased Literals* subset and literal triplets from the *Infobox Properties* were used to derive $\texttt{lit}_e$ and $\texttt{lit}_w$ features. Triplets from the *Mappingbased Objects*, *Mappingbased Literals* and *Infobox Properties* were used to derive the values of the $\texttt{pred}_e$ and $\texttt{pred}_w$ features. The categories of entities used to derive the $\texttt{cat}_e$ and $\texttt{cat}_w$ features were obtained from the *Article Categories* subset of the snapshot. Finally, all four aforementioned subsets were used to derive $\texttt{ans}_e$ and $\texttt{ans}_w$. All entity redirects were resolved using the *Transitive Redirects* subset.

B HYPERPARAMETER SETTINGS AND MODEL DESIGN CHOICES

Various hyperparameters of the proposed models and the baselines were set to the values that had been demonstrated as effective in the existing literature [4, 27]. The parameters of BM25F were trained to maximize MRR using the coordinate ascent procedure with 5 iterations, 1 restart, and the smallest parameter value increment of 0.02. In Eq. (3), ReLU was used as a non-linearity function σ, and the numbers of neurons in the first and second matching feature aggregation layers of NACER were set to 20 and 10, respectively. The dimensionality of $\mathbf{v}$ in the additive interaction function was set to 512. We considered n-grams up to size 3 and set the number of candidate entities to 400, following [32]. Following [37], the term weighting parameter λ in Eq. 2 was set to 3×10^{-4}. We used V1 configuration of InferSent[9] encoder as the implementation of BiLSTM encoder with max pooling. The pre-trained `bert-base-uncased` from the Hugging Face was used as the implementation of BERT. We fine-tuned GENRE for 10 epochs using the training split of QBLink-KG and set the beam size to 10. We compared the performance of KV-MemNN$_{\text{in}}$ and KV-MemNN$_{\text{out}}$ baselines using $H = 1, 2, 3, 4$ hops on the validation set and found out that both methods demonstrated the best performance when $H = 3$, which is the setting we used to report their results.

C TRAINING PROCEDURE

All variants of NACER and KV-MemNN were trained on the training split of QBLink-KG. To address overfitting, we utilized early stopping and saved the model parameters resulting in the smallest loss on the validation set. Adam optimizer [26] with the learning rate of 10^{-3} was used to train all models, except NACER with $f_{e\text{-dot}}$, which was trained with the learning rate 10^{-5}. KV-MemNN models were trained for 1000 epochs, while the variants of NACER were trained for a maximum of 100 epochs, except NACER with $f_{e\text{-dot}}$ and $f_{e\text{-add}}$, since we found out that these configurations required a larger number of epochs (1500) for convergence. NACER with the KEWER embeddings-based turn encoder was trained for 500 epochs. One query was used in each training iteration.

[9] https://github.com/facebookresearch/InferSent