Graph Embeddings Meet Link Keys Discovery for Entity Matching

Chloé Khadija Jradeh
khadija.jradeh@irit.fr
IRIT
Toulouse 1 Capitole University
Toulouse, France

Ensiyeh Raoufi*
ensiyeh.raoufi@lirmm.fr
LIRMM
University of Montpellier
Montpellier, France

Jérôme David†‡
jerome.david@inria.fr
Grenoble INP, LIG
Grenoble Alpes University
Grenoble, France

Pierre Larmande§
pierre.larmande@ird.fr
DIADE
University of Montpellier
Montpellier, France

François Scharffe¶
francois.scharffe@lirmm.fr
LIRMM
University of Montpellier
Montpellier, France

Konstantin Todorov‖
konstantin.todorov@lirmm.fr
LIRMM
University of Montpellier
Montpellier, France

Cassia Trojahn**††
cassia.trojahn-dos-santos@univ-grenoble-alpes.fr
Grenoble INP, LIG
Grenoble Alpes University
Grenoble, France

Abstract

Entity Matching (EM) automates the discovery of identity links between entities within different Knowledge Graphs (KGs). Link keys are crucial for EM, serving as rules allowing to identify identity links across different KGs, possibly described using different ontologies. However, the approach for extracting link keys struggles to scale on large KGs. While embedding-based EM methods efficiently handle large KGs they lack explainability. This paper proposes a novel hybrid EM approach to guarantee the scalability link key extraction approach and improve the explainability of embedding-based EM methods. First, embedding-based EM approaches are used to sample the KGs based on the identity links they generate, thereby reducing the search space to relevant sub-graphs for link key extraction. Second, rules (in the form of link keys) are extracted to explain the generation of identity links by the embedding-based methods. Experimental results demonstrate that the proposed approach allows link key extraction to scale on large KGs, preserving the quality of the extracted link keys. Additionally, it shows that link keys can improve the explainability of the identity links generated by embedding-methods, allowing for the regeneration of 77% of the identity links produced for a specific EM task, thereby providing an approximation of the reasons behind their generation.

CCS Concepts

• **Information systems** → **Entity resolution**.

Keywords

Entity matching, Knowledge graphs, Link keys, Embedding-based EM, Symbolic EM, Graph embeddings, Language models, Hybrid AI

ACM Reference Format:
Chloé Khadija Jradeh, Ensiyeh Raoufi, Jérôme David, Pierre Larmande, François Scharffe, Konstantin Todorov, and Cassia Trojahn. 2025. Graph Embeddings Meet Link Keys Discovery for Entity Matching. In *Proceedings of the ACM Web Conference 2025 (WWW '25), April 28-May 2, 2025, Sydney, NSW, Australia.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3696410.3714581

1 Introduction

Knowledge Graphs (KGs) offer an explicit representation of knowledge and have emerged as powerful tools for a range of applications, including recommendation systems, question answering, medical applications and data federation [23]. The distributed nature of data across multiple KGs rises different challenges, including addressing the task of Entity Matching (EM). This task involves automatically identifying the identity links between different KGs, which consist of entities from different KGs and referring to the same real-world object. For addressing the task of EM, *key-based approaches* involve the explicit definition or extraction of *keys* [35], which uniquely identify equivalent entities across multiple KGs. An example of a key is ({creator, title} key Work), indicating that when two entities of the class Work share values for the properties creator and title then they denote the same entity.

*Also with CNRS.
†Also with CNRS.
‡Also with Inria.
§Also with IRD.
¶Also with CNRS.
‖Also with CNRS.
**Also with CNRS.
††Also with Inria.

WWW '25, Sydney, NSW, Australia

ACM ISBN 979-8-4007-1274-6/25/04
https://doi.org/10.1145/3696410.3714581

To perform EM with keys, the KGs must be described using the same ontology, or their ontologies must be aligned. In order to overcome this limitation, keys have been generalised as *link keys* [5]. An example of a link key is as follows:

$$(\{\langle \text{author}, \text{auteur}\rangle, \langle \text{title}, \text{titre}\rangle\} \text{ linkkey } \langle \text{NonFiction}, \text{Essai}\rangle)$$

stating that whenever an entity of the class NonFiction and an entity of the class Essai, share values for roles author and auteur, and for roles title and titre, respectively, then they denote the same entity. The automatic extraction of link keys can be solely realised using Linkex [5]. However, due to its exhaustive nature, Linkex extracts all potential link key candidates generated from the entire input KGs, making it difficult to scale on large KGs. Key-based approaches, including link keys, encompass properties and classes that can be reused within a given domain. They can be combined with ontologies and ontology alignments to profit from logical reasoning.

With the rise of deep learning, there has been an increased adoption of *embedding-based* methods which automatically learn and extract features from KGs [16, 21]. Embedding-based EM models employ representation learning for EM across different KGs using reference sets of identity links among these KGs. The embedding module portrays each KG entity and relation as a vector in a lower-dimensional space. Consequently, embedding-based EM methods scale better on large and cross-lingual KGs, but lack explainability for the produced results, i.e., the reasons behind their generation. To address this issue, our proposed approach, HMatch, combines the strengths of both embedding-based methods and key-based approaches, aiming to achieve both scalability and explainability in EM. HMatch employs embedding-based methods such as BERT-INT [45] to establish identity links between the given pair of KGs. These identity links are used to sample the KGs, retaining only the sub-graphs necessary for extracting link keys. After, Linkex is deployed on these sub-graphs to extract link keys, which present explainable rules and can be reused for other EM task, even if no training data is available. For example, consider three large KGs within the same domain: KG_1, KG_2, and KG_3. While a reference set of identity links exists between KG_1 and KG_2, no such set exists between KG_2 and KG_3. Due to the scale of these KGs, Linkex cannot be directly applied to KG_1 and KG_2. However, using HMatch, we can first train embedding models like BERT-INT on the reference set of identity links to extract identity links between KG_1 and KG_2. These identity links can then be used to sample the KGs into sub-graphs, making it feasible to apply Linkex on these sub-graphs and extract link keys. Since KG_1, KG_2, and KG_3 belong to the same domain, the extracted link keys can be reused to construct identity links between KG_2 and KG_3 without the need to re-launch Linkex, even in the absence of a direct reference set of identity links, which prevents the application of BERT-INT.

Our approach allows, as well, for the extraction of sets of link keys explaining the generation of the identity links by the embedding-based approaches. These sets of link keys can be reused to reconstruct identity links by verifying which entities share values for the property pairs specified in the link keys.

The main contributions of this paper are: (a) a novel approach that combines embedding-based and key-based EM methods; (b) the reduction of Linkex's search space through the use of identity links generated by embedding-based techniques, guaranteeing its scalability; and (c) improved explainability of the identity links produced by embedding-based techniques through the use of link keys.

The paper is structured as follows. Section 2 introduces the background, while Section 3 presents the proposed framework. Section 4 details the space-reduction and explainability experiments. Section 5 analyzes and discusses the results. Section 6 reviews related work, and finally, Section 7 summarizes the contributions and outlines directions for future research.

2 Background Definitions

This section provides the formal definitions of KGs, EM and link keys.

Definition 2.1. A **knowledge graph** (KG) comprises a set of triples $\{(s, p, o)\}$, where each triple (s, p, o) is composed of a **subject** s, which is an entity representing a real-world object, a **property** p, which is a property or an attribute that describes the nature of the connection between the subject and the object, and an **object** o, which can either be an entity or an attribute value.

A KG can be accompanied by an **ontology** that defines the classes of the entities, the attributes, and the properties represented in the triples.

Definition 2.2. Let KG_1 and KG_2 be respectively a pair of source and target KGs. The **entity matching** task involves finding a set of identity links $L = \{(x_i \; owl{:}sameAs \; y_j)\}$, where $x_i \in KG_1$ and $y_j \in KG_2$, such that x_i and y_j refer to the same real-world entity. Each identity link (x_i owl:sameAs y_j) can be associated with a score s_{ij} indicating the confidence that x_i and y_j are the same entity. The identity links L represent the matching pairs of entities between the two KGs.

To address the task of EM, the concept of link keys has been introduced.

Definition 2.3. A **link key** between a pair of KGs KG_1 and KG_2 is an expression of the form:

$$\langle \{\langle \mathsf{P}_i, \mathsf{P}'_i\rangle\}_{i\in \mathsf{EQ}}, \{\langle \mathsf{Q}_j, \mathsf{Q}'_j\rangle\}_{j\in \mathsf{IN}} \text{ linkkey } \langle \mathsf{C}, \mathsf{D}\rangle\rangle, ^{1}$$

where: $\langle \mathsf{C}, \mathsf{D}\rangle$ is a pair of classes of the entities belonging, respectively, to KG_1 and KG_2, $\{\langle \mathsf{P}_i, \mathsf{P}'_i\rangle\}_{i\in \mathsf{EQ}}$ and $\{\langle \mathsf{Q}_j, \mathsf{Q}'_j\rangle\}_{j\in \mathsf{IN}}$ are sets of property pairs such that $\mathsf{P}_i, \mathsf{Q}_i$ belongs to KG_1 and $\mathsf{P}'_i, \mathsf{Q}'_i$ belongs to KG_2. The link key asserts that if two entities, belonging respectively to classes C and D, share all values for the properties $\{\langle \mathsf{P}_i, \mathsf{P}'_i\rangle\}$ and at least one value for each pair of properties $\{\langle \mathsf{Q}_j, \mathsf{Q}'_j\rangle\}$, then they are considered identical.

3 HMatch: a Hybrid Approach for EM

This section introduces HMatch.[2]. The approach consists of two components: (1) scaling component for link key extraction and (2) explainability component for embedding-based EM. In that way, HMatch acts in both ways on the interface between key and

[1] In this paper, we focus only on in-link keys, i.e., link keys with only the set of properties $\{\langle \mathsf{Q}_j, \mathsf{Q}'_j\rangle\}$.

[2] https://github.com/DACE-DL/HMatch/

embedding-based EM methods. We detail on each of the two components below.

The first one (depicted in Figure 2 in the appendix) aims to ensure the scalability of Linkex by reducing its search-space. Given a pair of a source and a target KGs, KG_1 and KG_2, along with a reference set of identity links between them, an embedding-based method (subject of choice) is applied to generate a set of identity links. The identity links whose score exceeds the one specified by the user are given to the sampling process, along with the original KGs. Then for each graph, the sampling process selects only the triples that refer to an entity occurring in a generated identity link. More specifically, given an identity link x owl:sameAs y, the process iterates over each triple in KG_1 and KG_2 to select triples whose subject matches the value of x and y, respectively. The selected triples form a new pair of KGs, KG'_1 and KG'_2, which are sub-graphs of KG_1 and KG_2, respectively. Next, Linkex is launched on the sampled KGs, KG'_1 and KG'_2 to output a set of link keys.

The second component aims to improve the explainability of the identity links given by embedding-based methods (depicted in Figure 3 in the appendix). The identity links produced by the embedding-based methods and whose score exceeds the one specified by the user, along with the sampled KGs are provided to Linkex. Linkex in turn outputs sets of link keys, allowing to regenerate the provided identity links, thereby explaining their entailment.

We now introduce the tools and models used in the framework. The embedding-based methods are TransEdge [42] and BERT-INT [45]. These state-of-the-art embedding-based EM models, have different foundations, and a notable performance compared to methods with similar frameworks on benchmark KGs [26, 42, 49]. We chose these two methods because TransEdge is entirely based on the graph structure and only uses the object properties, while BERT-INT mostly uses attribute values and is one of a few methods that use almost all literals and descriptions of the entities for EM. The performance of both methods is significantly better than those of their peers. To extract link keys we use Linkex, which is the sole tool capable of performing link key extraction.

BERT-INT. BERT-based Interaction Model for KG alignment [45] is an approach leveraging Bidirectional Encoder Representations from Transformers (BERT) [15] to tackle cross-lingual understanding and transfer learning tasks. This model uses a pre-trained multilingual BERT-based model to comprehend and represent text across different languages. Due to training on a large corpus of diverse and unlabeled text data, a pre-trained BERT is a language model specifically designed to acquire a deep understanding of language semantics and syntax, capturing contextual information within natural language. BERT-INT efficiently processes and comprehends the multilingual content of KG entities. BERT-INT initially embeds the attribute values of entities across the two KGs using BERT CLS embedding into a multi-lingual embedding space. Then, considering similarity matrices, it computes the interactions between the attributes and neighbors of each pair of entities. Finally, for the task of EM, the model uses a Multi-Layer Perceptron [36] to minimize the distance between the aligned entities in the embedding space. BERT-INT has achieved the best results so far on the DBP15K KGs [49] that are widely used for evaluating EM systems.

TransEdge. [42] embeds the KGs based on the translational KG embedding technique TransE [10]. TransE is founded on the notion that relationships between entities can be represented as translations in the embedding space, i.e. a relation predicate is a translation vector between the head and tail entity. However, TransEdge is an edge-centric model that distinguishes how a relation predicate is represented based on different contexts of entities holding that relation. Hence, using TransEdge KG embedding, relation predicates would have different contextualized representations according to a variety of contexts of their head-tail entity pairs. Furthermore, to address the challenge of insufficient aligned entities across the two KGs in each dataset, the approach employs a bootstrapping strategy [41] to augment the input data. This involves generating additional likely-aligned entity pairs by resampling from the existing data, thereby enhancing the representation of aligned entities and improving the matching model's performance.

Linkex. [1] uses a two-step process to extract link keys from two KGs. First, it indexes triples from each KG in hash tables, where keys represent objects and values represent pairs of subject-properties. Then, it finds common keys in both indexes to create a third index. This third index links each pair of subjects with its maximal set of shared properties. These shared properties are used to build a concept lattice, where each concept represents a candidate link key. The concept lattice materializes the partial order (subsumption) relationship between link key candidates, thus facilitating their selection. To facilitate the explainability of a given set of identity links, Linkex allows to extract subsets of link keys which maximizes the coverage of the given set of identity links by adopting the approach described in [6], specifically employing the "expand-best strategy". This strategy operates as a best-first search, systematically expanding the best combination of link keys based on an evaluation measure. The evaluation methods implemented are: (1) the minimum between precision and recall and (2) the f-measure. The first measure forces the algorithm to optimize the worst-case scenario between precision and recall, while the second measure prioritizes a balanced compromise. Using the link key lattice, the algorithm selectively considers anti-chains, the minimal sets of link keys concerning the subsumption relation. Finally, candidate link keys can be filtered using quality estimation measures from [5]. When the set of reference identity links is available, link keys can be evaluated using precision and recall, which measure the accuracy and completeness of the links generated by the candidate link keys. Let L^+ be a set of `owl:sameAs` links (positive examples) and L_c the links generated by a link key candidate c. The precision and recall of the link key candidate c with respect to L^+:

$$\text{Precision} = \frac{|L^+ \cap L_c|}{|L_c|} \quad \text{Recall} = \frac{|L^+ \cap L_c|}{|L^+|}$$

In summary, HMatch combines embedding-based methods with link key extraction to improve respectively their explainability and efficiency. It ensures the scalability of Linkex by reducing the search space using embedding-based identity links, while enhancing explainability by generating link keys that explain the derived identity by embedding-based methods. This integrated approach uniquely addresses the challenges of scalability and explainability for EM tasks.

4 Experiments

This section details the two types of experiments performed to test the two components of our approach.

Data sets. The approach has been evaluated on the DBP15K KGs [40] and the Memory Alpha-Star Trek Expanded Universe [3] KGs (referred to as Memory-alphaSTE).

DBP15K KGs. The DBP15K KGs, extracted from DBpedia [28], are widely used as benchmarks for EM tasks [40, 42, 45]. Available in English (En.), French (Fr.), Japanese (Ja.), and Chinese (Zh.), each dataset contains about 40K entities. Inheriting DBpedia's ontology structure, they include various types such as Person, Place, and Organization, and properties describing attributes like names, dates, and connections among entities (e.g., "president," "predecessor"). In each experiment, the En. KG was the source KG, tested respectively with Fr., Ja., and Zh. target KGs. Each KG pair has a reference set of identity links compromising 30K entities (15K per KG), with 30% used as the training set for each embedding-based EM model.

Memory-alphaSTE KGs. The Memory-alphaSTE KGs, part of the well known OAEI campaign's KG track, are derived from Memory Alpha, a collaborative Star Trek encyclopedia. The Star Trek franchise encompasses multiple television series, films, novels, games, and collectibles. The source KG, Memory Alpha, has around 250K entities and 180 relations, while the target Star Trek Expanded KG has about 55K entities and 130 relations. The reference set of identity links includes 3,560 entities, with 1,779 entities in each KG. These KGs were selected to demonstrate the approach's capability to ensure the scalability of Linkex, which initially struggled to scale on them.

Parameters. For each of the experiments performed, to ensure high-quality identity links, we select identity links whose scores surpass a specific threshold. Additionally, considering the size of the KGs, the support threshold of properties used by Linkex varies across experiments. These parameters used in the experiments are indicated in Table 1.

Tools. The tools used in the experiments were installed from the following links: HMatch: https://github.com/DACE-DL/HMatch/, Linkex: https://gitlab.inria.fr/moex/linkex, BERT-INT: https://github.com/kosugi11037/bert-int/tree/master/interaction_model, TransEdge: https://github.com/nju-websoft/TransEdge/tree/master/code.

4.1 Space-Reduction Experiments

Experimental setting. For each of the KGs, three experiments have been conducted:

(#1) Linkex on the original DBP15K/Memory-alphaSTE KGs (baseline),
(#2) Linkex on DBP15K/Memory-alphaSTE KGs sampled using BERT-INT,
(#3) Linkex on DBP15K/Memory-alphaSTE KGs sampled using TransEdge.

The quality of the link keys obtained in each task and for each of the experiments performed on the DBP15K KGs is shown in Table 2. We calculated the precision, recall and f-measure of the extracted link keys on original KGs using the reference sets of identity links. This allows to compare the quality of the link keys extracted from the sampled and original KGs. Table 2 displays the average precision, recall, and f-measure for the top 10 link keys sorted by f-measure. We have chosen the average of the first 10 link keys to demonstrate the effect of over-sampling. However, for the En. & Fr. task, the first 3 link keys — whether from the original KGs or the KGs sampled by BERT-INT or TransEdge — have an average f-measure of 0.58. Table 4 presents the variation of runtime across each experiment.

4.1.1 *Launching Linkex on DBP15K KGs.*

Original KGs (Exp. #1). Due to the size of the original DBP15K KGs, Linkex was not able to run on the original DBP15K KGs and a support threshold has been set to 0.1 meaning that only properties instantiated on at least 10% of instances are considered by Linkex. As shown in Table 2, running Linkex on the original En. & Fr. KGs revealed an average quality of link keys, indicated by f-measure score of 0.48 with a recall of 0.33 but a high precision of 0.88. In contrast, when Linkex was applied to the En. & Ja.\ Zh. tasks, it produced link keys of notably poor quality. This issue arises from Linkex's inability to handle languages that use different alphabets, resulting in infrequent agreement of property values in the En. and the Ja.\Zh. KGs.

The embedding-based models BERT-INT and TransEdge are now used to extract identity links for sampling the DBP15K KGs. For each model, the size of the KGs was reduced as indicated in Figure 1. Due to the relatively still large size of the KGs after the sampling (especially when the identity links are produced by BERT-INT), we run Linkex restricting its support threshold to 0.1.

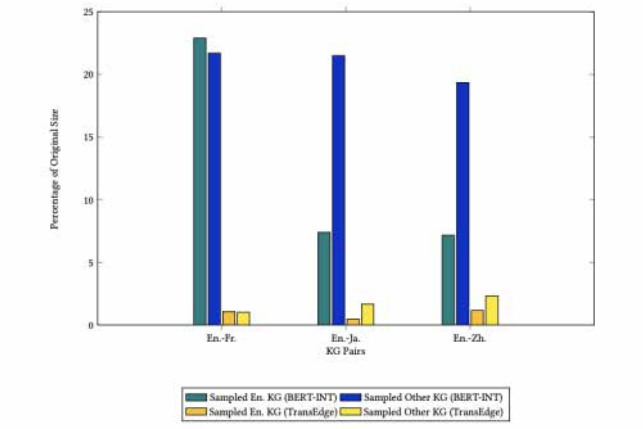

Figure 1: Reduction of KG sizes for different language pairs using BERT-INT and TransEdge

Sampled KGs using BERT-INT (Exp. #2). As shown in Table 2, running Linkex on the sampled En. & Fr. KGs preserves the quality of the extracted link keys. For the En. & Ja.\ Zh. tasks, the quality of the link keys was slightly improved. This improvement is due to the sampling process, which removed information resulting in generating link keys based on false-positive agreements between the property values of non-equivalent individuals.

[3] https://oaei.webdatacommons.org/tdrs/testdata/persistent/knowledgegraph/v4/knowledgegraph_v4.zip

Experiment Type	KGs	Sampling Method	Identity Links Score	Support Threshold
Space-Reduction	DBP15K	Original KGs	-	0.1
		Sampled using BERT-INT	0.75 & 0.85	
		Sampled using TransEdge		
	MaSTE	Original KGs	-	0.7
		Sampled using BERT-INT	0.5	0
		Sampled using TransEdge		
Explainability	DBP15K	Sampled using BERT-INT	0.75	0
		Sampled using TransEdge		
	MaSTE	Sampled using BERT-INT	0.5	
		Sampled using TransEdge		

Table 1: Experimental Parameters.

Experiment	Task	precision	recall	f-measure
#1 (Original KGs)	En. & Fr.	0.88	0.33	**0.48**
	En. & Ja.	$3x10^{-4}$	$2x10^{-3}$	$3x10^{-4}$
	En. & Zh.	0.17	$9x10^{-3}$	$1x10^{-2}$
#2 (KGs sampled with BERT-INT)	En. & Fr.	0.88	0.33	**0.48**
	En. & Ja.	$5x10^{-2}$	$5x10^{-3}$	$1x10^{-3}$
	En. & Zh.	0.54	$1x10^{-2}$	$3x10^{-2}$
#3 (KGs sampled with TransEdge)	En. & Fr.	0.71	0.35	**0.45**
	En. & Ja. [4]	0	0	0
	En. & Zh.	0	0	0

Table 2: Comparison of the quality of link keys extracted from the original and the sampled DBP15K KGs using identity links with a score higher than 0.75.

Experiment	Task	precision	recall	f-measure
#1 (Original KGs)	En. & Fr.	0.88	0.33	**0.48**
	En. & Ja.	$3x10^{-4}$	$2x10^{-3}$	$3x10^{-4}$
	En. & Zh.	0.17	$9x10^{-3}$	$1x10^{-2}$
#2 (KGs sampled with BERT-INT)	En. & Fr.	0.76	0.35	**0.46**
	En. & Ja.	$1x10^{-3}$	$5x10^{-3}$	$1x10^{-3}$
	En. & Zh.	0.44	$1x10^{-2}$	$2x10^{-2}$
#3 (KGs sampled with TransEdge)	En. & Fr.	0.8	0.16	**0.25**
	En. & Ja.	0	0	0
	En. & Zh.	0	0	0

Table 3: Comparison of the quality of link keys extracted from the original and the sampled DBP15K KGs using identity links with a score higher than 0.85.

Sampled KGs using TransEdge (Exp. #3). The number of identity links returned by TransEdge with a score higher than 0.75 is very small compared to the ones returned by BERT-INT, resulting in a huge reduction in the size of KGs. The quality of link keys have slightly decreased for all tasks due to over-fitting caused by the substantial reduction in the size of the KGs.

As shown in Table 3, the quality of the extracted link keys when sampling is performed using identity links with a score higher than 0.85 is lower than that when sampling is performed using a score higher than 0.75. This is because the large reduction in the search space prevents Linkex from extracting high-quality link keys where the over-fitting phenomenon is more evident.

However, when sampling is performed using BERT-INT the quality of link keys is slightly lower than that one of the link keys extracted from the original KGs and the KGs sampled with a score higher than 0.75. Since the number of identity links with a score above 0.85 is slightly higher than the ones with a score 0.75, this leads to a similar reduction in the search space. Based on Table 4, employing BERT-INT or TransEdge for sampling KGs and using them instead of the original ones for link keys extraction decreases Linkex's runtime. As expected, the higher sampling score used, the smaller the sampled KGs are, and the lower Linkex's runtime is.

4.1.2 Launching Linkex with Memory-alphaSTE KGs.

This experiment focuses on demonstrating how sampling enables Linkex to scale on very large KGs.

Original KGs (Exp. #1). Due to the large size of the original KGs, Linkex could not run without setting a high support threshold (0.7). However, this resulted in no link key being produced.

		Score of the identity links: **0.75**		Score of the identity links: **0.85**	
Sampling Method	**Original**	**BERT-INT**	**TransEdge**	**BERT-INT**	**TransEdge**
Exp.	#1	#2	#3	#2	#3
En. & Fr.	27.01	16.63	1.98	5.93	0.64
En. & Ja.	22.53	8.60	2	9.29	0.73
En. & Zh.	26.06	12.03	2.9	10.69	0.84

Table 4: Variation of runtime (in seconds) across the experiments performed on the different task of the DBP15K KGs.

Experiment	precision	recall	f-measure	runtime
#1 (Original KGs)	-	-	-	7.25
#2 (KGs sampled with BERT-INT)	0.55	0.83	0.66	4.99
#3 (KGs sampled with TransEdge)	0.58	0.8	0.67	3.55

Table 5: Comparison of the quality of link keys and Linkex runtime (in seconds) using original and sampled memory-alphaSTE KGs

Sampled KGs using BERT-INT(Exp. #2) and TransEdge (Exp. #3). To ensure a fair comparison between the following experiments, we use for sampling a score of 0.5, which is the highest score allowing to retrieve identity links between TransEdge and BERT-INT models. The results showing the quality of the extract link keys are shown in Table 5. Using BERT-INT, the size of memory-alpha and STE KGs was respectively reduced to 3.36% and to 6.92% of their original size. Using TransEdge, memory-alphaSTE KGs were respectively reduced to 2.23% and to 4.23% of their original size.

Sampling with BERT-INT or TransEdge enables Linkex to scale on large KGs, and results in extracting link keys with high recall.

Sampling with the identity links produced by TransEdge, particularly, results in higher percentage of KGs reduction and consequently lower runtime for Linkex. This, however, results in extracting better quality link keys compared to those extracted from the KGs sampled with BERT-INT.

4.2 Explainability Experiments

In these experiments, Linkex is used to extract the sets of links keys which explains the identity links produced by BERT-INT and TransEdge on DBP15K and Memory-alphaSTE KGs.

Experimental setting. For the DBP15K KGs, we only consider the En. & Fr. task. Results for the En. & Ja. \Zh. tasks are omitted due to the poor quality of the generated link keys, as discussed in Section 4.1. The displayed sets of link keys in Tables 6 and 7 have the best recall among those calculated optimizing the worst between precision and recall. We use this strategy since the precision of the extracted link keys is high on the considered KGs and we seek to maximise the recall allowing to cover more identity links.

4.2.1 Identity Links generated on DBP15K KGs. We choose the identity links with scores greater than 0.75. This allows to extract link keys that explain the most accurate identity links excluding those that result in the extraction of misleading link keys. The results for BERT-INT and TransEdge are presented in Tables 6 and 7, respectively. The prefixes used in the following tables are:

- dbp: ⟨http://[en-fr].dbpedia.org/property/⟩,
- foaf: ⟨http://xmlns.com/foaf/0.1/⟩.

BERT-INT. The set of link keys presented in Table 6 has the highest recall of **0.77** among all the other sets generated. This indicates that it can regenerate 77% of the identity links produced by BERT-INT for the En. and Fr. task. Additionally, this set has a high precision of **0.77**. To further investigate the ability of this set of link keys to generate identity links missed by BERT-INT, we examined which entities from the original KGs could be linked by this set and were able to regenerate, among other identity links, 3,036 correct identity links (approximately 20% of the reference set of identity links) that BERT-INT did not produce. Thus, this set of link keys not only explains the identity links produced by BERT-INT but also complements it by generating additional identity links that BERT-INT misses, improving both the its coverage and explainability.

TransEdge. The link key set displayed in Table 7 has a recall of **0.6**, allowing to cover 60% of the identity links produced by TransEdge. It has a precision of **0.8**. Additionally, this set of link keys also allows for the regeneration of 3,650 correct identity links (approximately 24% of the reference set), among other identity links, which were not generated by TransEdge.

4.2.2 Identity Links generated on Memory-alphaSTE KGs. We restrict ourselves to the identity links with a score greater than 0.5 as it is the highest common score between the identity links generated by BERT-INT and TransEdge on Memory-alphaSTE KGs. The prefixes used in Tables 9 and 8 are:

- rdfs: ⟨http://www.w3.org/2000/01/rdf-schema#⟩,
- ma: ⟨http://dbkwik.webdatacommons.org/memory-alpha.wikia.com/property/⟩,

Link Keys
{(foaf:name, foaf:name)}
{(dbp:birthDate, ns1:dateDeNaissance)}
{(dbp:name, ns1:nom)}
{(foaf:name, ns1:nom)}
{(dbp:name, ns1:titre)}
{(dbp:deathDate, ns1:dateDeDécès)}
{(dbp:length, ns1:durée), (dbp:released, ns1:sorti)}
{(dbp:name, foaf:name)}
{(dbp:title, foaf:name), (dbp:title, ns1:nom)}
{(foaf:name, ns1:titre), (dbp:title, ns1:titre)}
{(dbp:released, ns1:sorti), (dbp:title, ns1:titre)}
{(dbp:deathDate, ns1:jusqu′auFonction), (dbp:years, ns1:àPartirDuFonction)}
{(dbp:termEnd, ns1:jusqu′auFonction), (dbp:termStart, ns1:àPartirDuFonction), (dbp:years, ns1:nom)}
{(dbp:title, ns1:nom), (dbp:title, ns1:titre)}
{(dbp:termEnd, ns1:dateDéDécès), (dbp:termEnd, ns1:jusqu′auFonction), (dbp:termStart, ns1:àPartirDuFonction)}
{(dbp:termEnd, ns1:jusqu′auFonction), (dbp:termStart, ns1:àPartirDuFonction), (dbp:years, ns1:àPartirDuFonction)}
{(dbp:length, ns1:durée), (dbp:title, foaf:name), (dbp:title, ns1:titre)}

Table 6: The best-recall set of link keys explaining the identity links given by BERT-INT on the En. & Fr. task.

Link Keys
{(foaf:name, foaf:name)}
{(dbp:birthDate, dbp:dateDeNaissance)}
{(dbp:name, dbp:nom)}
{(dbp:name, dbp:titre)}
{(dbp:founded, foaf:cration)}
{(foaf:deathDate, ns1:dateDeDécès)}
{(dbp:title, dbp:titre)}
{(foaf:name, dbp:nom)}
{(dbp:termStart, dbp:àPartirDuFonction)}
{(dbp:titre, dbp:nom)}
{(dbp:years, dbp:àPartirDuFonction)}

Table 7: The best-recall set of link keys explaining the identity link of TransEdge on the En. & Fr. task.

- st: ⟨http://dbkwik.webdatacommons.org/stexpanded.wikia.com/property/⟩,
- skos: ⟨http://www.w3.org/2004/02/skos/core#⟩,
- dcm: ⟨http://dbkwik.webdatacommons.org/ontology/⟩.

BERT-INT. According to Table 8, the produced set of link keys shows an average precision of **0.58** and recall of **0.56**, i.e., it allows to regenerate 56% of the identity links produced by BERT-INT. Additionally this set of link keys enable to cover other, among others, 500 correct identity links (approximately 28% of the reference set of identity links) that BERT-INT misses.

TransEdge. According to Table 9, the set of link keys shows an average level of precision of **0.5** and recall of **0.5**. Additionally this

Link Keys
{(rdfs:label, rdfs:label)}
{(skos:altLabel, skos:altLabel)}
{(dcm:wikiPageWikiLinkText, st:className)}
{(ma:imagecap, dcm:wikiPageWikiLinkText), (ma:imagecap, rdfs:label) rdfs:label), (ma:imagecap, skos:altLabel), (dcm:wikiPageWikiLinkText, dcm:wikiPageWikiLinkText), (rdfs:label, dcm:wikiPageWikiLinkText), (skos:altLabel, dcm:wikiPageWikiLinkText)}
{(ma:dt, dcm:wikiPageWikiLinkText), (ma:dt, st:name), (ma:name, dcm:wikiPageWikiLinkText), (ma:name, st:name), (dcm:wikiPageWikiLinkText, dcm:wikiPageWikiLinkText), (dcm:wikiPageWikiLinkText, st:name), (dcm:wikiPageWikiLinkText, rdfs:label), (dcm:wikiPageWikiLinkText, skos:altLabel)}

Table 8: The best-recall set of link keys explaining the identity link of BERT-INT on memory-alphaSTE KGs.

Link Keys
{(dcm:wikiPageWikiLinkText, rdfs:label), (dcm:wikiPageWikiLinkText , skos:altLabel), (rdfs:label, rdfs:label), (rdfs:label, skos:altLabel), (skos:altLabel, rdfs:label), (skos:altLabel, skos:altLabel)}
{(ma:armament, st:weapons), (dcm:wikiPageWikiLinkText, st:className)}
{(dcm:wikiPageWikiLinkText, dcm:wikiPageWikiLinkText), (rdfs:label, dcm:wikiPageWikiLinkText)(skos:altLabel, dcm:wikiPageWikiLinkText)}

Table 9: The best-recall set of link keys explaining the identity link of TransEdge on memory-alphaSTE KGs.

Model	precision	recall	f-measure
BERT-INT	0.95	0.65	0.77
TransEdge	0.95	0.03	0.07

Table 10: The precision, recall and f-measure of the identity links with a score higher than 0.5 produced by BERT-INT and TransEdge models.

set of link keys enable to cover other, among others, 1280 identity links (approximately 72% of the reference set of identity links) in the reference set of identity links.

To further investigate why the recall of the link keys sets shown in Tables 8 and 9 are not optimal, we calculated the precision and recall of these identity links against the reference set of identity links (Table 10). The inability of Linkex to produce link keys that adequately cover the considered identity links is due to the average recall of the identity links produced by BERT-INT and the extremely low recall of those produced by TransEdge. This latter factor also explains the ability of the set of link keys in Table 9 to generate many identity links not covered by TransEdge.

The capacity of the generated sets of link keys to generate identity links, which are missed by embedding-based methods, can be attributed to the fact that embedding-based approaches often prioritize structural similarities over exact attribute matches.

5 Discussion

Results: The experiments reveal promising results in the association of embedding-based methods (such as BERT-INT or TransEdge) and key-based methods (such as Linkex) for the task of EM. More specifically, the proposed approach allows to reduce of the task of link keys extraction from a pair of original KGs to a pair of sampled KGs. This guarantees the scalability of Linkex on large KGs and allows to significantly reduce its runtime. This reduction in runtime does not compromise the quality of the extracted link keys, provided that over-fitting is avoided. Additionally, the explainability of the identity links produced by BERT-INT can be approximated by associating a set of rules, represented as link keys.

Limitations: The explainability of identity links is currently limited to cases where there is a syntactic overlap between direct attribute values, making it difficult to provide clear interpretations for matches without such an overlap. This explains the room for improvement of the recall of the sets of link keys produced in the the explainability experiments. Enhancing recall could be achieved by extracting more expressive link keys, such as those that include inverse and composed properties, leading to new agreement between these properties. These link keys will allow to cover more identity links and thus augmenting the recall. Also for sampling the original KGs, the approach requires to have reference set of identity links to train the embedding-based methods, which in turn will output the identity links used in sampling. To sidestep this requirement in the training phase, semi-supervised or unsupervised EM approaches are preferred [11, 14, 24, 29, 30]. Besides, since the sampling process depends on the identity links generated by embedding-based methods, the choice of the embedding-based method and of the score of the identity links produced by embedding-based methods must be adequate. As the quality and number of the identity links used in the sampling affects the quality and the size of the sampled KGs. This in turns affects the quality of the extracted link keys and the runtime of Linkex. It is worth noting that even if Linkex initially demonstrates strong performance, its overall effectiveness is tied to the accuracy of the identity links used for sampling. For instance, Linkex outperforms BERT-INT on Doremus KGs [3], where the best link key achieves an f-measure of 0.804 compared to BERT-INT's 0.57. However, when sampling is performed using identity links produced by BERT-INT, the quality of the extracted link keys from the sampled KGs decreases due to the quality of the identity links.

Implications: The reduction in runtime achieved by sampling does not negatively impact the quality of the extracted link keys. Additionally, since link keys can be reused for other EM tasks, without necessitating training, this framework provides a balance between efficiency and reusability, which is essential for matching large KGs. Associating link keys for the identity links produced by embedding-based methods helps approximate the reasons behind the generation of the identity links, by providing the properties pairs allowing the regeneration of these identity links and thus providing more interpretability in the results, it allows as well to generate other correct identity links missed by embedding-based methods, which can prioritize structural similarities over attribute similarity.

6 Related Work

This section discusses key and embedding-based EM methods, along with approaches that provide explanations for these later.

Keys and link keys for EM. Different methods for key extraction have been proposed [2, 7, 35, 44]. In [35], an algorithm for extracting keys from KGs without necessitating a complete scan of the KGs is proposed. It identifies first maximal non-keys (i.e. properties combinations that share values for at least two entities). Then it derives minimal keys based on the discovered set of non-keys. However, [35] struggles to handle large KGs and necessitates data with no errors or duplicates. A scalable method for discovering *almost keys*, resilient against erroneous data, was developed in [44]. An *almost key* is a set of properties that is not a key due to a few exceptions. This method uses heuristics to identify keys and efficiently derive almost keys from non-keys, scaling effectively on large KGs. Another algorithm for extracting keys and *pseudo keys* has been proposed in [7]. Pseudo keys are keys that tolerate some exceptions. However, approaches such as [7, 44] can not deal with KGs described using different ontologies. Link keys overcome this challenge. An approach based on pattern structure for discovering link keys was presented in [1]. However, this approach still requires considering the entire KGs for building the candidate link keys. Our approach, on the other side, consists of sampling the KGs to remove entities that are irrelevant for link keys extraction. Other approaches such as [34] compare various blocking workflows and nearest-neighbor methods, focusing on performance trade-offs in EM. In contrast, our approach retains only the identical individuals, eliminating the need for complex blocking workflows and nearest-neighbor methods, and allows for the efficient production of link keys.

Embeddings for EM. Embedding-based approaches have been largely adopted in EM [17, 40, 43, 50]. They involve representing entities, relations, or other structured data in a continuous vector space [37, 47]. With a focus on relations between the entities, Translational KG embedding methods such as MTransE [12], IPTransE [51], and TransE [10] are well-known approaches that interpret relations as translation vectors operating on entity embeddings. Several entity alignment models such as [41] have been designed by using translational KG embedding techniques. To investigate the benefits of these relation-centric methods in link key extraction, we use the TransEdge model which learns KG embeddings through contextualized relation representations. More recently, pre-trained language models, like BERT [15], have been increasingly utilized for EM in KGs [33, 45]. Language models can learn embeddings that encode the semantic information of entities. To investigate benefits of using language models for EM, we used the BERT-INT model [45] that has been efficiently applied on many benchmark KGs [17]. EAGER [32] integrates graph embeddings and attribute similarities through machine learning to perform EM. While it achieves strong performance, particularly on rich KGs, it falls short in explainability due to the opaque nature of embeddings. Limited studies [13, 25] have explored the use of large language models (LLMs) in EM, but further research is needed to enable LLMs to generate identity links enriched with confidence scores, data types, and relational properties for better supporting evidence.

Explainability of embedding-based EM models. Recently, there has been a push to explain the mechanics and outputs of embedding-based models [4, 19, 20, 27]. There are two main approaches: prediction explanations and model explanations [22]. As an example, [22] creates *model* explanations for deep analysis of KG embedding models by extracting propositional features from a KG. In parallel, other studies have focused on explaining the *predictions* made by different embedding models [9, 18, 38, 39]. For instance, [38] proposes KELPIE (Knowledge graph Embeddings for Link Prediction: Interpretable Explanations) which explains a prediction by computing the subset of training facts enabling the model to return it, while [39] explains link prediction and triple classification using entity co-occurrence data. [8] enhances the explainability of link prediction methods in KGs by improving KELPIE [38]. It reduces candidate explanations, and improves explanation effectiveness using a semantic similarity measure. The studies done in [46, 48] relate more closely to our research and delve into explaining the model's predictions on EM tasks. I-Align [46] uses Transformer encoders to create an EM model that explains each alignment prediction, and Xin *et al.* [48] introduces a Transformer-based EM model with a comprehensive reasoning process to provide evidence for EM. Unlike transformer-based explainability methods for EM, our approach extracts rules in the form of link keys possibly composed of data and relation properties, optimally covering the generated identity links. Other methods, such as LightEA [31] provides explanations based on relational properties only.

7 Conclusion and Perspectives

This paper introduces a framework that combines embedding-based methods with Linkex, both addressing the EM problem. For an EM task involving a pair of KGs, the proposed framework utilizes the identity links produced by BERT-INT or TransEdge on a pair of source and target KGs to sample them before applying Linkex. This results in a notable reduction in Linkex's runtime while maintaining the quality of the extracted link keys, as long as over-fitting is avoided. Moreover, this approach enhances the scalability of Linkex for extracting link keys from large KGs. Last but not least, the framework enables the extraction of sets of link keys that cover the identity links generated by embedding-based methods, providing an approximation of the reasons behind their generation and thus enhancing the explainability of these methods' results.

Future work includes eliminating the need for the sampling phase by providing the identity links directly to Linkex which in turn restricts its search space to the entities contained in those links. Additionally for improving the recall of the extracted link keys we aim for extracting more expressive link keys, i.e., link keys including complex property constructors. Another direction is to compare our relational and data property link keys with those from LightEA [31] on benchmark datasets.

Acknowledgments

This work is partially supported by the French National Research Agency ANR DACE-DL project, grant number ANR-21-CE23-0019.

A Supplementary Figures

This appendix contains additional figures referenced in Section 3.

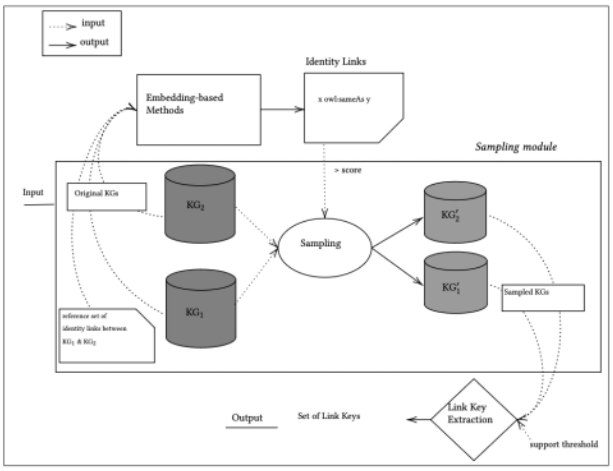

Figure 2: Space-reduction for link key extraction.

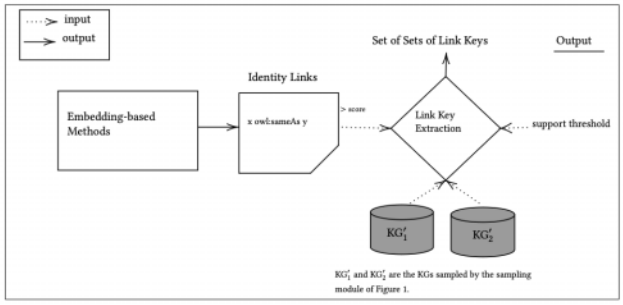

Figure 3: Explainability of the Identity Links.

References

[1] Nacira Abbas, Jérôme David, and Amedeo Napoli. 2020. Discovery of Link Keys in RDF Data Based on Pattern Structures: Preliminary Steps. In *CLA 2020 - The 15th International Conference on Concept Lattices and Their Applications (Proceedings of the 15th International Conference on Concept Lattices and Their Applications)*. Tallinn / Virtual, Estonia. https://hal.science/hal-02921643

[2] Manel Achichi, Mohamed Ben Ellefi, Danai Symeonidou, and Konstantin Todorov. 2016. Automatic Key Selection for Data Linking. 3–18. doi:10.1007/978-3-319-49004-5_1

[3] Manel Achichi, Pasquale Lisena, Konstantin Todorov, Raphaël Troncy, and Jean Delahousse. 2018. DOREMUS: A graph of linked musical works. In *The Semantic Web–ISWC 2018: 17th International Semantic Web Conference, Monterey, CA, USA, October 8–12, 2018, Proceedings, Part II 17*. Springer, 3–19.

[4] Sule Anjomshoae, Kary Främling, and Amro Najjar. 2019. Explanations of black-box model predictions by contextual importance and utility. In *Explainable, Transparent Autonomous Agents and Multi-Agent Systems: First International Workshop, EXTRAAMAS 2019, Montreal, QC, Canada, May 13–14, 2019, Revised Selected Papers 1*. Springer, 95–109.

[5] Manuel Atencia, Jérôme David, and Jérôme Euzenat. 2014. Data Interlinking through Robust Linkkey Extraction. In *Proceedings of the Twenty-First European Conference on Artificial Intelligence* (Prague, Czech Republic) *(ECAI'14)*. IOS Press, NLD, 15–20.

[6] Manuel Atencia, Jérôme David, and Jérôme Euzenat. 2019. Several Link Keys are Better than One, or Extracting Disjunctions of Link Key Candidates. In *Proceedings of the 10th International Conference on Knowledge Capture* (Marina Del Rey, CA, USA) *(K-CAP '19)*. Association for Computing Machinery, New York, NY, USA, 61–68. doi:10.1145/3360901.3364427

[7] Manuel Atencia, Jérôme David, and François Scharffe. 2012. Keys and Pseudo-Keys Detection for Web Datasets Cleansing and Interlinking. In *EKAW*.

[8] Roberto Barile, Claudia d'Amato, and Nicola Fanizzi. 2023. Explanation of Link Predictions on Knowledge Graphs via Levelwise Filtering and Graph Summarization. In *Proceedings of the AAAI Conference on Artificial Intelligence*. AAAI Press.

[9] Patrick Betz, Christian Meilicke, and Heiner Stuckenschmidt. 2022. Adversarial Explanations for Knowledge Graph Embeddings.. In *IJCAI*, Vol. 2022. 2820–2826.

[10] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems* 26 (2013).

[11] Jiaoyan Chen, Ernesto Jiménez-Ruiz, Ian Horrocks, Denvar Antonyrajah, Ali Hadian, and Jaehun Lee. 2021. Augmenting ontology alignment by semantic embedding and distant supervision. In *The Semantic Web: 18th International Conference, ESWC 2021, Virtual Event, June 6–10, 2021, Proceedings 18*. Springer, 392–408.

[12] Muhao Chen, Yingtao Tian, Mohan Yang, and Carlo Zaniolo. 2016. Multilingual knowledge graph embeddings for cross-lingual knowledge alignment. *arXiv preprint arXiv:1611.03954* (2016).

[13] Xuan Chen, Tong Lu, and Zhichun Wang. 2024. LLM-Align: Utilizing Large Language Models for Entity Alignment in Knowledge Graphs. doi:10.48550/arXiv.2412.04690

[14] Antonia Creswell, Kyriacos Nikiforou, Oriol Vinyals, Andre Saraiva, Rishabh Kabra, Loic Matthey, Chris Burgess, Malcolm Reynolds, Richard Tanburn, Marta Garnelo, et al. 2020. Alignnet: Unsupervised entity alignment. *arXiv preprint arXiv:2007.08973* (2020).

[15] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[16] Sarah G Elnaggar, Ibrahim E Elsemman, and Taysir Hassan A Soliman. 2023. Embedding-Based Deep Neural Network and Convolutional Neural Network Graph Classifiers. *Electronics* 12, 12 (2023), 2715.

[17] Nikolaos Fanourakis, Vasilis Efthymiou, Dimitris Kotzinos, and Vassilis Christophides. 2022. Knowledge Graph Embedding Methods for Entity Alignment: An Experimental Review. *arXiv preprint arXiv:2203.09280* (2022).

[18] Nicholas Halliwell, Fabien Gandon, and Freddy Lecue. 2021. User scored evaluation of non-unique explanations for relational graph convolutional network link prediction on knowledge graphs. In *Proceedings of the 11th Knowledge Capture Conference*. 57–64.

[19] Xiaochuang Han, Byron C Wallace, and Yulia Tsvetkov. 2020. Explaining black box predictions and unveiling data artifacts through influence functions. *arXiv preprint arXiv:2005.06676* (2020).

[20] Vikas Hassija, Vinay Chamola, Atmesh Mahapatra, Abhinandan Singal, Divyansh Goel, Kaizhu Huang, Simone Scardapane, Indro Spinelli, Mufti Mahmud, and Amir Hussain. 2024. Interpreting black-box models: a review on explainable artificial intelligence. *Cognitive Computation* 16, 1 (2024), 45–74.

[21] Meng He, Lijuan Duan, Baochang Zhang, and Shengwen Han. 2023. Knowledge Graph Embedding Method Based on Entity Metric Learning. In *Proceedings of the 2023 9th International Conference on Computing and Artificial Intelligence*. 381–387.

[22] Youmna Ismaeil, Daria Stepanova, Trung-Kien Tran, and Hendrik Blockeel. 2023. FeaBI: A Feature Selection-Based Framework for Interpreting KG Embeddings. In *International Semantic Web Conference*. Springer, 599–617.

[23] Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and S Yu Philip. 2021. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE transactions on neural networks and learning systems* 33, 2 (2021), 494–514.

[24] Chuanyu Jiang, Yiming Qian, Lijun Chen, Yang Gu, and Xia Xie. 2023. Unsupervised Deep Cross-Language Entity Alignment. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 3–19.

[25] Xuhui Jiang, Yinghan Shen, Zhichao Shi, Chengjin Xu, Wei Li, Zixuan Li, Jian Guo, Huawei Shen, and Yuanzhuo Wang. 2024. Unlocking the Power of Large Language Models for Entity Alignment. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Lun-Wei Ku, Andre Martins, and Vivek Srikumar (Eds.). Association for Computational Linguistics, Bangkok, Thailand, 7566–7583. doi:10.18653/v1/2024.acl-long.408

[26] Xuhui Jiang, Chengjin Xu, Yinghan Shen, Fenglong Su, Yuanzhuo Wang, Fei Sun, Zixuan Li, and Huawei Shen. 2023. Rethinking GNN-based Entity Alignment on Heterogeneous Knowledge Graphs: New Datasets and A New Method. *arXiv preprint arXiv:2304.03468* (2023).

[27] Insa Lawler and Emily Sullivan. 2021. Model explanation versus model-induced explanation. *Foundations of Science* 26 (2021), 1049–1074.

[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 (2015), 167–195.

[29] Shengxuan Luo and Sheng Yu. 2022. An Accurate Unsupervised Method for Joint Entity Alignment and Dangling Entity Detection. In *Findings of the Association for Computational Linguistics: ACL 2022*, Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (Eds.). Association for Computational Linguistics, Dublin, Ireland, 2330–2339. doi:10.18653/v1/2022.findings-acl.183

[30] Xin Mao, Wenting Wang, Yuanbin Wu, and Man Lan. 2021. From Alignment to Assignment: Frustratingly Simple Unsupervised Entity Alignment. In *Proceedings*

of the 2021 Conference on Empirical Methods in Natural Language Processing, Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (Eds.). Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 2843–2853. doi:10.18653/v1/2021.emnlp-main.226

[31] Xin Mao, Wenting Wang, Yuanbin Wu, and Man Lan. 2022. LightEA: A Scalable, Robust, and Interpretable Entity Alignment Framework via Three-view Label Propagation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (Eds.). Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 825–838. doi:10.18653/v1/2022.emnlp-main.52

[32] Daniel Obraczka, Jonathan Schuchart, and Erhard Rahm. 2021. Embedding-Assisted Entity Resolution for Knowledge Graphs. In *KGCW@ESWC*. https://api.semanticscholar.org/CorpusID:235357613

[33] Matteo Paganelli, Francesco Del Buono, Andrea Baraldi, Francesco Guerra, et al. 2022. Analyzing how BERT performs entity matching. *Proceedings of the VLDB Endowment* 15, 8 (2022), 1726–1738.

[34] George Papadakis, Marco Fisichella, Franziska Schoger, George Mandilaras, Nikolaus Augsten, and Wolfgang Nejdl. 2023. Benchmarking Filtering Techniques for Entity Resolution. In *39th IEEE International Conference on Data Engineering, ICDE 2023, Anaheim, CA, USA, April 3-7, 2023*. IEEE, 653–666. doi:10.1109/ICDE55515.2023.00389

[35] Nathalie Pernelle, Fatiha Saïs, and Danai Symeonidou. 2013. An automatic key discovery approach for data linking. *Journal of Web Semantics* 23 (2013), 16–30. doi:10.1016/j.websem.2013.07.001

[36] Frank Rosenblatt. 1958. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review* 65, 6 (1958), 386.

[37] Andrea Rossi, Denilson Barbosa, Donatella Firmani, Antonio Matinata, and Paolo Merialdo. 2021. Knowledge graph embedding for link prediction: A comparative analysis. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 15, 2 (2021), 1–49.

[38] Andrea Rossi, Donatella Firmani, Paolo Merialdo, and Tommaso Teofili. 2022. Explaining link prediction systems based on knowledge graph embeddings. In *Proceedings of the 2022 international conference on management of data*. 2062–2075.

[39] Tathagata Sengupta, Cibi Pragadeesh, Partha Pratim Talukdar, et al. 2017. Inducing interpretability in knowledge graph embeddings. *arXiv preprint arXiv:1712.03547* (2017).

[40] Zequn Sun, Wei Hu, and Chengkai Li. 2017. Cross-lingual entity alignment via joint attribute-preserving embedding. In *The Semantic Web–ISWC 2017: 16th International Semantic Web Conference, Vienna, Austria, October 21–25, 2017, Proceedings, Part I 16*. Springer, 628–644.

[41] Zequn Sun, Wei Hu, Qingheng Zhang, and Yuzhong Qu. 2018. Bootstrapping entity alignment with knowledge graph embedding.. In *IJCAI*, Vol. 18.

[42] Zequn Sun, Jiacheng Huang, Wei Hu, Muhao Chen, Lingbing Guo, and Yuzhong Qu. 2019. Transedge: Translating relation-contextualized embeddings for knowledge graphs. In *The Semantic Web–ISWC 2019: 18th International Semantic Web Conference, Auckland, New Zealand, October 26–30, 2019, Proceedings, Part I 18*. Springer, 612–629.

[43] Zequn Sun, Qingheng Zhang, Wei Hu, Chengming Wang, Muhao Chen, Farahnaz Akrami, and Chengkai Li. 2020. A benchmarking study of embedding-based entity alignment for knowledge graphs. *arXiv preprint arXiv:2003.07743* (2020).

[44] Danai Symeonidou, Vincent Armant, Nathalie Pernelle, and Fatiha Saïs. 2014. SAKey: Scalable Almost Key Discovery in RDF Data. In *The Semantic Web – ISWC 2014: 13th International Semantic Web Conference, Riva Del Garda, Italy, October 19-23, 2014. Proceedings, Part I* (Riva del Garda, Italy). Springer-Verlag, Berlin, Heidelberg, 33–49. doi:10.1007/978-3-319-11964-9_3

[45] Xiaobin Tang, Jing Zhang, Bo Chen, Yang Yang, Hong Chen, and Cuiping Li. 2020. BERT-INT: a BERT-based interaction model for knowledge graph alignment. *interactions* 100 (2020), e1.

[46] Bayu Distiawan Trisedya, Flora D Salim, Jeffrey Chan, Damiano Spina, Falk Scholer, and Mark Sanderson. 2023. i-Align: an interpretable knowledge graph alignment model. *Data Mining and Knowledge Discovery* 37, 6 (2023), 2494–2516.

[47] Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. 2017. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering* 29, 12 (2017), 2724–2743.

[48] Kexuan Xin, Zequn Sun, Wen Hua, Wei Hu, and Xiaofang Zhou. 2022. Informed multi-context entity alignment. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*. 1197–1205.

[49] Kaisheng Zeng, Chengjiang Li, Lei Hou, Juanzi Li, and Ling Feng. 2021. A comprehensive survey of entity alignment for knowledge graphs. *AI Open* 2 (2021), 1–13.

[50] Rui Zhang, Bayu Distiawan Trisedya, Miao Li, Yong Jiang, and Jianzhong Qi. 2022. A benchmark and comprehensive survey on knowledge graph entity alignment via representation learning. *The VLDB Journal* 31, 5 (2022), 1143–1168.

[51] Hao Zhu, Ruobing Xie, Zhiyuan Liu, and Maosong Sun. 2017. Iterative Entity Alignment via Joint Knowledge Embeddings.. In *IJCAI*, Vol. 17. 4258–4264.