# Enriching Documents with Compact, Representative, Relevant Knowledge Graphs

**Shuxin Li**[1] , **Zixian Huang**[1] , **Gong Cheng**[1]* , **Evgeny Kharlamov**[2,3] and **Kalpa Gunaratna**[4]
[1]National Key Laboratory for Novel Software Technology, Nanjing University, China
[2]Bosch Center for Artificial Intelligence, Robert Bosch GmbH, Germany
[3]Department of Informatics, University of Oslo, Norway
[4]Samsung Research America, Mountain View CA, USA
{sxli, zixianhuang}@smail.nju.edu.cn, gcheng@nju.edu.cn, evgeny.kharlamov@de.bosch.com,
k.gunaratna@samsung.com

## Abstract

A prominent application of knowledge graph (KG) is document enrichment. Existing methods identify mentions of entities in a background KG and enrich documents with entity types and direct relations. We compute an entity relation subgraph (ERG) that can more expressively represent indirect relations among a set of mentioned entities. To find compact, representative, and relevant ERGs for effective enrichment, we propose an efficient best-first search algorithm to solve a new combinatorial optimization problem that achieves a trade-off between representativeness and compactness, and then we exploit ontological knowledge to rank ERGs by entity-based document-KG and intra-KG relevance. Extensive experiments and user studies show the promising performance of our approach.

## 1 Introduction

A Knowledge Graph (KG) is a graph where vertices are entities interconnected with relations and annotated with types and attributes [Arenas *et al.*, 2016]. Increasingly many KGs have been developed for various domains and applications [Kharlamov *et al.*, 2017a; Kharlamov *et al.*, 2017b]. An important application is *document enrichment* where words or phrases in a document are linked to entities in a given background KG, and then the KG is leveraged to help readers better comprehend the document with entity types [Tonon *et al.*, 2016] and direct relations between pairs of entities [Gunaratna *et al.*, 2017]. However, these methods cannot find indirect relations among a set of entities, represented as an *entity relation subgraph* (ERG). As illustrated in Fig. 1, an ERG in a KG connects three companies mentioned in a news article and usefully shows how they are indirectly related via an intermediate company [Huang *et al.*, 2019].

**Challenges.** There exist algorithms for searching and ranking ERGs to connect a given set of entities [Tong and Faloutsos, 2006; Kasneci *et al.*, 2009a; Kasneci *et al.*, 2009b; Chen *et al.*, 2011; Cheng *et al.*, 2016; Cheng *et al.*, 2017; Cheng *et al.*, 2019], but directly using them for document enrichment faces two challenges. (1) To properly enrich a document, an ERG should be *representative*—covering all or the salient entities mentioned in the document. Users also prefer *compact* ERGs where entities are connected by short paths [Cheng *et al.*, 2017]. However, representativeness and compactness are conflicting. There may not exist a compact ERG that covers all the salient entities if they are disconnected or far from each other in the KG. (2) For effective enrichment, an ERG should be *relevant* to the context, namely the document content, which is ignored by existing context-independent methods for ranking ERGs [Cheng *et al.*, 2017].

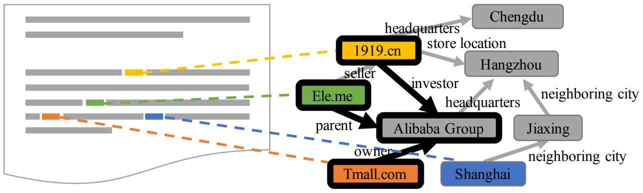


Figure 1: A document mentioning four entities in a KG. The bold subgraph of the KG is an ERG connecting three mentioned entities.

**Our approach.** We overcome these limitations with a two-step approach named **CR**$^2$. In the first step, we achieve a trade-off between representativeness and compactness by computing the most salient subset of entities mentioned in the document such that compact ERGs connecting these entities are guaranteed to exist in the KG. Then we easily find all such compact ERGs by performing an existing search algorithm [Cheng *et al.*, 2019]. In the second step, we rank these ERGs by their relevance and return the top-ranked result.

**Technical contributions.** (1) To balance representativeness and compactness, we model a new combinatorial optimization problem and we present an efficient best-first search algorithm. (2) We consider both document-KG and intra-KG relevance. Our novel measure effectively exploits ontological knowledge to compute entity-based relevance.

## 2 Preliminaries

**Knowledge graph.** A KG is an undirected graph $G = \langle \mathbb{V}, \mathbb{E} \rangle$ where $\mathbb{V}$ is a set of vertices representing entities and $\mathbb{E} \subseteq \mathbb{V} \times \mathbb{V}$ is a set of undirected edges representing entity relations. Edge directions, relation types, and entity attributes

*Contact Author

are ignored since they are not essential to our approach. Each entity $v \in \mathbb{V}$ is annotated with its type which is a class defined in an ontology, denoted by $\texttt{type}(v) \in \mathbb{T}$. Classes in $\mathbb{T}$ constitute a subsumption hierarchy. We only consider class hierarchies and leave more general ontologies for future work.

**Graph terminology.** A neighbor of vertex $v$ is adjacent to $v$ by an edge. Let $\texttt{NBR}(v) \subseteq \mathbb{V}$ represent all the neighbors of $v$ in $G$. We define path, tree, and subgraph in a standard way. The distance between two vertices $u$ and $v$, denoted by $\texttt{dist}(u, v)$, is the length of a shortest path connecting $u$ and $v$ in $G$, or $+\infty$ if no such path exists. The diameter of a graph, denoted by $\texttt{diam}(\cdot)$, is the greatest distance between pairs of vertices in the graph. A central vertex in a graph minimizes eccentricity, i.e., the greatest distance from other vertices.

**ERG.** Let $Q \subseteq \mathbb{V}$ be a subset of vertices in $G$ with $|Q| \geq 2$. An ERG connecting $Q$, denoted by $G' = \langle V_{G'}, E_{G'} \rangle$, is a subgraph of $G$ such that (1) $G'$ is connected, (2) $G'$ contains $Q$, i.e., $Q \subseteq V_{G'}$, and (3) $G'$ is minimal w.r.t. (1) and (2), i.e., no proper subgraph of $G'$ satisfies (1) and (2). Therefore, each ERG is a tree where leaf vertices are from $Q$.

## 3 Balancing Representativeness and Compactness

The first step of CR$^2$ finds compact ERGs connecting the most salient entities mentioned in a document. The novelty is an efficient algorithm for a new combinatorial optimization problem that balances representativeness and compactness.

### 3.1 Problem Formulation

The compactness of an ERG can be measured by its diameter. For a set of input entities, Cheng *et al.* [2019] presented an algorithm named **OptimSearch** which efficiently searches a KG and finds ERGs that connect all the input entities and do not exceed a predefined diameter bound. In brief, OptimSearch performs pruned searches starting from each input entity and merges paths having a common end vertex into an ERG. For details please see Cheng *et al.* [2019].

We want to reuse OptimSearch, but this algorithm will return empty results if no ERG can be found due to the long distance between some input entities in the KG. Unfortunately, it happens frequently in our application of document enrichment because a document such as a news article often mentions a variety of entities. When it is impossible to optimize both representativeness and compactness of an ERG, our proposed trade-off is to find ERGs that *maximize their representativeness while having bounded compactness*. We achieve this trade-off by (1) computing the most salient subset of mentioned entities such that ERGs connecting these entities with a bounded diameter are guaranteed to exist in the KG, and then (2) performing OptimSearch with these entities as input to efficiently find compact ERGs connecting them.

Formally, let $Q \subseteq \mathbb{V}$ be the set of entities in $G$ that are mentioned in a document, which can be identified by any off-the-shelf entity linker [Shen *et al.*, 2015]. For entity $v \in Q$, let $\texttt{sal}(v) \geq 0$ be the salience of $v$ in the document. The concrete implementation of $\texttt{sal}(v)$ is outside the scope of our research. For our experiments we will rely on an existing implementation [Ni *et al.*, 2016] where $v$ and the document are both represented as TF-IDF weighted word vectors and their cosine similarity is calculated as $\texttt{sal}(v)$. Let $D$ be a predefined diameter bound. A subset of entities $Q' \subseteq Q$ with $|Q'| \geq 2$ is *representable*, denoted by $\texttt{repr}(Q') = yes$, if $G$ contains an ERG $G'$ connecting $Q'$ with $\texttt{diam}(G') \leq D$; otherwise we write $\texttt{repr}(Q') = no$. For example, consider $Q = \{\texttt{1919.cn}, \texttt{Ele.me}, \texttt{Tmall.com}, \texttt{Shanghai}\}$ in Fig. 1. Given $D = 3$ , whereas $Q$ is not representable, its subset $Q' = \{\texttt{1919.cn}, \texttt{Ele.me}, \texttt{Tmall.com}\}$ is representable due to the existence of an ERG connecting $Q'$ with $\texttt{diam} = 2$.

Our primary task is a combinatorial optimization problem:

$$\begin{aligned} Q_{\max} &= \mathop{\arg\max}_{Q' \subseteq Q \text{ such that } \texttt{repr}(Q') = yes} \texttt{score}(Q')\,, \\ &\text{where } \texttt{score}(Q') = \sum_{v \in Q'} \texttt{sal}(v)\,. \end{aligned} \tag{1}$$

Equation (1) is a new problem. Its efficient solution is our focus below in this section. Then we can easily perform OptimSearch with $Q_{\max}$ as input to find all the ERGs that connect all the entities in $Q_{\max}$ and have a diameter of $D$ at most.

### 3.2 Theoretical Foundations

A naive way of computing $Q_{\max}$ is to decide the representability of each subset $Q' \subseteq Q$ by actually searching for a diameter-bounded ERG. However, $Q$ has an exponential number of subsets, and performing that many searches is inefficient. Our solution relies on the following lemma and corollary to overcome these limitations.

Lemma 1 suggests a convenient way of deciding the representability of $Q'$ by distance computation, thereby avoiding the expensive process of actually searching for an ERG.

**Lemma 1.** *$Q'$ is representable iff there exists $c \in \mathbb{V}$ such that*

1. *for each $q \in Q'$, it holds that $\texttt{dist}(q, c) \leq \lceil \frac{D}{2} \rceil$, and*
2. *if $D$ is odd and there exists $q \in Q'$ such that $\texttt{dist}(q, c) = \lceil \frac{D}{2} \rceil$, then there must exist $c' \in \texttt{NBR}(c)$ with $\texttt{dist}(q, c') = \lceil \frac{D}{2} \rceil - 1$ for all such $q$.*

*Proof.* We prove necessity by showing that a central vertex in a diameter-bounded ERG connecting $Q'$ satisfies the conditions of $c$. We prove sufficiency by merging certain shortest paths from each $q \in Q'$ to $c$ into a diameter-bounded ERG. Full proof: Theorem 1 in Li *et al.* [2020]. □

The existence of $c$ when $D$ is even, or the existence of $\langle c, c' \rangle$ when $D$ is odd, is called a *certificate* for the representability of $Q'$. We say (the representability of) $Q'$ is certified with $c$. For example, $\{\texttt{1919.cn}, \texttt{Ele.me}, \texttt{Tmall.com}\}$ in Fig. 1 is certified with $\texttt{Alibaba Group}$ for all $D \geq 2$.

Corollary 1 shows certificates are close to $Q'$, yielding a smaller search space of a certificate than Lemma 1 for odd $D$.

**Corollary 1.** *The entity $c$ in Lemma 1 also satisfies that there exists $q \in Q'$ such that $\texttt{dist}(q, c) \leq \lfloor \frac{D}{2} \rfloor$.*

*Proof.* When $D$ is even, the corollary is derived from Condition 1 of Lemma 1. For odd $D$, we can easily prove by contradiction. We omit the details due to space limitations. □

**Algorithm 1** Computation of $Q_{\text{max}}$

```
Input: $G, Q, D$;    Output: $Q_{\text{max}}$
 1: $P \leftarrow$ empty priority queue
 2: for all $q \in Q$ do
 3:     $P$.insert($\langle q, q, pr_{q|q} \rangle$)
 4:     $V_q \leftarrow \{q\}$
 5: $C \leftarrow \emptyset$
 6: $Q_{\text{max}} \leftarrow \emptyset$
 7: while $P$ is not empty do
 8:     $\langle c, q, pr_{c|q} \rangle \leftarrow P$.pull()
 9:     if $pr_{c|q} \leq \mathtt{score}(Q_{\text{max}})$ then
10:         break the while loop
11:     if $c \notin C$ then
12:         $Q_c \leftarrow$ QMaxCertWith($G$, $Q$, $c$, $D$)
13:         $C \leftarrow C \cup \{c\}$
14:         if $\mathtt{score}(Q_c) > \mathtt{score}(Q_{\text{max}})$ then
15:             $Q_{\text{max}} \leftarrow Q_c$
16:     if $\mathtt{dist}(c, q) < \lfloor \frac{D}{2} \rfloor$ then
17:         for all $c' \in \mathtt{NBR}(c)$ do
18:             if $c' \notin V_q$ then
19:                 $P$.insert($\langle c', q, pr_{c'|q} \rangle$)
20:                 $V_q \leftarrow V_q \cup \{c'\}$
21: return $Q_{\text{max}}$
```

## 3.3 Algorithm

Now we describe our algorithm for computing $Q_{\text{max}}$. We do not know which $q \in Q$ to include in $Q_{\text{max}}$. So the basic idea is for each $q \in Q$ and each entity $c$ that is at most $\lfloor \frac{D}{2} \rfloor$ hops away from $q$, we find the optimal subset of $Q$ that is certified with $c$, and we take the optimal subset over all such $q$ and $c$. Further, rather than brute-force search, we perform *best-first search* by visiting the most promising $c$ first, and terminating the search process when it is guaranteed that no better subset of $Q$ can be certified with an unvisited entity. The idea is detailed in Algorithm 1.

We run $|Q|$ independent searches starting from distinct input entities in $Q$. The frontiers of the searches are kept in a shared priority queue $P$ comprising entity-entity-priority triples (lines 1–3). During each search starting from $q \in Q$, a triple $\langle c, q, pr_{c|q} \rangle$ represents a possible certificate $c$ for some subset of $Q$ with priority $pr_{c|q}$. We will detail the computation of priority later. Each search maintains its own set of visited entities $V_q$ (line 4). An entity $c$ can be visited in different searches, but is checked at most once using the subroutine *QMaxCertWith* which finds $Q_c$, the optimal subset of $Q$ that is certified with $c$. We will detail QMaxCertWith later. The set of checked entities is referred to as $C$ (line 5).

$Q_{\text{max}}$ denotes the optimal representable subset of $Q$ found so far (line 6). Iteratively, the algorithm performs best-first search to check entities that are at most $\lfloor \frac{D}{2} \rfloor$ hops away from each $q \in Q$ (lines 7–20). In each iteration, we pull out of $P$ the triple $\langle c, q, pr_{c|q} \rangle$ having the highest priority $pr_{c|q}$ (line 8), which represents an upper bound on the $\mathtt{score}$ of subsets of $Q$ that can be certified with $c$ or its descendants in the search starting from $q$. Therefore, if $pr_{c|q}$ is not better than the current $\mathtt{score}(Q_{\text{max}})$, the algorithm can be terminated and the current $Q_{\text{max}}$ is guaranteed to be optimal (lines 9–10). Otherwise, a better subset of $Q$ may be certified with $c$ or its descendant. If $c$ has not been checked in other searches, it will be checked using QMaxCertWith to find $Q_c$ (lines 11–13). If $Q_c$ is better than the current $Q_{\text{max}}$, a substitution will be made to update $Q_{max}$ (lines 14–15). The search then continues and will expand the unvisited neighbors of $c$ if they are at most $\lfloor \frac{D}{2} \rfloor$ hops away from $q$ (lines 16–20).

**Algorithm 2** QMaxCertWith

```
Input: $G, Q, c, D$;    Output: $Q_c$
 1: $Q_c \leftarrow \emptyset$
 2: $S \leftarrow \{q \in Q : \mathtt{dist}(q, c) \leq \lceil \frac{D}{2} \rceil\}$
 3: $T \leftarrow \{q \in Q : \mathtt{dist}(q, c) = \lceil \frac{D}{2} \rceil\}$
 4: if ($D$ is even or $T = \emptyset$) and $|S| \geq 2$ then
 5:     $Q_c \leftarrow S$
 6: else
 7:     for all $c' \in \mathtt{NBR}(c)$ do
 8:         $V_{c'} \leftarrow (S \setminus T) \cup \{q \in T : \mathtt{dist}(q, c') = \lceil \frac{D}{2} \rceil - 1\}$
 9:         if $\mathtt{score}(V_{c'}) > \mathtt{score}(Q_c)$ and $|V_{c'}| \geq 2$ then
10:             $Q_c \leftarrow V_{c'}$
11: return $Q_c$
```

**Subroutine QMaxCertWith.** The computation of $Q_{\text{max}}$ relies on QMaxCertWith. This subroutine is detailed in Algorithm 2. It computes $Q_c$, the optimal subset of $Q$ that is certified with $c$, or returns $\emptyset$ if no such subset exists (line 1). Let $S$ be the subset of $Q$ that satisfy Condition 1 of Lemma 1 given $c$ (line 2). Let $T$ be the subset of $Q$ that need to be considered for Condition 2 of Lemma 1 (line 3). When $D$ is even or $T$ is empty, Condition 2 of Lemma 1 is not triggered, and Condition 1 tells that $Q_c$ is exactly $S$ (lines 4–5). Otherwise, according to Condition 2, entities in $T$ that appear in a subset of $Q$ certified with $c$ should satisfy additional distance constraints about some $c' \in \mathtt{NBR}(c)$. Therefore, for each $c' \in \mathtt{NBR}(c)$ we find the optimal subset of $Q$ for which $\langle c, c' \rangle$ is a certificate, denoted by $V_{c'}$ (lines 7–8), and we take the optimal subset over all such $c'$ as $Q_c$ (lines 9–10).

**Computation of priority.** The best-first search for $Q_{\text{max}}$ relies on priority $pr_{c|q}$. It represents an upper bound on the $\mathtt{score}$ of subsets of $Q$ that can be certified with $c$ or its descendants in the search starting from $q$:

$$
\begin{aligned}
& pr_{c|q} = \mathtt{score}(Q_{\text{ub}}(c|q))\,, \\
& \text{where } Q_{\text{ub}}(c|q) = \{q\} \cup \{q' \in (Q \setminus \{q\}) : \\
& \qquad\qquad \mathtt{dist}(c, q) + \mathtt{dist}(c, q') \leq D\}\,.
\end{aligned} \tag{2}
$$

This heuristic guarantees the optimality of the returned $Q_{\text{max}}$.

**Theorem 1.** *Algorithm 1 correctly solves the task of Eq. (1).*

*Proof.* We prove by contradiction. We show that before the algorithm returns a suboptimal representable subset of $Q$, it has found the optimal subset by checking entities along a shortest path from some $q \in Q$ to $c$ using QMaxCertWith. Full proof: https://github.com/nju-websoft/CR2. □

**Run-time analysis.** For input $G = \langle \mathbb{V}, \mathbb{E} \rangle$ and $Q$, let $n = |\mathbb{V}|$, $m = |\mathbb{E}|$, $g = |Q|$. Let $d$ be the run-time of computing $\mathtt{dist}$. The run-time of Algorithm 1 consists of:

- $O(g(n+m))$ for $g$ searches,
- $O(g^2nd)$ for $O(gn)$ computations of priority,
- $O(gn\log(gn))$ for $O(gn)$ insert-pull pairs of priority queue operations, and
- $O((n+m)gd)$ for $O(n)$ calls of QMaxCertWith.

For large KGs, online calculation of `dist` is time-consuming while offline materialization of `dist` between all pairs of vertices requires prohibitively large space. We achieve a tradeoff using a *distance oracle* [Akiba *et al.*, 2013]. This offline precomputed index stores for each vertex its distance from a set of landmark vertices in the graph. It has a moderate size and allows reasonably fast online calculation of `dist` for every pair of vertices. For details please see Akiba *et al.* [2013].

# 4 Measuring Relevance

We have computed $Q_{\max}$ and used OptimSearch to find representative and compact ERGs connecting $Q_{\max}$ in $G$. In the second step of CR$^2$, we rank these ERGs by their relevance. The novelty is to measure both document-KG and intra-KG relevance based on entities and ontological knowledge.

## 4.1 Problem Formulation

Let $\mathbf{X}$ be a set of ERGs computed in the first step. Our primary task is to measure the relevance of each $G' \in \mathbf{X}$, denoted by $\texttt{rel}(G')$. We consider relevance from two perspectives: the relevance of $G'$ to the document content ($\texttt{rel}_\text{D}$), and the relevance of the elements in $G'$ to each other ($\texttt{rel}_\text{I}$):

$$\texttt{rel}(G') = (1-\alpha)\cdot\texttt{rel}_\text{D}(G') + \alpha\cdot\texttt{rel}_\text{I}(G')\,, \tag{3}$$

where $\alpha \in [0,1]$ is a parameter to tune the relative importance of the two perspectives. Below we compute $\texttt{rel}_\text{D}$ and $\texttt{rel}_\text{I}$.

## 4.2 Document-KG Relevance ($\texttt{rel}_\textbf{D}$)

To effectively enrich a document, an ERG $G' = \langle V_{G'}, E_{G'}\rangle$ should be relevant to the document content. Whereas previous research computes the relevance of an ERG to a document using a simple word-based measure [Viswanathan and Ilango, 2012], we propose to measure *entity-based relevance* and hence we can exploit *ontological knowledge*.

Formally, recall that $Q$ denotes the set of entities in $G$ that are mentioned in the document. We compute the document-KG relevance of $G'$ by calculating the average relatedness ($\texttt{r}$) between pairs of entities in $Q$ and $V_{G'}$ as follows:

$$\texttt{rel}_\text{D}(G') = \frac{1}{|Q|\cdot|V_{G'}|}\sum_{v_i\in Q,\ v_j\in V_{G'}}\texttt{r}(v_i,v_j)\,. \tag{4}$$

For $\texttt{r}(v_i,v_j)$, recall that each entity $v$ is annotated with its type which is a class denoted by $\texttt{type}(v)\in\mathbb{T}$. We consider a $|\mathbb{T}|$-dimensional vector space where the $l$-th dimension corresponds to class $t_l \in \mathbb{T}$. We represent $v_i, v_j$ as two vectors $\overrightarrow{v_i}, \overrightarrow{v_j}$ in this space and calculate their cosine similarity:

$$\texttt{r}(v_i,v_j) = \cos(\overrightarrow{v_i},\overrightarrow{v_j})\,. \tag{5}$$

For $\overrightarrow{v_i}, \overrightarrow{v_j}$, we use their types and also the types of their neighbors in $G$. Neighbors help to catch implicit relatedness. Specifically, let $\overrightarrow{v_{i,l}}$ be the $l$-th dimension of $\overrightarrow{v_i}$. We define

$$\overrightarrow{v_{i,l}} = \begin{cases}1 & \text{if } t_l = \texttt{type}(v_i)\,,\\ \texttt{tf-ief}(t_l|v_i) & \text{otherwise}\,,\end{cases} \tag{6}$$

where $\texttt{tf-ief}(t_l|v_i) \in [0,1]$ denotes the *type frequency—inverse entity frequency* of $t_l$ w.r.t. $v_i$. It adapts the well-known term frequency—inverse document frequency in information retrieval to knowledge graphs:

$$\begin{aligned}&\texttt{tf-ief}(t_l|v_i) = \texttt{tf}(t_l|v_i)\cdot\texttt{ief}(t_l)\,,\\ &\text{where } \texttt{tf}(t_l|v_i) = \frac{|\{v\in\texttt{NBR}(v_i) : \texttt{type}(v) = t_l\}|}{|\texttt{NBR}(v_i)|}\,,\\ &\text{and } \texttt{ief}(t_l) = \frac{\texttt{ic}(t_l)}{\log|\mathbb{V}|}\,,\end{aligned} \tag{7}$$

where $\texttt{tf}(t_l|v_i) \in [0,1]$ calculates the proportion of $t_l$'s instances in $v_i$'s neighbors, and $\texttt{ief}(t_l) \in [0,1]$ calculates the normalized information content of $t_l$. We follow the standard definition of information content:

$$\texttt{ic}(t_l) = -\log\frac{|v\in\mathbb{V} : t_l\in\texttt{TYPS}(v)|}{|\mathbb{V}|}\,. \tag{8}$$

It is inversely proportional to the logarithm of the number of instances of $t_l$. Here, $\texttt{TYPS}(v)$ denotes a set of classes consisting of $\texttt{type}(v)$ and its ancestors in the class hierarchy of $\mathbb{T}$. In other words, we perform *subsumption-based reasoning* to obtain implicit entity types from the hierarchy.

## 4.3 Intra-KG Relevance ($\texttt{rel}_\textbf{I}$)

Users prefer ERGs where elements are relevant to each other, e.g., containing entities having the same or similar types because a set of homogeneous entities form a semantically cohesive whole that is more meaningful than a set of divergent entities [Cheng *et al.*, 2017]. To implement such *entity-based relevance*, we (again) exploit *ontological knowledge*.

Formally, we compute the intra-KG relevance of $G'$ by calculating the average similarity ($\texttt{sim}$) between pairs of entities in $V_{G'}$ as follows:

$$\texttt{rel}_\text{I}(G') = \frac{1}{\binom{|V_{G'}|}{2}}\sum_{v_i,v_j\in V_{G'},\ v_i\neq v_j}\texttt{sim}(v_i,v_j)\,. \tag{9}$$

For $\texttt{sim}(v_i,v_j)$, inspired by Zhu *et al.* [2017], we compute the similarity between $\texttt{type}(v_i)$ and $\texttt{type}(v_j)$ based on their *relative positions in the class hierarchy* of $\mathbb{T}$. Two classes are similar if they are close together and are commonly subsumed under a specific class in the hierarchy:

$$\texttt{sim}(v_i,v_j) = \frac{1}{1+\texttt{len}(\texttt{type}(v_i),\texttt{type}(v_j))\cdot k^{\texttt{ic}(\texttt{lcs}(\texttt{type}(v_i),\texttt{type}(v_j)))}}\,, \tag{10}$$

where $\texttt{len}(\texttt{type}(v_i),\texttt{type}(v_j))$ is the length of a shortest path between $\texttt{type}(v_i)$ and $\texttt{type}(v_j)$ in the class hierarchy, $k \in (0,1]$ is a parameter to tune, and $\texttt{ic}(\texttt{lcs}(\texttt{type}(v_i),\texttt{type}(v_j)))$ is the information content ($\texttt{ic}$) of the least common subsumer ($\texttt{lcs}$) of $\texttt{type}(v_i)$ and $\texttt{type}(v_j)$ in the class hierarchy. The $\texttt{lcs}$ of two classes is their lowest-level shared ancestor in the class hierarchy.

# 5 Experiments

Our main research hypothesis tested in the experiments is **RH1**: our CR$^2$ approach can *effectively* enrich documents with top-ranked ERGs. Due to the lack of benchmark for this

| Question | Response (mean±SD) | | | rANOVA | LSD post-hoc |
|---|---|---|---|---|---|
| | CR$^2$ | RDF2Vec | RankingSA | ($p$-value) | ($p < 0.05$) |
| Q1: This graph is relevant to the document. | **2.68**±0.83 | 2.52±0.83 | 2.48±0.85 | 6.606e-7 | CR$^2$ > RDF2Vec = RankingSA |
| Q2: This graph is meaningful. | **2.79**±0.79 | 2.70±0.79 | 2.60±0.83 | 1.000e-5 | CR$^2$ > RDF2Vec > RankingSA |
| Q3: This graph helped me comprehend the document. | **2.60**±0.90 | 2.49±0.88 | 2.42±0.89 | 1.410e-4 | CR$^2$ > RDF2Vec = RankingSA |

Table 1: Questions and responses (4-point Likert scale) on top-ranked ERGs.

task, we conducted a user study and we compared CR$^2$ with a state-of-the-art method [Viswanathan and Ilango, 2012]. Besides, the computation of $Q_{\text{max}}$ in CR$^2$ is a non-trivial problem. We tested research hypothesis **RH2**: our Algorithm 1 can *efficiently* compute $Q_{\text{max}}$. Since we were the first to address this problem, we compared our algorithm with two naive baselines. Our experiments were performed on an Intel Xeon E5-1607 (3.10 GHz) with 40GB memory for Java. Source code and data: https://github.com/nju-websoft/CR2.

### 5.1 Datasets

**Documents.** We ran our experiments over documents sampled from the Signal Media One-Million News Articles Dataset (Signal-1M) [Corney *et al.*, 2016]. This large and diverse dataset contains 1M news and blog articles in English, collected from 93K news sources.

**KG.** We used the well-known DBpedia [Lehmann *et al.*, 2015] (version 2016-10) as our KG. This encyclopedic KG in RDF format contains 5.9M entities and 18.3M relations extracted from Wikipedia. We imported entity relations from the *Mappingbased Objects* file, entity types from the *Instance Types* file, and a class hierarchy from the *DBpedia Ontology*.

**Entities.** We used DBpedia Spotlight [Mendes *et al.*, 2011], a tool recommended by DBpedia, to identify mentions of DBpedia entities in documents. Our analysis showed that the mean and median of the number of entities mentioned in a document are 10.16 and 7, respectively.

### 5.2 RH1: Effectiveness of Ranking ERGs

**Procedure.** We used 100 documents from Signal-1M. To improve their diversity, for each $|Q| = 2, 4, \ldots, 20$, we randomly sampled 10 documents mentioning $|Q|$ entities identified by DBpedia Spotlight. We recruited 25 university students to participate, and we assigned each document to 5 participants. Participants were provided with the top-ranked ERG returned by each method to compare in a blind manner, i.e., ERGs returned by different methods were shown in random order and the methods were anonymous. For each ERG, participants responded to three questions as 4-point Likert items shown in Table 1: strongly disagree (1), disagree (2), agree (3), or strongly agree (4).

**Configuration of CR$^2$.** We set the diameter bound $D = 4$ following Cheng *et al.* [2017]. For Eq. (3), we set $\alpha = 0.5$ but we may obtain better results by tuning this parameter. For Eq. (10), we set $k = 0.4$ following Zhu *et al.* [2017].

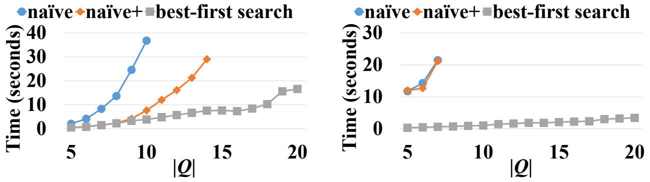
**Baselines.** We compared CR$^2$ with two baseline methods. The first baseline, **RankingSA** [Viswanathan and Ilango, 2012], is a state-of-the-art context-aware method for ranking ERGs. It computes the relevance of an ERG to a document using their word-based similarity. For a fair comparison, RankingSA is fed with ERGs found in the first step of CR$^2$ to rank. The second baseline is a variant of CR$^2$. Recall that a main feature of relevance measurement in CR$^2$ is the use of ontological knowledge. In this variant, ontology is replaced by **RDF2Vec** [Ristoski *et al.*, 2019], a state-of-the-art graph embedding method for KGs in RDF format. Specifically, the vector representations of entities $\overrightarrow{v_i}, \overrightarrow{v_j}$ in Eq. (5) are their embedding vectors generated by RDF2Vec (K2V SG 200). The similarity between entities $\texttt{sim}(v_i, v_j)$ in Eq. (9) is the cosine similarity between their embedding vectors.

**Results.** As shown in Table 1, on all the three questions Q1–Q3, CR$^2$ obtained the highest scores and significantly outperformed the two baselines according to LSD post-hoc tests ($p < 0.05$), thereby *supporting research hypothesis RH1*. Specifically, ERGs selected by CR$^2$ were more relevant to documents (Q1) and more effectively helped participants comprehend documents (Q3). Our entity-based ontological relevance was demonstrated to have an advantage over embedding-based relevance (RDF2Vec) and word-based relevance (RankingSA). Further, CR$^2$ and RDF2Vec selected more meaningful ERGs than RankingSA (Q2). We attributed this improvement to the consideration of intra-KG relevance in CR$^2$ and RDF2Vec. Their cohesive ERGs were more meaningful to humans. However, there is still room for improving CR$^2$ since the mean responses it received were in the range of 2.60–2.79, between borderline (2.5) and agree (3). It was partially due to the inadequacy of available knowledge in DBpedia. We will experiment with other KGs in future work.

### 5.3 RH2: Efficiency of Computing $Q_{\text{max}}$

**Procedure.** We used 6,400 documents from Signal-1M. To improve their diversity, for each $|Q| = 5, 6, \ldots, 20$, we randomly sampled 400 documents mentioning $|Q|$ entities identified by DBpedia Spotlight. We set a timeout of 100 seconds. A run of an algorithm on a document was terminated when reaching the timeout, and its run-time was set to the timeout value. We experimented with two diameter bounds having opposite parity following Cheng *et al.* [2019]: $D = 4$ and $D = 3$. Larger values of $D$ would be too relaxed to bound effectively due to the small-world phenomenon observed in DBpedia [Cheng *et al.*, 2016; Cheng *et al.*, 2019].

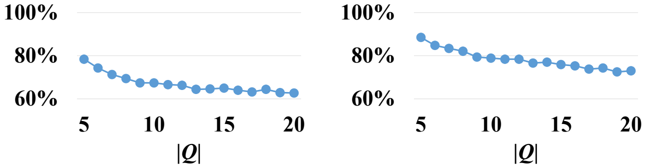
**Baselines.** Since we were the first to address the problem of computing $Q_{\text{max}}$ defined in Eq. (1), we compared our Algorithm 1 with two naive baseline algorithms. The first baseline, denoted by **naive**, finds $Q_{\text{max}}$ in an Apriori-style manner. Iteratively, it joins pairs of representable subsets of $Q$ containing



(a) $D = 4$

(b) $D = 3$

Figure 2: Run-time for computing $Q_{\max}$.



(a) $\frac{\texttt{score}(Q_{\max})}{\texttt{score}(Q)}$

(b) $\frac{|Q_{\max}|}{|Q|}$

Figure 3: Representativeness of computed ERGs under $D = 4$.

$i$ entities into larger subsets containing $i + 1$ entities, and removes generated non-representable subsets. Finally, it returns the optimal representable subset as $Q_{\max}$. To decide the representability of a subset $Q'$, it performs the OptimSearch algorithm [Cheng *et al.*, 2019] to search for diameter-bounded ERGs that connect $Q'$, and is informed of $\texttt{repr}(Q') = yes$ when the first of such ERGs is found, or $\texttt{repr}(Q') = no$ if no such ERG exists. The second baseline, denoted by **naive+**, is an enhanced version of naive. Recall that OptimSearch produces ERGs by searching and merging paths. In naive+, paths are cached to speed up OptimSearch.

**Results.** The average run-time of each algorithm on a document is shown in Fig. 2a and Fig. 2b for $D = 4$ and $D = 3$, respectively. To avoid reporting distorted results caused by timeout, when the proportion of timeout runs of an algorithm exceeded 10% for some $|Q|$, its run-time at that point is not shown in the figures. The run-time of naive and naive+ rose quickly when $|Q|$ increased because the number of subsets of $Q$ they processed was exponential in $|Q|$. Timeout runs reached 10% after $|Q| > 10$ for naive and $|Q| > 14$ for naive+ in Fig. 2a, and only after $|Q| > 7$ for both algorithms in Fig. 2b, indicating their poor scalability. By contrast, our best-first search based Algorithm 1 never reached timeout and its run-time grew slowly. When $|Q| = 7$, the median of the number of entities mentioned in a document, our algorithm only used 1.50 seconds under $D = 4$ and 0.61 second under $D = 3$, thereby *supporting research hypothesis RH2*. However, its run-time became less satisfying when $|Q|$ was close to 20, but such documents were very rare in the corpus. Besides, Fig. 3 depicts the representativeness of ERGs computed under $D = 4$. We observed $\frac{\texttt{score}(Q_{\max})}{\texttt{score}(Q)} > 70\%$ in Fig. 3a and $\frac{|Q_{\max}|}{|Q|} > 60\%$ in Fig. 3b for all $|Q|$, showing that these compact ERGs were reasonably representative, thereby *demonstrating the possibility of satisfyingly balancing representativeness and compactness*.

## 6 Related Work

To enrich a document with KGs, in contrast with Gunaratna *et al.* [2017] extracting direct relations (i.e., edges) between mentioned entities, we extract ERGs which can more expressively represent indirect relations (i.e., subgraphs) among a set of entities. Schuhmacher *et al.* [2014] extract all the (numerous) length-bounded paths between mentioned entities. Whereas this volume of enrichment is suitable for machine use, e.g., for computing document similarity, we consider human readers and hence we enrich with compact subgraphs. Apart from KGs, other resources such as Q&A pairs [Tang *et al.*, 2017] and data visualizations [Lin *et al.*, 2018] have also been exploited. Compared with our work, they face different challenges and use fundamentally different techniques.

From a technical point of view, our approach builds on a recent algorithm for finding diameter-bounded ERGs [Cheng *et al.*, 2019], but this algorithm will easily return empty results if it is directly used. A trade-off between representativeness and compactness is needed, for which we model a new combinatorial optimization problem and propose an efficient solution. Our solution extends Li *et al.* [2020]; entity salience is now supported, and the depth of search is improved from $\lceil \frac{D}{2} \rceil$ to $\lfloor \frac{D}{2} \rfloor$. Moreover, different from existing context-independent methods for ranking ERGs [Cheng *et al.*, 2017], we consider document-KG relevance. Our approach exploits ontological knowledge to compute entity-based relevance. It outperforms word-based relevance [Viswanathan and Ilango, 2012] and graph embedding based relevance [Ristoski *et al.*, 2019]. The use of ontology also distinguishes our work from community search over social networks [Chen *et al.*, 2019] which relies on graph structure for relevance measurement.

## 7 Conclusion

We studied how to efficiently compute compact, representative, and relevant ERGs from a KG to connect a set of entities mentioned in a document for enrichment. We balanced representativeness and compactness by solving a new combinatorial optimization problem with a best-first search algorithm, and we measured document-KG and intra-KG relevance based on entities and ontological knowledge. The computed ERGs may find application in Web browsing and computer-assisted journalism.

In future work, we will explore learning to rank ERGs. We also plan to improve the overall efficiency of our implementation by integrating multiple steps into a hybrid process.

## Acknowledgements

This work was partially supported by the National Key R&D Program of China (2018YFB1005100) and the Six Talent Peaks Program of Jiangsu Province (RJFW-011).

## References

[Akiba *et al.*, 2013] Takuya Akiba, Yoichi Iwata, and Yuichi Yoshida. Fast exact shortest-path distance queries on large networks by pruned landmark labeling. In *SIGMOD*, pages 349–360, 2013.

[Arenas *et al.*, 2016] Marcelo Arenas, Bernardo Cuenca Grau, Evgeny Kharlamov, Sarunas Marciuska, and Dmitriy Zheleznyakov. Faceted search over rdf-based knowledge graphs. *J. Web Semant.*, 37-38:55–74, 2016.

[Chen *et al.*, 2011] Chen Chen, Guoren Wang, Huilin Liu, Junchang Xin, and Ye Yuan. SISP: a new framework for searching the informative subgraph based on PSO. In *CIKM*, pages 453–462, 2011.

[Chen *et al.*, 2019] Lu Chen, Chengfei Liu, Kewen Liao, Jianxin Li, and Rui Zhou. Contextual community search over large social networks. In *ICDE*, pages 88–99, 2019.

[Cheng *et al.*, 2016] Gong Cheng, Daxin Liu, and Yuzhong Qu. Efficient algorithms for association finding and frequent association pattern mining. In *ISWC, Part I*, pages 119–134, 2016.

[Cheng *et al.*, 2017] Gong Cheng, Fei Shao, and Yuzhong Qu. An empirical evaluation of techniques for ranking semantic associations. *IEEE Trans. Knowl. Data Eng.*, 29(11):2388–2401, 2017.

[Cheng *et al.*, 2019] Gong Cheng, Daxin Liu, and Yuzhong Qu. Fast algorithms for semantic association search and pattern mining. *IEEE Trans. Knowl. Data Eng.*, Early Access:1–13, 2019.

[Corney *et al.*, 2016] David Corney, Dyaa Albakour, Miguel Martinez-Alvarez, and Samir Moussa. What do a million news articles look like? In *NewsIR*, pages 42–47, 2016.

[Gunaratna *et al.*, 2017] Kalpa Gunaratna, Amir Hossein Yazdavar, Krishnaprasad Thirunarayan, Amit P. Sheth, and Gong Cheng. Relatedness-based multi-entity summarization. In *IJCAI*, pages 1060–1066, 2017.

[Huang *et al.*, 2019] Zixian Huang, Shuxin Li, Gong Cheng, Evgeny Kharlamov, and Yuzhong Qu. MiCRon: making sense of news via relationship subgraphs. In *CIKM*, pages 2901–2904, 2019.

[Kasneci *et al.*, 2009a] Gjergji Kasneci, Shady Elbassuoni, and Gerhard Weikum. MING: mining informative entity relationship subgraphs. In *CIKM*, pages 1653–1656, 2009.

[Kasneci *et al.*, 2009b] Gjergji Kasneci, Maya Ramanath, Mauro Sozio, Fabian M. Suchanek, and Gerhard Weikum. STAR: steiner-tree approximation in relationship graphs. In *ICDE*, pages 868–879, 2009.

[Kharlamov *et al.*, 2017a] Evgeny Kharlamov, Dag Hovland, Martin G. Skjæveland, Dimitris Bilidas, Ernesto Jiménez-Ruiz, Guohui Xiao, Ahmet Soylu, Davide Lanti, Martin Rezk, Dmitriy Zheleznyakov, Martin Giese, Hallstein Lie, Yannis E. Ioannidis, Yannis Kotidis, Manolis Koubarakis, and Arild Waaler. Ontology Based Data Access in Statoil. *J. Web Semant.*, 44:3–36, 2017.

[Kharlamov *et al.*, 2017b] Evgeny Kharlamov, Theofilos Mailis, Gulnar Mehdi, Christian Neuenstadt, Özgür L. Özçep, Mikhail Roshchin, Nina Solomakhina, Ahmet Soylu, Christoforos Svingos, Sebastian Brandt, Martin Giese, Yannis E. Ioannidis, Steffen Lamparter, Ralf Möller, Yannis Kotidis, and Arild Waaler. Semantic access to streaming and static data at Siemens. *J. Web Semant.*, 44:54–74, 2017.

[Lehmann *et al.*, 2015] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, and Christian Bizer. DBpedia - A large-scale, multilingual knowledge base extracted from Wikipedia. *Semant. Web*, 6(2):167–195, 2015.

[Li *et al.*, 2020] Shuxin Li, Gong Cheng, and Chengkai Li. Relaxing relationship queries on graph data. *J. Web Semant.*, In Press:1–13, 2020.

[Lin *et al.*, 2018] Allen Yilun Lin, Joshua Ford, Eytan Adar, and Brent J. Hecht. VizByWiki: Mining data visualizations from the web to enrich news articles. In *WWW*, pages 873–882, 2018.

[Mendes *et al.*, 2011] Pablo N. Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. DBpedia Spotlight: shedding light on the web of documents. In *I-SEMANTICS*, pages 1–8, 2011.

[Ni *et al.*, 2016] Yuan Ni, Qiong Kai Xu, Feng Cao, Yosi Mass, Dafna Sheinwald, Huijia Zhu, and Shao Sheng Cao. Semantic documents relatedness using concept graph representation. In *WSDM*, pages 635–644, 2016.

[Ristoski *et al.*, 2019] Petar Ristoski, Jessica Rosati, Tommaso Di Noia, Renato De Leone, and Heiko Paulheim. RDF2Vec: RDF graph embeddings and their applications. *Semant. Web*, 10(4):721–752, 2019.

[Schuhmacher and Ponzetto, 2014] Michael Schuhmacher and Simone Paolo Ponzetto. Knowledge-based graph document modeling. In *WSDM*, pages 543–552, 2014.

[Shen *et al.*, 2015] Wei Shen, Jianyong Wang, and Jiawei Han. Entity linking with a knowledge base: Issues, techniques, and solutions. *IEEE Trans. Knowl. Data Eng.*, 27(2):443–460, 2015.

[Tang *et al.*, 2017] Yixuan Tang, Weilong Huang, Qi Liu, Anthony K. H. Tung, Xiaoli Wang, Jisong Yang, and Beibei Zhang. QALink: Enriching text documents with relevant q&a site contents. In *CIKM*, pages 1359–1368, 2017.

[Tong and Faloutsos, 2006] Hanghang Tong and Christos Faloutsos. Center-piece subgraphs: problem definition and fast solutions. In *SIGKDD*, pages 404–413, 2006.

[Tonon *et al.*, 2016] Alberto Tonon, Michele Catasta, Roman Prokofyev, Gianluca Demartini, Karl Aberer, and Philippe Cudré-Mauroux. Contextualized ranking of entity types based on knowledge graphs. *J. Web Semant.*, 37-38:170–183, 2016.

[Viswanathan and Ilango, 2012] V. Viswanathan and Krishnamurthi Ilango. Ranking semantic relationships between two entities using personalization in context specification. *Inf. Sci.*, 207:35–49, 2012.

[Zhu and Iglesias, 2017] Ganggao Zhu and Carlos Angel Iglesias. Computing semantic similarity of concepts in knowledge graphs. *IEEE Trans. Knowl. Data Eng.*, 29(1):72–85, 2017.