A Framework for Knowledge-Derived Query Suggestions

Saed Rezayi[1], Nedim Lipka[2], Vishwa Vinay[2], Ryan A. Rossi[2], Franck Dernoncourt[2], Tracy H. King[2], Sheng Li[1]
[1]Department of Computer Science, University of Georgia, Athens, GA, USA
[2]Adobe Research, San Jose, CA, USA
{saedr,sheng.li}@uga.edu
{lipka,vinay,ryrossi,dernonco,tking}@adobe.com

***Abstract*—Search engines for domain-specific media collections often rely on rich metadata being available for the content items. The annotations may not be complete or rich enough to support an adequate retrieval effectiveness. As a result, some search queries receive only a small result set (low recall) and others might suffer from reduced relevance (low precision). To alleviate this, we present a framework that exploits external knowledge to provide entity-oriented reformulation suggestions for queries that contain entities. We propose that queries be added as *surrogate nodes* to an external Knowledge Graph (KG) via the use of state-of-the-art entity linking algorithms. Embedding methods are invoked on the augmented graph, which contains additional edges between surrogate nodes and KG entities. We introduce a new evaluation setting to evaluate the quality of these embeddings. Experimental results on seven datasets confirm the effectiveness of the approach.**

***Index Terms*—Query Suggestion, Knowledge Graph, Graph Embedding, Link Prediction, Entity Linking**

I. Introduction

Knowledge Graphs (KGs) organize information and objects into a graph structure in which entities (i.e., nodes) can be traversed through relations (i.e., edges), and every traversal represents a fact which consists of two entities (head and tail) and a relation between them (i.e., a `(head, relation, tail)` triple). In this manner, a KG is a formal representation of abstract concepts and real world objects. For example the word "bass" can refer to a type of fish or describe tones of low in music. However, they are two different entities in the KG and have different local structure around them which might help disambiguate different senses of the word. The construction of a knowledge graph is a well-developed research area in its own right, and the availability of large and general purpose KGs have fuelled their popularity across a range of application settings [1].

Knowledge graphs are sparse data structures, meaning that the number of facts per entity is very small. For example, one of the largest real world knowledge graphs, Freebase, has a fact-to-entity ratio of 16 [2]. The graph structure of a KG enables reasoning over the individual *facts*. One such task is that of Link Prediction (LP) [3]–[5]—inferring new relations among entities given a snapshot of a graph. LP has many applications from friendship suggestion in social networks [6]–[8] to predicting associations between molecules in biological networks [9].

Fig. 1: An example illustrating the proposed pipeline. In this scenario, the search engine (Adobe Stock) yields small number of results for the query "Spider-Man 3". Our model suggests a new query based on an EL/LP pipeline. The suggested query, "Marvel Comics", provides search results with higher recall.

In this paper, we consider the problem of *query suggestions* within an information retrieval setting and frame it as an application of link prediction. Here, the user has provided the search engine with a short keyword query. The user has examined the returned results and our objective is to provide reformulation assistance so that the refined query is more likely to return relevant results ranked high. Typical algorithms for this task rely on historical data, where associations between queries and their constituent words can be mined, which might work for common queries for which there is plentiful data.

We propose a knowledge-aware query suggestion method that leverages an existing KG to surface entities related to those present in the query. In an entity-oriented search, we expect to rank related entities (nodes in the KG) using a link prediction model where the head is an entity present in the query and the tail entities are candidates from the KG. This ranking can further be restricted to selected relation types. This process requires an Entity Linking (EL) method that allows the linking of mentions in the query to the entities of the underlying knowledge graph.

Despite advances in entity linking methods, there are inherent problems in the use of these algorithms in the current setting. Search queries tend to be short keyword based phrases, allowing little or no context for the EL algorithm to leverage. Given the earlier example, it is almost impossible to distinguish between different instances of "bass" if there is no context around them. As a result there is a significant chance of error in the entity linking which can propagate through the link prediction phase and produce bad candidates. Knowledge graphs can be useful in addressing this challenge. The information within the graph structure is leveraged by KG embedding methods [10] that we expect would be useful in disambiguating alternate interpretations of words/entities in the query.

In this paper, we propose a framework that leverages state-of-the-art entity linking and KG embedding methods to help identify query reformulation candidates. We introduce the notion of *surrogate nodes* which are nodes corresponding to queries added into the KG. For each mention in the query (an entity), we add a new node to the KG and connect it to its related entities (output of the EL algorithm) already present in the KG. The inclusion of surrogate nodes improves the semantics of the KG by adding different senses of a query. It also offers a mechanism to grow/refine the KG based on emerging/evolving set of entities present in user search queries.

We then perform LP on the enhanced KG (based on similarity in embedding space, see Algorithm 1), thereby providing related entities that are interpreted as query suggestions. We evaluate this new problem setting, an EL-LP pipeline, by the construction of a new benchmark and design of relevant metrics to measure the performance of our approach against the baseline. Additionally, we undertake a query suggestion study which demonstrates how providing additional knowledge to the KG can disambiguate the user intent. In Figure 1, the user may not know that in order to get information for "Spider-Man 3", the keyword "Marvel Comics" can be used. In this sense our goal is to assist the user more effectively benefit from the search experience.

We summarize our contributions as follows:

1) We propose the framing of knowledge-aware query suggestion as a link prediction task. The key idea of our framework is to add surrogate nodes which represent query entities to an existing knowledge graph.
2) We propose a framework that tolerates imperfect entity linking but still identifies relevant related entities to those present in the query.
3) We conduct a thorough evaluation of this new problem setting, including a range of metrics, to illustrate the utility of our method.
4) We conduct a case study to qualitatively demonstrate the effectiveness of our approach.

The rest of the paper is organized as follows: We first survey the related works in the area of link prediction and query suggestion/generation and explain how our work is different from previous studies. Next, in Section III we provide the required definitions, introduce our problem and the formal framework for our problem (contribution 1). In Section IV, we present our new evaluation setting and the metrics we use to assess our work (contribution 2). We examine our model in Section V against several benchmark datasets and carry out analyses to show the effectiveness of our method. In Section VI, we provide a case study and conclude our work along with suggested directions for future work in Section VII.

II. Related Work

A. *Query Reformulation*

The problem of query suggestion has been widely studied recently [11]–[14]. Most of the work on query suggestion leverages user log data such as their clicks [11], [12], [14]. Most of these works focus on using only user log data [] as opposed to knowledge graph embeddings as done in our work. In contrast, our work focuses on leveraging an external knowledge graph embedding. and studies for query reformulation. Recently, there have been a few works that leverage knowledge graph embeddings (KGE) for tasks related to query suggestion [15]–[17]. In particular, Recent work has also studied the problem of suggesting graph-queries for exploring knowledge graphs [18]. However, these works all solve a different problem than the one we study in this paper.

While there are some recent work on query reformulation [19]–[23], none of them study the same problem as the one we study in this work. In particular, a recent paper by Hirsch et al. [20] conducts a large-scale study investigating query reformulations by users. Another work by Wang et al. [19] proposed a reinforcement learning approach that uses a seq2seq model trained with user query log data to perform query reformulations. None of these works leverage a knowledge graph or its embedding to improve query reformulation task.

B. *Link Prediction*

Link Prediction has been extensively studied in social networks [8], web graphs [24], biological networks [9], information networks [25], and KG [26]. Methods have been

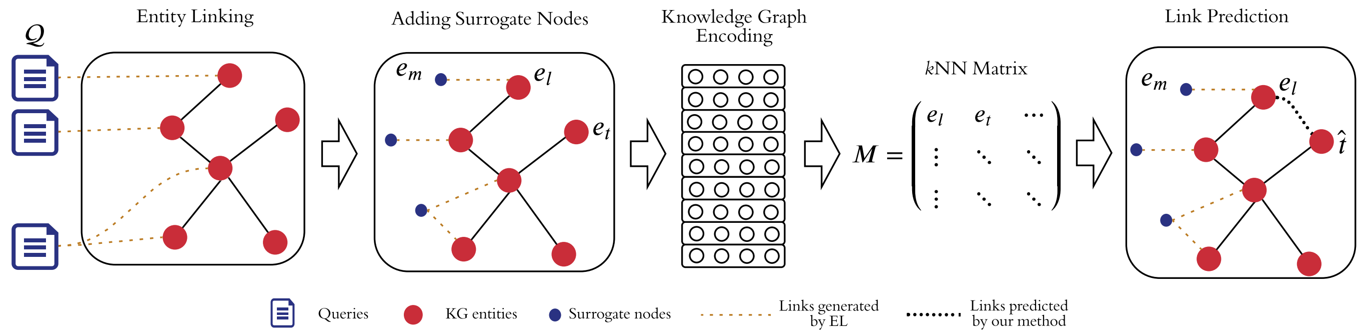

Fig. 2: Using existing EL algorithms, we link the mentions (m) in the input queries to KG entities: linked entities (e_l). Then we create a node per mention: surrogate nodes (e_m). Next we obtain node embeddings and use the vector representations to calculate the kNN matrix (M). In this example e_t is the nearest neighbor to e_l so we call it the predicted tail ($\hat{t}$). $\hat{t}$ is used as a suggested query. The pair $\langle m, \hat{t}\rangle$ is then evaluated.

proposed for predicting links in different types of graphs including bipartite graphs [27], homogeneous graphs [5], and knowledge graphs [4], [28]. Methods for link prediction are all essentially based on either the notion of proximity in the graph [29]–[33] or the notion of structural similarity/roles [34]. There have been some work on using motifs for link prediction [5], [35]. Complex link prediction methods have been developed recently that leverage embeddings [28] derived from graph autoencoders [36], [37], graph neural networks [30], spectral methods [26], [31], among many others [38]–[40]. Other work has focused on the evaluation of different link prediction methods [41]. While most work has focused on transductive (within-network) link prediction [30], [36], there are some recent inductive (across-network) link prediction methods [38]–[40].

More recently, there has been a lot of work on link prediction in knowledge graphs [4], [29], [42], [43]. Most link prediction methods for knowledge graph are based on proximity/distance in the graph, and leverage paths [29], tensor factorization [44], random walks [45], and other local proximity-based mechanisms [43]. There have also been some recent work that enriches the graph to improve link prediction using multilingual textual descriptions [25]. The proposed framework in this work is agnostic to the link prediction method, and can naturally leverage any state-of-the-art approach. Unlike previous methods, our proposed framework can choose from a variety of EL and LP algorithms in a plug-in manner. In addition, our method can be used when user data is sparse. It searches for queries that exist in the underlying KG, and hence does not need behavioral and retrieval data.

III. Proposed Model

In this section we first define the terminology used in this paper, formally, then describe the problem we are trying to solve, and finally explain how we address this problem.

A. Preliminary Definition

Knowledge Graph: Let $\mathcal{G} = (\mathcal{E}, \mathcal{R}, \mathcal{T})$ be the knowledge graph, where $\mathcal{E}$ is a set of textual entities, $\mathcal{R}$ is a set of textual relations, and $\mathcal{T}$ is a set of triplets in the form of (h, r, t), where $h, t \in \mathcal{E}$ are *head* and *tail* entities and $r \in \mathcal{R}$ is a relation between two entities.

Entity Linking: Let $Q = \{q_1, \cdots, q_n\}$ be a set of textual queries. Every query q is defined as a sequence of words $q = (w_1 \cdots w_v)$. Every subsequence of words in q that represents an entity e is called entity mention and denoted by m if $e \in \mathcal{E}$. The process of mapping mention m to entity e is called Entity Linking, EL : $m \mapsto e$.

B. Problem Definition

Knowledge-Derived Query Suggestion: The standard query suggestion task is considered as a ranking problem where a set of candidate suggested queries are ranked and provided to the user given an initial query, q, and a scoring function. Existing methods take q_i and predict the next query $P(q_{i+1}|q_i)$ using a sequence-to-sequence model. These methods cannot handle out-of-vocabulary words and have unpredictable behavior for rare combinations. We redefine the task of query suggestion such that it would not require user data (historical or behavioral). The central idea is that KG entities are meaningful suggestions because they are curated objects whose coverage are not limited based on word popularity. To incorporate the KG we use an entity linking algorithm which assigns a set of KG entities to the query and use these linked entities to find relevant suggestions for the initial query.

In summary, given the above setting, a knowledge graph $\mathcal{G} = (\mathcal{E}, \mathcal{R}, \mathcal{T})$, and a query $q \in Q$, the goal is to return a ranked list of *relevant* entities, $\langle e_1, e_2, \cdots\rangle$, where $e_i \in \mathcal{E}$ and relevance is inferred based on the distance in the embedding space.

C. Proposed Model

Our model incorporates a state-of-the-art EL algorithm which allows us to map queries to entities in the underlying KG. Knowledge-derived query suggestion relies on properly identifying the entities in the query. With a perfect EL system, we can annotate the mentions in the query with known entities and the task is concluded. However, since queries are short with little or no context, EL algorithms fall short in this disambiguation task. This makes the task of query reformulation challenging and motivates our method. In what follows we explain how imperfect EL can address these shortcomings.

First, we employ existing EL algorithms to map mentions in queries to multiple KG entities. We consider several linked entities because the EL method may make incorrect predictions and considering multiple entities improves the recall of our model while maintaining good precision. Further, the output of most entity linking methods are accompanied by confidence scores, and we can use these scores to weight edges when we later connect them to the knowledge graph entities. Next, for each mention m in the query set, we add a new node (i.e., an entity) e_m to the KG, which we refer to as a surrogate node. We connect surrogate nodes to linked entities if they are present in the KG, thus the new links have the form of $\langle e_m, e_l \rangle$, where e_l is the linked entity. Incorporating surrogate nodes into the KG changes the structure of the underlying KG. These textual entities introduce new semantics which we exploit in our similarity-based link prediction module.

Once we construct the augmented KG, we use a KG embedding algorithm to compute low dimensional embeddings for its entities. Given the vector representation of an entity, we propose a LP model which ranks entities based on their relevance to the surrogate nodes. We define LP formally as follows: given a head entity e_l, the goal is to infer a tail entity $\hat{t}$ that completes a link $\langle e_l, \hat{t} \rangle$.

$$\hat{t} = \underset{e_i \in \mathcal{E}}{\operatorname{argmax}} f(\mathbf{e_l}, \mathbf{e_i}) \tag{1}$$

In this equation lower case notations in bold refer to embeddings and $f(.)$ is a score function that minimizes the distance between the two entities.

Given the embedding of a linked entity, $\mathbf{e_l} \in \mathbb{R}^d$ and the set of embeddings $\{\mathbf{e_i}\}$ where $0 < i < |\mathcal{E}|$ and $\mathbf{e_i} \in \mathbb{R}^d$, we search for top k most similar entities in the embedding space.

$$M = \underset{0 < i < |\mathcal{E}|}{k\text{-argmin}} \|\mathbf{e_l} - \mathbf{e_i}\| \tag{2}$$

In other words, for each surrogate node, e_m, we find k nearest neighbors (kNN) of its linked entities and predict links in the form of $(e_m, \hat{t})$, where $\hat{t}$ is an entity belonging to kNN of e_l. This process is presented in Figure 2 and Algorithm 1.

Algorithm 1 Link Prediction Process

Input: KG $\mathcal{G}$ and the set of queries $\mathcal{Q}$
Output: Predicted links L
for each $q \in Q$ **do**
  $m \leftarrow$ mention in q
  $S[q] = \{e_m | \; q : m \mapsto e_l \in \mathcal{E}\}$
  for each $\langle e_m \rangle \in S[q]$ **do**
    Add (e_m, e_l) to $\mathcal{G}$
  end for
end for
Calculate embeddings for all the nodes in enhanced $\mathcal{G}$
Calculate M which is the kNN matrix for all of the embedding vectors.
$L = \{\}$ *# predicted links/tails per query*
for each $q \in Q$ **do**
  for each $e_m \in S[q]$ **do**
    for each $\hat{t} \in M[e_m]$ **do**
      $L[q].append(\langle m, \hat{t} \rangle)$
    end for
  end for
end for

IV. Evaluation

One approach to assess the quality of the outcome of a search engine is to measure how satisfied the users are with the results, and the user satisfaction is quantified by several methods in the Information Retrieval community, such as relevance of the results to the query or quantifying the click information, etc.

Evaluating a system where ground truth information regarding the relevance of the returned documents to the target query is available is standard. However, in our problem setting we have a set of queries with unsatisfactory search results and the ground truth is not obtainable. Furthermore, the link prediction algorithm provides a set of entities as suggestions for reformulating the original query. Since our approach adds surrogate nodes that are not present in the ground truth, it is not possible to use the standard link prediction setting to evaluate our framework. Hence, we propose a rank-based evaluation technique that measures how well the distance metric ranks the entities.

A. Rank-based Evaluation

A surrogate node is a query in the KG which is connected to n entities obtained by one or multiple EL algorithms, and we predict k other entities per linked entity. Thus we have at most $n \times k$ predicted entities. To sort these entities based on importance we consider their euclidean distance from their associated linked entity in the embedding space. To evaluate this sorted list of predicted links we count how many of them are present in the KG if we consider the top 10 and 50 linked entities. We normalize these numbers by the total number of predicted links to obtain Hits@10 and Hits@50.

$$Hits@k = \sum_{i=1}^{|L|} 1 \text{ if } rank_{(h,r,t)_i} \leq k$$

Additionally, we borrow two other metrics from information retrieval community: mean Average Precision (mAP) which measures the percentage of relevant suggested entities, and Normalized Discounted Cumulative Gain (NDCG) to give more weight to highly relevant suggested entities compared to moderately relevant ones.

Given a ranked list of predicted links per query we mark them as relevant if they exist in the KG and calculate AP:

$$\text{AP} = \frac{1}{n_r} \sum_{i=1}^{n} (P(i) \times \text{rel}(k))$$

where n_r is the number of relevant links, $\text{rel}(k) \in \{0, 1\}$ indicates if the link is relevant or not, and $P(i)$ is the precision at i in the ranked list. Once we obtain AP for each query we can average across all queries to find MAP:

$$\text{MAP} = \frac{1}{N}\sum_{q=1}^{N} \text{AP(q)}$$

where N is the number of all queries.

$$NDCG = \frac{DCG}{iDCG}$$

where *DCG* is Discounted Cumulative Gain which calculates the sum of all the relevance scores in a suggested query set and *iDCG* is the same measure for the sorted set of suggested queries:

$$DCG = \sum_{i=1}^{N} \frac{rel_i}{\log_2(i+1)}$$

where logarithm in the denominator takes the position of the suggested query into consideration.

B. Similarity-based Evaluation

In addition to rank-based metrics we can capture the relatedness of the suggested queries to the intended query by measuring the similarity between the two. The similarity can be defined in text space or in the embedding space. Hasibi et al., [46] proposed lexical similarity as a feature to measure relevance. For this metric we use Jaro edit distance to capture spelling mismatches.

$$\text{sim}_{lex} = \max_{\hat{t}\in L}(1 - dist(\hat{t}, q))$$

where $\hat{t}$ is the predicted tail entity and q is the target query.

To further measure the quality of suggested queries and following the idea proposed by [47], we use a word embedding algorithm to obtain the vector representations of the target query and the suggested query and calculate the cosine similarity of the two vectors and report it as a performance measure. This is only possible if the click information of the users is available. To this end we choose a dataset which provides session based query log information from AOL (more details in Section V). Given two embedding vectors, $\mathbf{e}_1$ and $\mathbf{e}_2$, cosine similarity is defined as follows:

$$\text{sim}_{emb} = \frac{\mathbf{e}_1 \cdot \mathbf{e}_2}{\|\mathbf{e}_1\|\|\mathbf{e}_2\|}$$

C. Baseline

In this new problem setting, we define a strong baseline. We compare our model with the case where we predict links in the form of $\langle \text{EL}(m), \hat{t}\rangle$, where $\text{EL}(m)$ is the top linked entity for the mention m. Additionally, we establish a *gold standard* upper bound in which we know the true entity for each mention and predict links in the form of $\langle s, \hat{t}\rangle$ where s is the true linked entity for mention m which means we have an error-free EL oracle. Please note that we cannot compare our model with conventional methods because our problem setting is different. In fact our model can be built on top of any query suggestion framework and improve their results (as we will show in section VI).

TABLE I: Statistics of the benchmark datasets. We used BLINK [48] to obtain accuracy measures in the last column which is a large scale entity linking algorithm.

Dataset	#documents	#mentions	EL accuracy
AIDA-YAGO2	946	18,448	80.27%
ACE2004	36	257	86.89%
AQUAINT	50	727	85.88%
MSNBC	20	656	85.09%
WNED-CWEB	320	11,154	68.25%
WNED-WIKI	320	6,821	80.67%
Yahoo!	980	2,114	60.07%

V. Experiments

This section describes the datasets and evaluates our model based on the evaluation setting explained in previous section.

A. Datasets

We consider several benchmark datasets to evaluate our framework. These datasets are designed for tasks other than query suggestion, such as entity linking, named entity recognition, etc.

AIDA-YAGO2: This dataset consists of hand annotated Reuters news articles and contains assignments of entities to the mentions of named entities [49]. Mentions in this dataset can be mapped to YAGO, Wikidata, and Freebase entities.

ACE2004: ACE is a subset of ACE co-reference dataset annotated by Amazon's MTurk [50], Originally, The Automatic Content Extraction (ACE) program presented this dataset for information extraction-related tasks such as entity recognition, relation recognition, and event extraction [51].

AQUAINT: This dataset is a news corpus consists of text data in English, drawn from three sources: the Xinhua News Service, the New York Times News Service, and the Associated Press [52]. It has been used for evaluating entity disambiguation and entity linking tasks [48], [53].

MSNBC: This dataset contains top two stories in the ten MSNBC news categories [54].

WNED-CWEB and **WNED-WIKI**: These datasets were introduced in [55] for the task of entity disambiguation and entity linking. WNED-CWEB and WNED-WIKI are obtained from large web corpora, namely Clueweb12 [56], and Wikipedia, respectively.

Moreover, we consider a dataset from information retrieval: Yahoo data search query log is part of the Yahoo Webscope program [57] that contains queries obtained from Yahoo web search. The basic statistics of these datasets are presented in Table I.

For all the listed datasets in Table I, true entity and the context around the mention are provided. True entity is the label that links mentions to Wikidata entities [58], and we use

TABLE II: Rank based results for all datasets. The entity linking for Yahoo dataset is missing as the ground truth is not available for this dataset/task.

		Baseline				Our Approach				Gold Standard			
Dataset	EL Accuracy	Hits@10	Hits@50	mAP	NDCG	Hits@10	Hits@50	mAP	NDCG	Hits@10	Hits@50	mAP	NDCG
ACE2004	84.43	1.81	12.14	18.78	14.55	2.17	13.41	22.40	17.20	14.50	72.46	97.00	56.88
AIDA-YAGO2	79.51	1.65	7.16	33.72	44.93	4.21	10.60	61.78	54.34	11.25	38.25	73.36	63.79
AQUAINT	86.62	1.45	3.26	15.27	16.77	5.47	13.14	88.23	34.41	10.87	16.30	92.43	62.98
MSNBC	84.28	1.00	6.52	15.37	19.64	10.94	22.61	57.32	42.55	13.91	51.09	93.65	53.95
WNED-CWEB	67.47	3.43	11.38	43.84	43.20	6.47	16.29	68.10	59.01	10.40	34.58	77.48	61.12
WNED-WIKI	79.76	2.66	13.45	38.17	26.82	5.06	16.12	66.50	40.40	11.42	42.46	85.35	54.45
Yahoo!	-	2.28	22.57	32.14	16.71	14.49	25.88	92.22	47.90	23.81	55.69	97.66	57.80

these labels to measure the performance of the entity linking and investigate how this performance affects the performance of our final model. State-of-the-art entity linking accuracy on these datasets is also provided in Table I. For these datasets we report rank-based metrics, as we can establish baseline and the upper bound as explained in previous section.

To evaluate our work using similarity-based metrics, we use **AOL** dataset which is a collection of 20M web queries collected from 650k users from 01 March, 2006 to 31 May, 2006. Time stamp data and click through information are available and if the user clicked on a search result, the rank of the item on which they clicked is also listed [59]. The presence of click information enables us to perform similarity-based evaluation. We split queries into sessions, every 30 minutes is considered a session [60]. The queries in each session form a context for the target query (the last query in a session is the target query if the user clicks on the search result). We perform basic preprocessing (e.g., removing punctuation and converting to lower case), and select 10,000 sessions at random for the experiment.

For our case study, we use a set of queries issued to Adobe Stock. This dataset only provides query information, and the ground truth and click information are not provided. Hence none of the evaluations discussed in Section IV could be performed on this dataset. As a result, we picked 50 queries and asked 20 annotators to annotate the search result from two search engines (Google Image and Adobe Stock). Given the annotated pairs of queries and search results, we can study the quality of suggested queries.

As an underlying KG we use **FB15k-237** which contains 15K entities and 237 relations [61]. It is based on Freebase and is a subset of FB15k [62] where redundant relations have been removed. This KG has been widely used in the literature [63]–[65] for the task of graph completion. We use it as an external source of knowledge to suggest alternative queries. We employ the existing mapping between the entities of FB15k-237, denoted by *mid*s, and Wikidata entities. This mapping is required as the EL algorithm we use maps the mentions to Wikidata entities. Besides, we construct a reduced version of the FB15k-237 to only consider the intersection of the linked entities of the query log mentions and the knowledge graph entities.

B. Rank-based results

We compare the predicted links from our approach with links predicted by the baseline and gold standard. We report Hits@10, Hits@50, mAP and NDCG. Table II presents the performance measures for this analysis. Our approach lies between the performance of the baseline and gold standard, outperforming the baseline across all datasets.

Consider the case when the accuracy of EL is low (e.g., 67.47% for WNED-CWEB). In this case, we observe a considerable increase in performance over the baseline—a 100% increase in Hits@10. This is likely due the top linked entity being identical to the correct entity and is consistent with our underlying hypothesis. This suggests that our approach is most helpful when EL fails. We hypothesize that short queries are most likely to encounter low EL accuracy, and these are the focus of our case study (Section VI).

We further study the cumulative precision with regard to the distance used to rank the suggested queries. To do this, we first sort the predicted links based on distance, then calculate precision for each threshold considering all links with distance between 0 and that threshold. This is illustrated in Figure 3 for WNED-CWEB dataset. In Figure 3, precision at threshold 1 is the precision when considering all the predicted links, i.e., nothing is filtered out based on distance. Ideally, we should observe a sigmoid shape with the center at 0.5 (similar to the gold standard). Our model follows a similar trend. The precision of the baseline varies only up to about 0.4 and is not monotonic.

C. Similarity-based results

We report similarity-based results on AOL dataset. For this task we perform entity linking on the context of each session and predict tails using baseline and our approach. As described in Section IV we use lexical and semantic similarity metrics to measure the relevance. For lexical similarity we obtain 59.3% similarity between the predicted tail and the target query (we retrieve this based on click information that is available in AOL dataset) for our approach, while achieve 48.2% similarity when using the baseline.

We then compute the embedding of the top predicted tail in each approach (using a pretrained BERT model [66]), also compute the embedding of the target query and calculate the cosine similarity between the two embeddings per session. We

TABLE III: (left) Results of case study broken down by configuration. Shown are percentage of participants rating result pages to be relevant and Krippendorff's α. (right) Results of filtered queries with lower rating variance in order to achieve a Krippendorff's $\alpha > 0.7$

	Baseline		Our Approach		
	(exclusive)	(expanded)	(exclusive)	(expanded)	
Google Image	52.5% (.408)	54.6% (.210)	60.0% (.492)	60.7% (.534)	**60.4%** (.414)
Adobe Stock	40.0% (.289)	53.8% (.629)	56.6% (.741)	62.5% (.561)	**51.6%** (.623)
	51.1% (.484)		**62.0%** (.619)		

	Baseline	Our Approach
Google Image	**74.2%** (.337)	**80.5%** (.370)
Adobe Stock	**40.0%** (.698)	**75.4%** (.756)

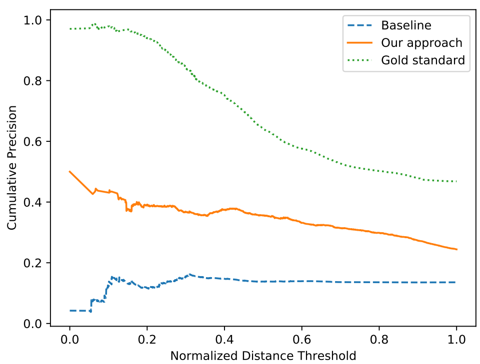

Fig. 3: Cumulative precision at distance-based thresholds.

obtain 90.6% for cosine similarity averaged across all sessions when using our approach compared to 86.3% for baseline.

VI. Case Study

We investigate the quality of the suggested query candidates in a case study based on 210 queries and 50 MTurk participants. Participants judged the relevance of results obtained from Google Images and from Adobe Stock. We collected 10 ratings per query, search engine, and approach triple. We randomly sampled from real-world queries issued on Adobe Stock where the results were reported as unsatisfactory by users. We consider basic variants of using suggested entities as search queries that are produced by the baseline and our method: (*exclusive*) uses exclusively suggested entities as query, and, (*expanded*) uses the original query in conjunction with the suggested entities.

Overall—with a rather low overall Krippendorff's $\alpha = .55$—employing Knowledge-Derived Query Suggestion techniques such as the introduced baseline or our proposed approach lead on average to 57% relevant retrieval results. Our approach is outperforming the baseline in all experiment configurations. While Google's image search produces more relevant results compared to Adobe Stock due to its larger repository, our approach has the largest impact on Adobe Stock relative to the baseline. Table III (left) shows a breakdown of the experiment configurations including the corresponding Krippendorff's α scores. Our approach tends to produce larger inter-rater reliability.

A Krippendorff's α of .7 which allows tentative conclusions according to [67] can be achieved with the removal of 26% of queries that have the highest rating variance. Independent of the setup, in 64.8% participants found the results relevant. Table III (right) shows the corresponding results for several experiment configurations.

VII. Conclusion and Future Work

This paper proposed a query suggestion framework that exploits an external source of knowledge. Using state-of-the-art entity linking, we added queries represented as surrogate nodes to an external KG and showed how the inclusion of different senses of a query boosts the retrieval effectiveness. We devised a link prediction mechanism that returns a ranked list of queries similar to the linked entities and we proposed metrics to evaluate the list of suggested queries. We performed extensive experiments on seven benchmark datasets to show the superiority of our model over the baseline. We also carried out a case study to assess the qualitative effectiveness of our model.

As a future direction, we hope to focus on the problem of producing improved query suggestions. Currently our model suggests an alternative query but a hierarchical encoding scheme can enable users to have the ability to choose from generalization, i.e., integrating suggested entities into a higher-level entity, e.g., *student* and *faculty* into *university member*, or specialization, i.e., identifying sub-groups of the target query, e.g., *employee* to *developer* and *engineer*.

References

[1] S. Ji, S. Pan, E. Cambria, P. Marttinen, and S. Y. Philip, "A survey on knowledge graphs: Representation, acquisition, and applications," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[2] J. Pujara, E. Augustine, and L. Getoor, "Sparsity and noise: Where knowledge graph embeddings fall short," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. ACL, 2017, pp. 1751–1756.

[3] L. Duan, S. Ma, C. Aggarwal, T. Ma, and J. Huai, "An ensemble approach to link prediction," *Transactions on Knowledge and Data Engineering*, vol. 29, no. 11, pp. 2402–2416, 2017.

[4] S. M. Kazemi and D. Poole, "Simple embedding for link prediction in knowledge graphs," in *Advances in neural information processing systems*, 2018, pp. 4284–4295.

[5] A. R. Benson, R. Abebe, M. T. Schaub, A. Jadbabaie, and J. Kleinberg, "Simplicial closure and higher-order link prediction," *Proceedings of the National Academy of Sciences*, vol. 115, no. 48, pp. E11 221–E11 230, 2018.

[6] M. Al Hasan and M. J. Zaki, "A survey of link prediction in social networks," in *Social network data analytics*. Springer, 2011, pp. 243–275.

[7] M. A. Ahmad, Z. Borbora, J. Srivastava, and N. Contractor, "Link prediction across multiple social networks," in *IEEE International Conference on Data Mining Workshops*, 2010, pp. 911–918.

[8] L. M. Aiello, A. Barrat, R. Schifanella, C. Cattuto, B. Markines, and F. Menczer, "Friendship prediction and homophily in social media," *ACM Transactions on the Web (TWEB)*, vol. 6, no. 2, pp. 1–33, 2012.

[9] J. Jiang, L.-P. Liu, and S. Hassoun, "Learning graph representations of biochemical networks and its application to enzymatic link prediction," 2020.

[10] Q. Wang, Z. Mao, B. Wang, and L. Guo, "Knowledge graph embedding: A survey of approaches and applications," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 12, pp. 2724–2743, 2017.

[11] L. Li, H. Deng, A. Dong, Y. Chang, R. Baeza-Yates, and H. Zha, "Exploring query auto-completion and click logs for contextual-aware web search and query suggestion," in *WWW*, 2017, pp. 539–548.

[12] J. Zhong, W. Guo, H. Gao, and B. Long, "Personalized query suggestions," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 1645–1648.

[13] J.-Y. Jiang and W. Wang, "Rin: reformulation inference network for context-aware query suggestion," in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, 2018, pp. 197–206.

[14] R. Li, L. Li, X. Wu, Y. Zhou, and W. Wang, "Click feedback-aware query recommendation using adversarial examples," in *The World Wide Web Conference*, 2019, pp. 2978–2984.

[15] C. Rosset, C. Xiong, X. Song, D. Campos, N. Craswell, S. Tiwary, and P. Bennett, "Leading conversational search by suggesting useful questions," in *Proceedings of The Web Conference*, 2020, pp. 1160–1170.

[16] C. Shi, J. Ding, X. Cao, L. Hu, B. Wu, and X. Li, "Entity set expansion in knowledge graph: a heterogeneous information network perspective," *Frontiers of Computer Science*, vol. 15, no. 1, pp. 1–12, 2020.

[17] X. Huang, J. Zhang, D. Li, and P. Li, "Knowledge graph embedding based question answering," in *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, 2019, pp. 105–113.

[18] M. Lissandrini, D. Mottin, T. Palpanas, and Y. Velegrakis, "Graph-query suggestions for knowledge graph exploration," in *Proceedings of The Web Conference 2020*. Association for Computing Machinery, 2020, p. 2549–2555.

[19] X. Wang, C. Macdonald, and I. Ounis, "Deep reinforced query reformulation for information retrieval," 2020.

[20] S. Hirsch, I. Guy, A. Nus, A. Dagan, and O. Kurland, "Query reformulation in e-commerce search," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 1319–1328.

[21] S.-C. Lin, J.-H. Yang, R. Nogueira, M.-F. Tsai, C.-J. Wang, and J. Lin, "Query reformulation using query history for passage retrieval in conversational search," 2020.

[22] M. Das, J. Li, E. Fosler-Lussier, S. Lin, S. Rust, Y. Huang, and R. Ramnath, "Sequence-to-set semantic tagging for complex query reformulation and automated text categorization in biomedical ir using self-attention," in *Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing*, 2020, pp. 14–27.

[23] G. Verma, V. Vinay, S. Bansal, S. Oberoi, M. Sharma, and P. Gupta, "Using image captions and multitask learning for recommending query reformulations," in *European Conference on Information Retrieval*. Springer, 2020, pp. 681–696.

[24] M. Zhang, Z. Cui, S. Jiang, and Y. Chen, "Beyond link prediction: Predicting hyperlinks in adjacency space." in *AAAI*, 2018, p. 6.

[25] G. A. Gesese, M. Alam, and H. Sack, "Semantic entity enrichment by leveraging multilingual descriptions for link prediction," 2020.

[26] B. Pachev and B. Webb, "Fast link prediction for large networks using spectral embedding," *Journal of Complex Networks*, vol. 6, no. 1, pp. 79–94, 2018.

[27] J. Kunegis, E. W. De Luca, and S. Albayrak, "The link prediction problem in bipartite networks," in *International Conference on Information Processing and Management of Uncertainty in Knowledge-based Systems*. Springer, 2010, pp. 380–389.

[28] Y. Tay, A. T. Luu, and S. C. Hui, "Non-parametric estimation of multiple embeddings for link prediction on dynamic knowledge graphs," in *AAAI*, 2017, pp. 1243–1249.

[29] M. Zhang, Q. Wang, W. Xu, W. Li, and S. Sun, "Discriminative path-based knowledge graph embedding for precise link prediction," in *European Conference on Information Retrieval*. Springer, 2018, pp. 276–288.

[30] M. Zhang and Y. Chen, "Link prediction based on graph neural networks," in *Advances in Neural Information Processing Systems*, 2018, pp. 5165–5175.

[31] C. Sharma, J. Chauhan, and M. Kaul, "Learning representations using spectral-biased random walks on graphs," 2020.

[32] J. Shao, Z. Zhang, Z. Yu, J. Wang, Y. Zhao, and Q. Yang, "Community detection and link prediction via cluster-driven low-rank matrix completion," in *IJCAI*, 2019, pp. 3382–3388.

[33] A. Kumar, S. Mishra, S. S. Singh, K. Singh, and B. Biswas, "Link prediction in complex networks based on significance of higher-order path index (shopi)," *Physica A: Statistical Mechanics and its Applications*, vol. 545, p. 123790, 2020.

[34] N. Ahmed, R. A. Rossi, J. Lee, T. Willke, R. Zhou, X. Kong, and H. Eldardiry, "Role-based graph embeddings," *IEEE Transactions on Knowledge and Data Engineering*, no. 11, 2020.

[35] R. A. Rossi, A. Rao, S. Kim, E. Koh, and N. K. Ahmed, "From closing triangles to closing higher-order motifs," in *Proceedings of The Web Conference (WWW)*, 2020.

[36] G. Salha, S. Limnios, R. Hennequin, V.-A. Tran, and M. Vazirgiannis, "Gravity-inspired graph autoencoders for directed link prediction," in *CIKM*, 2019, pp. 589–598.

[37] G. Salha, R. Hennequin, and M. Vazirgiannis, "Simple and effective graph autoencoders with one-hop linear models," 2020.

[38] R. A. Rossi, R. Zhou, and N. K. Ahmed, "Deep inductive network representation learning," in *WWW*, 2018, pp. 953–960.

[39] Y. Hao, X. Cao, Y. Fang, X. Xie, and S. Wang, "Inductive link prediction for nodes having only attribute information," *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, vol. 3, no. 4, p. 5, 2020.

[40] D. Daza, M. Cochez, and P. Groth, "Inductive entity representations from text via link prediction," 2020.

[41] Y. Yang, R. N. Lichtenwalter, and N. V. Chawla, "Evaluating link prediction methods," *Knowledge and Information Systems*, vol. 45, no. 3, pp. 751–782, 2015.

[42] A. Rossi, D. Firmani, A. Matinata, P. Merialdo, and D. Barbosa, "Knowledge graph embedding for link prediction: A comparative analysis," 2020.

[43] P. Rosso, D. Yang, and P. Cudré-Mauroux, "Beyond triplets: hyper-relational knowledge graph embedding for link prediction," in *Proceedings of The Web Conference*, 2020, pp. 1885–1896.

[44] I. Balazevic, C. Allen, and T. Hospedales, "TuckER: Tensor factorization for knowledge graph completion," in *EMNLP-IJCNLP*, 2019, pp. 5185–5194.

[45] Y. Tay, A. T. Luu, S. C. Hui, and F. Brauer, "Random semantic tensor ensemble for scalable knowledge graph link prediction," in *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, 2017, pp. 751–760.

[46] F. Hasibi, K. Balog, and S. E. Bratsberg, "Dynamic factual summaries for entity cards," in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2017, pp. 773–782.

[47] M. Dehghani, S. Rothe, E. Alfonseca, and P. Fleury, "Learning to attend, copy, and generate for session-based query suggestion," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. Association for Computing Machinery, 2017, pp. 1747–1756.

[48] L. Wu, F. Petroni, M. Josifoski, S. Riedel, and L. Zettlemoyer, "Scalable zero-shot entity linking with dense entity retrieval," 2020.

[49] J. Hoffart, M. A. Yosef, I. Bordino, H. Fürstenau, M. Pinkal, M. Spaniol, B. Taneva, S. Thater, and G. Weikum, "Robust disambiguation of named entities in text," in *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, 2011, pp. 782–792.

[50] L. Ratinov, D. Roth, D. Downey, and M. Anderson, "Local and global algorithms for disambiguation to wikipedia," in *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, 2011, pp. 1375–1384.

[51] G. R. Doddington, A. Mitchell, M. A. Przybocki, L. A. Ramshaw, S. M. Strassel, and R. M. Weischedel, "The automatic content extraction (ace) program-tasks, data, and evaluation." in *Lrec*, vol. 2, no. 1. Lisbon, 2004, pp. 837–840.

[52] D. Milne and I. H. Witten, "Learning to link with wikipedia," in *Proceedings of the 17th ACM conference on Information and knowledge management*, 2008, pp. 509–518.

[53] N. De Cao, G. Izacard, S. Riedel, and F. Petroni, "Autoregressive entity retrieval," in *International Conference on Learning Representations*, 2020.

[54] S. Cucerzan, "Large-scale named entity disambiguation based on wikipedia data," in *Proceedings of the 2007 joint conference on empiri-*

cal methods in natural language processing and computational natural language learning (EMNLP-CoNLL), 2007, pp. 708–716.
[55] Z. Guo and D. Barbosa, "Robust named entity disambiguation with random walks," *Semantic Web*, vol. 9, no. 4, pp. 459–479, 2018.
[56] E. Gabrilovich, M. Ringgaard, and A. Subramanya, "Facc1: Freebase annotation of clueweb corpora, version 1 (release date 2013-06-26, format version 1, correction level 0)," http://lemurproject.org/clueweb12/, 2013, accessed: 2020-10-15.
[57] Yahoo, "Yahoo search query log," http://research.yahoo.com/Academic_Relations, 2013, accessed: 2020-10-16.
[58] D. Vrandečić and M. Krötzsch, "Wikidata: a free collaborative knowledgebase," *Communications of the ACM*, vol. 57, no. 10, pp. 78–85, 2014.
[59] G. Pass, A. Chowdhury, and C. Torgeson, "A picture of search," in *Proceedings of the 1st international conference on Scalable information systems*. Association for Computing Machinery, 2006, pp. 1–es.
[60] B. J. Jansen, A. Spink, C. Blakely, and S. Koshman, "Defining a session on web search engines," *Journal of the American Society for Information Science and Technology*, vol. 58, no. 6, pp. 862–871, 2007.
[61] K. Toutanova, D. Chen, P. Pantel, H. Poon, P. Choudhury, and M. Gamon, "Representing text for joint embedding of text and knowledge bases," in *Proceedings of the 2015 conference on empirical methods in natural language processing*. ACL, 2015, pp. 1499–1509.
[62] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," in *Advances in neural information processing systems*. Curran Associates, Inc., 2013, pp. 2787–2795.
[63] R. Wang, B. Li, S. Hu, W. Du, and M. Zhang, "Knowledge graph embedding via graph attenuated attention networks," *IEEE Access*, vol. 8, pp. 5212–5224, 2019.
[64] R. Bamler, F. Salehi, and S. Mandt, "Augmenting and tuning knowledge graph embeddings," in *Uncertainty in Artificial Intelligence*. PMLR, 2020, pp. 508–518.
[65] Z. Sun, Z.-H. Deng, J.-Y. Nie, and J. Tang, "Rotate: Knowledge graph embedding by relational rotation in complex space," in *International Conference on Learning Representations*. OpenReview.net, 2019. [Online]. Available: https://openreview.net/forum?id=HkgEQnRqYQ
[66] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, and A. M. Rush, "Huggingface's transformers: State-of-the-art natural language processing," 2020.
[67] K. Krippendorff, *Content analysis: An introduction to its methodology*. Thousands Oaks, California 91320: Sage publications, 2018.