Spatial Link Prediction with Spatial and Semantic Embeddings

Genivika Mann[1], Alishiba Dsouza[1(✉)], Ran Yu[1,2], and Elena Demidova[1,2]

[1] Data Science & Intelligent Systems (DSIS), University of Bonn, Bonn, Germany
{genivika.mann,dsouza,elena.demidova}@cs.uni-bonn.de, ran.yu@uni-bonn.de
[2] Lamarr Institute for Machine Learning and Artificial Intelligence, Bonn, Germany
https://lamarr-institute.org/

Abstract. Semantic geospatial applications, such as geographic question answering, have benefited from knowledge graphs incorporating information regarding geographic entities and their relations. However, one of the most critical limitations of geographic knowledge graphs is the lack of semantic relations between geographic entities. The most extensive knowledge graphs specifically tailored to geographic entities are extracted from unstructured sources, with these graphs often relying on datatype properties to describe the entities, resulting in a flat representation that lacks entity relationships. Therefore, predicting links between geographic entities is essential for advancing semantic geospatial applications. Existing neural link prediction methods for knowledge graphs typically rely on pre-existing entity relations, making them unsuitable for scenarios where such information is absent. In this paper, we tackle the challenge of predicting spatial links in sparsely interlinked knowledge graphs by introducing two novel approaches: supervised spatial link prediction (SSLP) and unsupervised inductive spatial link prediction (USLP). These approaches leverage the wealth of literal values in geographic knowledge graphs through spatial and semantic embeddings. To assess the effectiveness of our proposed methods, we conduct evaluations on the WorldKG geographic knowledge graph, which incorporates geospatial data extracted from OpenStreetMap. Our results demonstrate that the SSLP and USLP approaches substantially outperform state-of-the-art link prediction methods.

Keywords: Knowledge Graph Completion · Spatial Link Prediction · Literals

1 Introduction

Knowledge graphs (KGs) serve as standardized semantic knowledge representations that facilitate the integration, inference, and relationship establishment among heterogeneous data sources. Domain-specific geographic knowledge graphs are specialized knowledge graphs that focus on representing locations on Earth. Although knowledge graphs are widely adopted in various semantic applications, their incompleteness remains a challenging problem. Link prediction in

T. R. Payne et al. (Eds.): ISWC 2023, LNCS 14265, pp. 179–196, 2023.
https://doi.org/10.1007/978-3-031-47240-4_10

knowledge graphs has attracted a lot of research attention recently [7,21,28,30]. However, existing link prediction methods primarily focus on well-connected graphs with well-defined and structured object properties. Such methods often neglect rich semantic information in datatype properties that capture essential attributes of geographic entities such as names, descriptions, and spatial coordinates.

An example of a recently proposed geographic knowledge graph is WorldKG [8], which includes various entities representing geographic locations extracted from OpenStreetMap (OSM)[1]. As illustrated in Fig. 1, WorldKG contains the entity *wkg:10021976* representing the city of Leicester, located in the county of Leicestershire in the United Kingdom. Although WorldKG also contains the entities *wkg:838090640* representing the United Kingdom and *wkg:302324104* representing the county of Leicestershire, there are no links between these entities in the knowledge graph. The entity representing Leicester includes the spatial relations *wkgs:isInCountry* and *wkgs:isInCounty* associating this entity with the literals *"United Kingdom"* and *"Leicestershire"*, while lacking the links to the corresponding entities. Spatial link prediction can help interlink these geographic entities to further exploit the information in the knowledge graph.

In the context of link prediction to identify spatial relations between geographic entities, for instance, *wkgs:capitalCity*, *wkgs:isInCountry*, *wkgs:isInCounty*, spatial and literal values are critical. These values play a vital role in indicating entity proximity and the types of spatial relationships. Spatial link prediction can enhance the expressiveness of knowledge graphs, making it possible to solve complex spatial queries, reveal transitive relations, and eliminate geographic disambiguation issues. Downstream tasks in geospatial question answering, data retrieval, and cross-domain semantic data-driven applications in mobility, tourism, logistics, and city planning can also benefit significantly from accurate representations of spatial semantics in knowledge graphs.

Only few link prediction methods utilize textual entity descriptions [4,12] or numeric literals [23,27,28] to supplement the information provided through the graph structure. However, these approaches do not perform well without structural information from entity relations or in the presence of heterogeneous textual and numerical datatype properties of varied lengths [10]. Name disambiguation is another challenge when linking geographic entities, primarily due to the presence of homonymous names, synonyms, and variations. For instance, *Toronto* is a city's name in Canada and the USA. Similarly, *Germany*, *DE*, and *Deutschland* refer to the same country. Explicit spatial and contextual information is crucial for accurate spatial linking. Furthermore, existing approaches typically operate in transductive settings, aiming to predict the links between entities known at training time, whereas predicting links in the inductive settings, where the entities unseen during training appear, is a more difficult task [2].

[1] OpenStreetMap, OSM and the OpenStreetMap magnifying glass logo are trademarks of the OpenStreetMap Foundation, and are used with their permission. We are not endorsed by or affiliated with the OpenStreetMap Foundation.

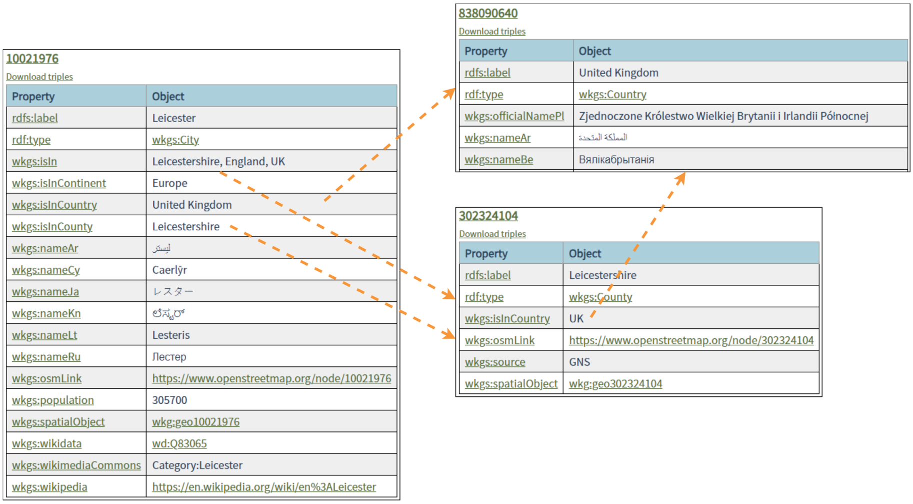

Fig. 1. An excerpt from the WorldKG knowledge graph [8], illustrating three entities *wkg:10021976, wkg:838090640, wkg:302324104* representing Leicester, United Kingdom and Leicestershire, respectively. Arrows in orange indicate potential spatial links currently missing in the WorldKG knowledge graph.

In this paper, we tackle the problem of spatial link prediction and introduce two novel approaches: Supervised Spatial Link Prediction (SSLP) and Unsupervised Spatial Link Prediction (USLP). These approaches are designed to operate in different modes, namely transductive and inductive link prediction. The SSLP architecture leverages location embedding and word embeddings to capture literal and spatial semantics, followed by enhancement of the tail embeddings using multi-head attention and a hierarchy-based scoring function to learn the containment hierarchy of geographic entities. In USLP, we score the tail entities for a given triple by computing the similarity between head, relation, and tail in different latent spaces based on geographic proximity and literal properties.

In summary, the main contributions of this paper are as follows:

- We propose two novel approaches, SSLP and USLP, for supervised and unsupervised spatial link prediction. These approaches leverage literal and geospatial semantics by incorporating spatial and semantic embeddings.
- We assess the performance of existing knowledge graph completion methods on the task of spatial link prediction in real-world scenarios where the knowledge graph lacks entity relationships.
- Through extensive experiments, we demonstrate that our proposed approaches outperform the baseline methods by a significant margin in terms of the Hits@k metric. These results highlight the effectiveness of our proposed approaches in spatial link prediction tasks.

2 Problem Statement

In this section, we formally define *RDF Knowledge Graph*, *Geographic Entity* and *Spatial Relation*, following which we state the problem of *Spatial Link Prediction* addressed by our work.

Definition 1 (RDF Knowledge Graph). *An RDF Knowledge Graph is a directed edge labeled multigraph represented by a set of triples* $\mathcal{G} = \{(h, r, t) \in \mathcal{E} \times \mathcal{R} \times (\mathcal{E} \cup \mathcal{L})\}$*, where* $\mathcal{E}$ = *a set of entities,* $\mathcal{R}$ = *a set of relations,* $\mathcal{L} = \mathcal{T} \cup \mathcal{N}$ *is the union of the set of textual literal values* $\mathcal{T}$ *and numeric literal values* $\mathcal{N}$*.*

For any triple $(h, r, t) \in \mathcal{G}$, we refer to the entity h as the *head* or *subject* entity, the entity t as the *tail* or *object* entity and the edge label r as *relation* or *predicate* of the triple. $\mathcal{G}_E$ is a set of relational triples representing object links, and $\mathcal{G}_L$ is a set of triples representing datatype properties linking entities to literal values. Hence, $\mathcal{G}_E = \{(h, r, t) \in \mathcal{G} \mid t \in \mathcal{E}\}$ and $\mathcal{G}_L = \{(h, r, t) \in \mathcal{G} \mid t \in \mathcal{L}\}$. The triples in $\mathcal{G}$ are the union of the two disjoint sets $\mathcal{G}_E$ and $\mathcal{G}_L$, therefore $\mathcal{G} = \mathcal{G}_E \cup \mathcal{G}_L$ and $\mathcal{G}_E \cap \mathcal{G}_L = \emptyset$.

Let $\mathcal{E}_{geo} \subseteq \mathcal{E}$ be the set of all geographic entities in the knowledge graph.

Definition 2 (Geographic Entity). *An entity* $e \in \mathcal{E}_{geo}$ *is a geographic entity* $\Leftrightarrow \exists r \in \mathcal{R}$ *such that* r *associates* e *to geographic coordinates (latitude and longitude).*

Spatial relations are connections between two geographic entities. These connections can imply physical entity relations such as containment, intersection, and adjacency, or conceptual relations such as capital, country, and suburb.

Definition 3 (Spatial Relation). *Let* $\mathcal{R}_{spatial} \subset \mathcal{R}$ *be a set of all spatial relations. A relation* $r_{sp} \in \mathcal{R}_{spatial}$ *is a spatial relation if* $\forall (h, r_{sp}, t) \in \mathcal{G}$ *it holds* $h, t \in \mathcal{E}_{geo}$*, i.e.,* h *and* t *are geographic entities.*

Spatial link prediction is the task of predicting spatial relations between geographic entities in a knowledge graph.

Definition 4 (Spatial Link Prediction). *Given a knowledge graph* $\mathcal{G}$*, a geographic entity* $h \in \mathcal{E}_{geo}$ *and spatial relation* $r_{sp} \in \mathcal{R}_{spatial}$*, find the geographic entity* $t \in \mathcal{E}_{geo}$ *such that* $(h, r_{sp}, t) \in \mathcal{G}$ *holds.*

3 Approach

We tackle the spatial link prediction problem with spatial and semantic embeddings and propose novel supervised and unsupervised approaches. In this section, we describe the embedding generation process and the proposed approaches. The supervised SSLP approach operates in the *transductive* link prediction setup. In transductive link prediction, the model predicts links between entities known during training; hence, training and prediction are conducted on the same set of

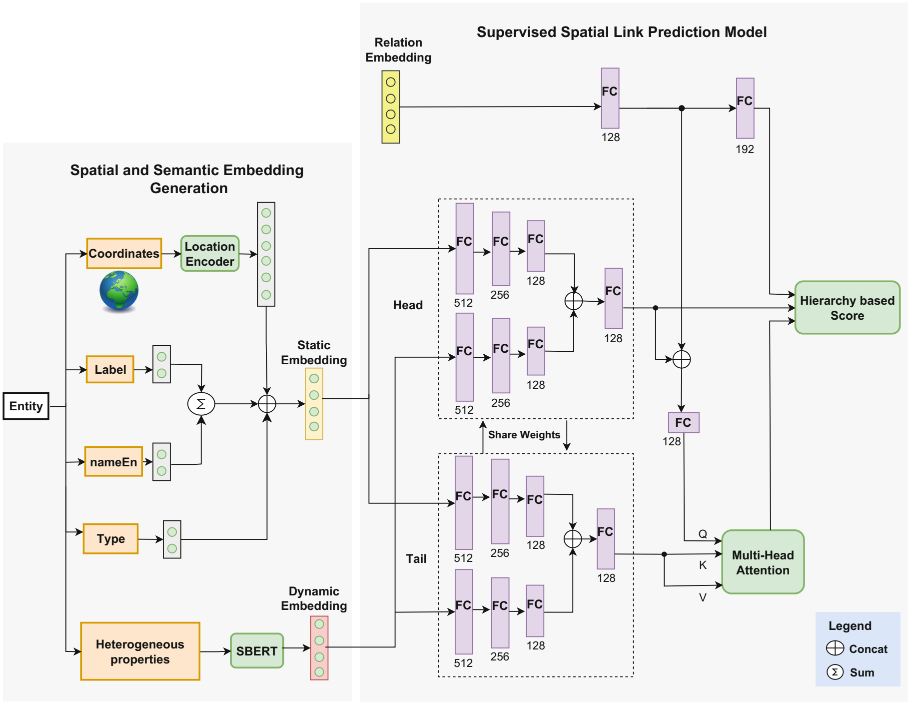

Fig. 2. Overview of spatial and semantic embedding generation of an entity and the architecture of SSLP model.

entities [2]. However, novel entities can emerge in knowledge graphs over time. The *inductive* link prediction setup reflects this scenario by facilitating spatial link prediction for unseen entities that do not appear during model training. Hence, in contrast to the transductive mode, links are predicted between seen and unseen entities of the knowledge graph. Our unsupervised USLP approach operates in this setting.

3.1 Spatial and Semantic Embedding of Entities

We utilize the datatype properties of a geographic entity to compute entity embeddings that capture its spatial and semantic information. The embedding generation process overview is illustrated in Fig. 2.

A geographic entity in a knowledge graph has geographic coordinates (latitude and longitude) associated with it. We embed these coordinates using the location encoding scheme proposed by Mai et al. [16] as a d_{loc} dimensional vector. Using sine and cosine functions of different frequencies, the location coordinates $x \in \mathbb{R}^2$ in 2D space are embedded as $\mathbf{X} \in \mathbb{R}^{d_{loc}}$-dimensional distributed representation. We consider the *rdfs:label* and *wkgs:nameEn* properties, which contain the entity label and name in the English language, and embed each of these values using pre-trained fastText word embedding [6]. We sum the word embeddings for *rdfs:label* and *wkgs:nameEn* to obtain a single embedding $\mathbf{W} \in \mathbb{R}^{d_{label}}$. The *rdf:type* property associates an entity to a class in an ontology. The entity type value is embedded using pre-trained fastText word embeddings to generate type embeddings $\mathbf{T} \in \mathbb{R}^{d_{type}}$ that capture word semantics.

The location, name-related embeddings, and type embeddings are concatenated to produce the static embedding $\mathbf{S} \in \mathbb{R}^{d_{loc}+d_{label}+d_{type}}$ of an entity. To create the dynamic embedding for each entity, we first concatenate the remaining heterogeneous predicates and their values to form a single sentence and then embed the sentence using the SBERT model [20]. The SBERT model employs Siamese and triplet network structures to produce $\mathbf{D} \in \mathbb{R}^{d_{dynamic}}$ semantically meaningful sentence embedding. Our spatial and semantic embeddings do not rely on links between entities; hence, such embeddings offer a robust alternative to predicting links in sparse knowledge graphs by exploiting datatype properties.

3.2 Supervised Spatial Link Prediction Approach

An overview of the SSLP architecture is presented in Fig. 2. We utilize the spatially and semantically rich static and dynamic entity embeddings to infer links for a given triple $(h, r, t) \in \mathcal{G}$. The relation r is embedded using fastText word embedding to reflect the semantics of relation names.

First, the architecture refines an entity's static and dynamic embedding, employing three fully connected layers with the ReLU activation function. After refinement, the static and dynamic embeddings are concatenated and passed through a fully connected layer to facilitate their fusion. This refinement and fusion operation is applied on both the head and tail entity embeddings, and weights are shared as done in a Siamese network to transform both head and tail embeddings to the same vector space. This ensures the feature learning from the head and tail entity occurs similarly, regardless of whether the entity appears in the head or tail position.

The head embedding obtained at this stage is used for scoring and does not pass through further layers, while multi-head attention is applied to the tail embedding to incorporate head and relation information in the tail embedding and to focus on relevant sections of the embedding. The head and relation embeddings are concatenated and passed through a linear layer to serve as the query, while the tail embedding serves as the key and value input for attention computation. The output of the attention block is treated as the final tail embedding. The head, relation, and tail embeddings are then projected into polar coordinates using a hierarchy-based score function.

The HAKE (Zhang et al. 2020 [30]) score function learns a hierarchy in an embedding space by only using triples to project entities to polar coordinates. We divide entities into hierarchical levels using *rdf:type* and modify the score function f_r to explicitly perform type-based hierarchical penalization using a hierarchy term f_{hterm}. This enables the modeling of semantic hierarchies of geographic entities in the embedding space. For instance, continents should be placed at the highest level, followed by countries, states, and districts, and finally, containing suburbs and burroughs at the lower hierarchy levels. The score function is formulated as follows:

$$f_r(h,t) = -\|\mathbf{h}_m \circ \mathbf{r}_m - \mathbf{t}_m\|_2 - \lambda \|\sin((\mathbf{h}_p + \mathbf{r}_p - \mathbf{t}_p)/2)\|_1 + \eta f_{hterm}. \quad (1)$$

Here, $\mathbf{h}_m, \mathbf{t}_m \in \mathbb{R}^k$, $\mathbf{r}_m \in \mathbb{R}^k_+$ are the radial coordinates, $\mathbf{h}_p, \mathbf{t}_p, \mathbf{r}_p \in [0, 2\pi)^k$ are the angular coordinates of head, tail, and relation embeddings respectively and $\lambda \in \mathbb{R}$ is a learnable weight of phase term. f_{hterm} is the hierarchical penalization term weighed by the learnable parameter $\eta \in \mathbb{R}$. The hierarchical penalization is computed as:

$$f_{hterm}(h,r,t) = \mathbf{r}_{dir}\left(\mathbf{h}_{level} - \mathbf{t}_{level}\right), \quad (2)$$

where $\mathbf{h}_{level}, \mathbf{t}_{level} \in \mathbf{Z}_+$ are hierarchy levels using *rdf:type* of head and tail entity respectively and $\mathbf{r}_{dir} \in [-1, +1]$ controls whether the head or tail entity should have a higher hierarchy level for the given relation.

Sampling negative triples is crucial for training models. In our approach, we sample our space to include a diversity of classes by sampling tails that have a different *rdf:type* than the true tail entity and also address hard cases by sampling tails of the same *rdf:type* as the true tail. Sampling tails from different classes allows us to incorporate class distribution during learning. The loss is computed using self-adversarial negative sampling loss (Sun et al. 2019 [22]):

$$L = -\log \sigma(\gamma - f_r(\mathbf{h}, \mathbf{t})) - \sum_{i=1}^{n} p(h'_i, r, t'_i) \log \sigma(f_r(\mathbf{h}'_i, \mathbf{t}'_i) - \gamma). \quad (3)$$

The parameter γ is a fixed margin, σ is the sigmoid function and (h'_i, r, t'_i) is the ith negative triple. The probability distribution of sampling negative triples with α temperature of sampling is given by (Sun et al. 2019 [22]):

$$p(h'_j, r, t'_j | \{(h_i, r, t_i)\}) = \frac{\exp \alpha f_r(\mathbf{h}'_j, \mathbf{t}'_j)}{\sum_i \exp \alpha f_r(\mathbf{h}'_i, \mathbf{t}'_i)}. \quad (4)$$

3.3 Unsupervised Spatial Link Prediction Approach

An overview of the USLP architecture is presented in Fig. 3.

In contrast to the supervised SSLP approach, the unsupervised USLP approach does not require training data for link prediction and operates in the inductive mode. The triple scores are computed by comparing the similarity of head entity, relation, and tail entity features in three spaces, namely *geographic space*, *name space*, and *class space*. In the *geographic space*, the coordinates of head and tail entities are represented using geohash. A geohash is an alphanumeric string that serves as a unique identifier and compact representation of regions using bounding boxes on the Earth surface [9]. This method splits the Earth surface into grid cells of various sizes depending on the length of the geohash.

The precision of the geohash is selected based on the relation in the triple, such that relations that may have a larger spatial distance between the entities are assigned a shorter geohash length to represent a larger area. Similarly, relations where the entities are assumed to be spatially closer, for example, *addrSuburb*, *addrHamlet*, are represented using longer geohash values. The centroid of the rectangular geohash grid is computed for both head and tail entities, and the Haversine distance between the geohash centroids serves as the similarity score in the *geographic space*.

Knowledge graphs often contain spatial relations where the object is a literal string representing the tail entity instead of the link to the entity. This literal value can match candidate tail entities based on their *rdfs:label* and *wkgs:nameEn* properties. In the *name space*, the pre-trained fastText word embeddings of the tail entity generated using *rdfs:label* and *wkgs:nameEn* property are compared with the embeddings of the literal string present in the object position of the spatial relation. The cosine similarity between the two embeddings is used to score the triples in this space. The *class space* also computes the cosine similarity between the embeddings of relation name and tail *rdf:type* class to assign a higher score to entities whose *rdf:type* is semantically similar to relation names in the shared semantic latent space. For example, the relation *wkgs:isInCountry* has a high similarity score with all entities of *rdf:type* country.

The final triple score is computed as the sum of the similarity scores of the embeddings in all three spaces. Tail entities closer to the triple head and relation in the spatial and semantic vector spaces have a higher likelihood of being linked and will therefore achieve a higher score using this scoring scheme.

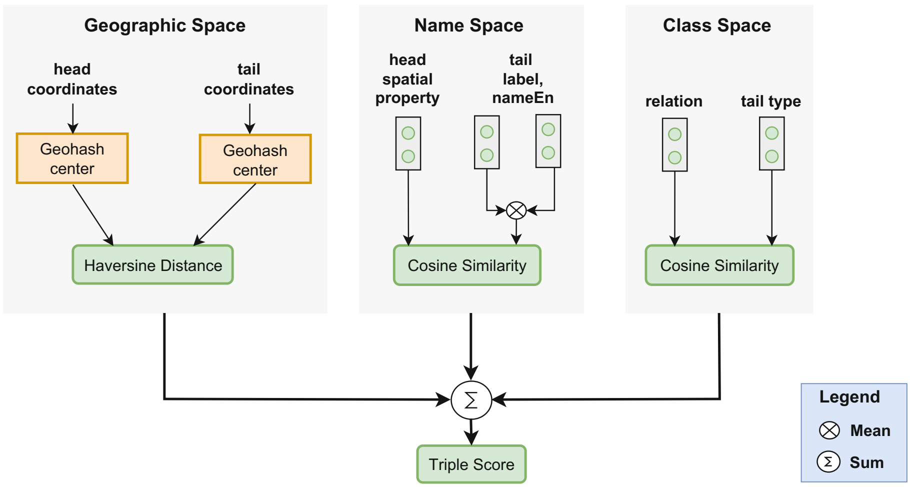

Fig. 3. Architecture of the USLP model.

4 Evaluation Setup

This section describes datasets, ground truth creation, baselines, and evaluation metrics. All experiments were conducted on AMD Ryzen 9 3900X 12-Core processor @ 2.2 GHz and 128 GB of memory using NVIDIA RTX A6000 GPU, CUDA 12.1, Python 3.10.9, and PyTorch 2.0.0. The SSLP model was trained for 3000 epochs with a batch size of 64, a learning rate of 0.001 optimized using Adam optimizer.

4.1 Datasets

For our spatial link prediction experiments, we use the WorldKG knowledge graph constructed by Dsouza et al. [8] using data from OpenStreetMap (OSM), one of the richest sources of openly available semantic volunteered geographic information. Each entity in WorldKG has geographic coordinates and heterogeneous spatial and non-spatial properties with textual and numeric literal values indicating entity names in different languages, areas, populations, geographic divisions such as a county, state, district, hamlet, and links to Wikipedia, Wikidata, and OSM.

4.2 Ground Truth Creation

In the current dataset version, the WorldKG knowledge graph (version 1.0) contains information extracted from OSM tags and does not contain any object property triples, lacking links between entities. All spatial relations are represented as datatype properties, with a string as the object representing a geographic entity. We extracted all triples with predicates that indicate spatial relations in the knowledge graph and prepared a set of candidate entities that were subclasses of *wkgs:Place*. Rule-based matching, using Haversine distance from subject entities, string matching of literal strings in the object position with entity names, and the type and relation-based filtering of subject/object entities, was performed on a subset of triples containing instances covering all spatial relations. We also used the identity link *wkgs:wikidata*, which links entities present in Wikidata and WorldKG, for matching a subset of spatial relations using equivalent relations present in the former knowledge graph.

The ground truth was used to create two transductive datasets, namely **TD1** and **TD2**, for evaluating the supervised approaches. The dataset **TD1** contains ground truth triples generated using WorldKG only, while **TD2** additionally contains triples created using WorldKG and Wikidata identity links. Furthermore, we created the inductive dataset **ID1** containing novel entities in the test triples for evaluating the unsupervised approaches. The test triples of **ID1** are generated by a stratified sampling of 133 triples from the ground truth. Table 1 contains statistics regarding the number of entities, relations, and triples in the three datasets.

Table 1. Dataset statistics. Rows indicate the count of each statistic.

Statistic	Dataset		
	TD1	TD2	ID1
#Entities	223297	284305	313385
#Spatial relations	14	14	14
#Literal relations	354	431	128
#Literal triples	1571175	1077982	1853
#Train triples	247419	323066	323066
#Validation triples	6145	10493	39943
#Test triples	26087	15931	133

4.3 Baselines

We consider state-of-the-art and well-established link prediction methods that can be classified into the following categories as our baselines:

- Translation distance model **TransE** [7] treats relations as a translation from head entity to tail entity. Roto-translational models **RotatE** [22] and **HAKE** [30] consider relations as rotations in a complex vector space.
- Tensor Decomposition models **RESCAL** [18], **DistMult** [29] and **ComplEx** [24].
- Deep learning models such as **ConvKB** [17] which uses convolutional neural networks and **CompGCN** [25] employing graph neural networks.
- **LiteralE** [12] and **Literal2Entity-DistMult** variant proposed by [5] where the former uses literal values to enrich KG embeddings and the latter performs graph transformations using literals.
- For the unsupervised approach, a naive baseline **Levenshtein similarity** (LS) computes string similarity between *rdfs:label* and *wkgs:nameEn* properties of the entities to score tail triples.

All baselines were trained using the default hyperparameters and the evaluation settings reported in the respective publications.

4.4 Evaluation Metrics

We evaluate our models on the spatial link prediction task by following the standard link prediction setup. For each triple $(h, r_{sp}, t) \in \mathcal{G}_{test}$, the set of corrupted triples $\mathcal{T}^-$ is generated by replacing the true tail entity t with all other geographic entities in the knowledge graph, hence $\mathcal{T}^- = \{(h, r_{sp}, t') \mid t' \in (\mathcal{E}_{geo} - \{t\})\}$. The model scores the true triple (h, r_{sp}, t) and corrupted triples in $\mathcal{T}^-$. The triples' scores are then sorted to obtain the rank of the true triple. We use the *filtered* evaluation setting [7] and filter the corrupted triples $\mathcal{T}^-$ to exclude the triples present in the training and validation set. We summarize the overall performance of the models using *Mean rank* (MR), *Mean reciprocal rank* (MRR), and *Hits@k* for $k \in \{1, 3, 5, 10\}$. *Mean rank* is the average rank of the test triples. *Mean reciprocal rank* is the mean over the reciprocal of individual ranks of test triples. *Hits@k* is the ratio of test triples present among the top k ranked triples.

5 Evaluation

This section aims to assess the performance of our proposed SSLP and USLP approaches against prominent baselines.

5.1 Performance in Transductive Setting

The link prediction results in the transductive setting for datasets TD1 and TD2 are summarized in Table 2. Our proposed USLP approach outperforms all baselines on both datasets regarding MRR and Hits@k and achieves the second-best mean rank on TD1. Its performance is closely followed by the proposed SSLP approach, with Hits@3,5,10 lying around three percentage points below the unsupervised model for TD1 and TD2. This result can be attributed to the additional spatial and literal information exploited by our approaches, along with triples for link prediction. The Literal2Entity variant with DistMult score function [5] also utilizes KG literals to transform the knowledge graph and performs well compared to other baselines, which only consider relational triples. It achieves a competitive MRR and Hits@1 compared to USLP on TD1; however, all other metrics are relatively lower than SSLP and USLP on both datasets. A possible reason for its performance lag is the absence of spatial semantics and hierarchical information in the literal transformations and scoring function used in this model, which can be beneficial when predicting spatial links. The LiteralE baseline fails on all metrics, despite utilizing literal triples. A possible reason for its negligible performance can be the inability of the model to scale to a large number of literal relations present in our datasets.

The remaining baselines use knowledge graph triples for predicting links, with their performance lying below a margin of around 20% points in terms of Hits@k compared to USLP and SSLP on both datasets. The sparsity of graph neighborhood and lack of sufficient structural information affects the link prediction ability of the baselines CompGCN and ConvKB employing graph neural networks and convolutions. Simple models such as TransE, RotatE, and HAKE, which treat relations as geometric operations, perform poorly regarding all metrics on TD1 and TD2. These approaches randomly initialize entity and relation embeddings and transform them based on observed triples in the graph, causing performance drop when predicting links in sparse knowledge graphs. DistMult achieves better Hits@k and MRR than other baselines on TD1 and TD2; overall, its results are lower than our approaches by a considerable margin on the TD2 dataset, especially in the case of Hits@1 and Hits@3 metrics.

The ablation study results of SSLP on dynamic embedding, multi-head attention, scoring function, and hierarchy penalization are shown in Table 3. Removing

Table 2. Spatial link prediction performance in transductive setting. The best results are highlighted in bold, and runner-up results are underlined.

Model	TD1 Dataset						TD2 Dataset					
	MR	MRR	H@1	H@3	H@5	H@10	MR	MRR	H@1	H@3	H@5	H@10
TransE	1104	0.147	0.065	0.174	0.229	0.306	12594	0.199	0.114	0.237	0.286	0.357
DistMult	1328	0.861	0.780	0.934	0.968	0.977	22005	0.660	0.568	0.747	0.799	0.808
HAKE	1592	0.327	0.214	0.426	0.475	0.499	13194	0.305	0.235	0.366	0.402	0.409
RotatE	1730	0.791	0.743	0.831	0.849	0.865	31671	0.010	0.002	0.004	0.006	0.021
ComplEx	34410	0.005	0.002	0.005	0.007	0.012	41360	0.025	0.016	0.026	0.032	0.041
CompGCN	750	0.095	0.000	0.004	0.078	0.396	17508	0.229	0.020	0.396	0.485	0.580
ConvKB	2555	0.099	0.019	0.132	0.179	0.249	11348	0.247	0.022	0.405	0.531	0.655
RESCAL	446	0.006	0.000	0.001	0.001	0.000	12337	0.002	0.000	0.000	0.000	0.000
LiteralE	70129	0.000	0.000	0.000	0.000	0.000	43276	0.022	0.016	0.018	0.022	0.039
L2E-DistMult	986	0.936	0.919	0.951	0.958	0.963	7113	0.797	0.776	0.811	0.820	0.830
SSLP	**144**	0.894	0.823	0.962	0.972	0.978	6505	0.819	0.805	0.831	0.836	0.844
USLP	188	**0.964**	**0.941**	**0.990**	**0.995**	**0.996**	**68**	**0.856**	**0.845**	**0.862**	**0.866**	**0.872**

Table 3. Ablation study of SSLP. The best results are highlighted in bold, and runner-up results are underlined.

Component	TD1 Dataset						TD2 Dataset					
	MR	MRR	H@1	H@3	H@5	H@10	MR	MRR	H@1	H@3	H@5	H@10
w/o Dynamic Embedding	212	0.669	0.527	0.781	0.851	0.903	6304	0.508	0.383	0.599	0.658	0.684
w/o Attention	170	0.839	0.798	0.859	0.894	0.923	9787	0.682	0.639	0.702	0.725	0.758
w/ DistMult score	327	0.336	0.104	0.434	0.616	0.802	13969	0.226	0.034	0.333	0.604	0.633
w/o Hierarchy term ($\eta = 0$)	280	0.876	0.816	0.922	0.934	0.964	5444	0.582	0.423	0.713	0.743	0.786
w/ Hierarchy term (η = 0.25)	224	0.886	**0.831**	0.929	0.959	0.974	6732	0.649	0.458	**0.832**	**0.853**	**0.867**
w/ Hierarchy term (η = 0.5)	213	0.802	0.658	0.937	0.951	0.976	**4635**	0.754	0.688	0.813	0.825	0.836
SSLP	**144**	**0.894**	0.823	**0.962**	**0.972**	**0.978**	6505	**0.819**	**0.805**	0.831	0.836	0.844

dynamic embedding in SSLP results in a decline in performance across all metrics in both datasets. This result highlights the expressiveness of these embeddings in serving as a latent representation for the heterogeneous predicates present in entities and capturing valuable supplementary information such as provenance, description, population, currency, etc., which boosts link prediction performance. The removal of the Multi-head attention component resulted in a sharp decline in performance for TD1 and TD2, with Hits@k for all k falling by a margin of at least ten percentage points compared to SSLP on the latter dataset. Replacing our hierarchy-based scoring function with the DistMult score also causes a performance drop across all metrics, especially in the case of Hits@1, showcasing the benefit of our hierarchy-based scoring function. To examine the effect of incorporating hierarchy penalization in the scoring function, we experimented with different values for the weight parameter η. The hierarchy penalization in our scoring caused a significant improvement in the performance of SSLP, which uses $\eta = 0.8$ on both datasets, with a rise of Hits@1 by around 38% points for TD2 compared to scoring with $\eta = 0$. The weight initialization of $\eta = 0.25$ also produced the highest Hits@k for k = 3, 5, 10 in TD2.

Table 4. Spatial link prediction performance in the inductive setting. The best results are highlighted in bold, and runner-up results are underlined.

Model	`ID1 Dataset`					
	MR	MRR	H@1	H@3	H@5	H@10
LS w/ rdfs:label	<u>89</u>	<u>0.051</u>	0.000	<u>0.053</u>	<u>0.083</u>	<u>0.135</u>
LS w/ wkgs:nameEn	121	0.035	0.000	0.023	0.045	0.105
USLP	**3**	**0.832**	**0.774**	**0.857**	**0.902**	**0.939**

5.2 Performance in Inductive Setting

Table 4 shows the results of unsupervised approaches on the ID1 dataset. The naive baseline computes string similarity using Levenshtein distance between *rdfs:label* and *wkgs:nameEn*, showing poor results across all metrics. String matching using literal values for predicting entity links introduces ambiguity with no mechanism to disambiguate candidate entities. Our proposed approach, USLP, fuses spatial proximity with entity naming semantics and type information aiding in precise disambiguation and substantially improved results on the ID1 dataset.

Table 5 reports the results of feature analysis on the USLP model. We compute the evaluation metrics for different combinations of spaces and the inclusion of dynamic embeddings along with the spaces of USLP. We observe that for all spaces except *name space*, each space considered individually is unsuitable for the spatial link prediction task, with the Hits@1 metric lying close to zero. On the other hand, the *name space* attains Hits@k values for all k in the range of around 65 to 75% points. The *geographic space* captures vital information for predicting spatial links, with its Hits@k values significantly higher than *class space* and latent space of dynamic embeddings considered individually or together. Combining *geographic* and *name* spaces further improves the performance. The combination of *geographic*, *name*, and *class* space used in USLP is the most effective for spatial link prediction, as indicated by their consistently higher performance in terms of all metrics. Our approach aligns entities along multiple semantic spaces by considering their similarity and is robust against disambiguation challenges. By incorporating dynamic embeddings with USLP, Hits@k metrics for $k = 1, 3, 5$ are lower; however, Hits@10 increases. This result can be due to noise present in the heterogeneous properties of entities, which reduces the recall of the model but increases Hits@10 with more true entities scoring in the top 10 ranks.

6 Related Work

In this section, we discuss related work in knowledge graph link prediction.

Link Prediction. Traditional knowledge graph link prediction methods are based on rule mining or generating random walks in knowledge graphs. Rule mining approaches [1] mine generic and conditional declarative rules using KG triples, while random walk-based approaches [13] perform combinations of constrained, weighed, random walks and use path ranking algorithms to tune the weights of these walks. These methods directly exploit observable features of the graph but do not consider latent features of entities. Embedding-based approaches transform the high-dimensional knowledge graph to low-dimensional vector spaces, while preserving semantic information and considering latent features of entities. Rossi et al. [21] classify these approaches as Tensor Decomposition, Geometric, and Deep learning models. Tensor Decomposition models view the knowledge graph as a 3D adjacency matrix and decompose this tensor to low dimensional vectors to generate entity and relation representations, with the scoring function a formulation of a bilinear product [24] or non-bilinear [3]. Geometric models interpret relations as a geometric operation in a latent space, where the operation can be a form of translation in the case of Translational models [7] or rotation-like transformations either separately or along with translations for Roto-translational models [22,30]. Deep Learning models use convolutional neural networks, reinforcement learning, or graph neural networks to predict links [17,25]. However, all these approaches do not consider the literal values in a knowledge graph and rely on the connectivity between entities.

Link Prediction Using Literals. Few approaches use the literal values present in knowledge graphs for link prediction. Kristiadi et al. in [12] propose an extension module over existing link prediction methods, named LiteralE, to directly enrich entity embeddings with literal information using a learnable parameterized function. Li et al. [14] perform numeric link prediction by considering the attribute semantics of literals in KG embeddings with a more comprehensive representation of attribute semantics. These approaches face scalability issues due to constructing a literal vector or an attribute matrix. The papers [23,27,28] also predict numeric literals in knowledge graphs using ensemble learning, attribute value regression, or by simultaneously adding numerical attribute prediction loss to triple loss. Blum et al. [5] propose three knowledge graph transformations and add additional entities and relations in the knowledge graph to enable existing approaches to leverage literals. Biswas et al. [4] use attentive bidirectional Gated Recurrent Unit (GRU)-based encoder-decoder for link prediction and consider textual entity descriptions and graph walks of KG. Literal-based approaches consider either numeric literals or require long textual entity descriptions and only utilize this information to supplement the structural information provided by the existing links in the graph.

Table 5. Feature analysis of USLP. The best results are highlighted in bold, and runner-up results are underlined.

Feature	ID1 Dataset					
	MR	MRR	H@1	H@3	H@5	H@10
Geographic Space	14	0.340	0.150	0.414	0.541	0.692
Name Space	22	0.695	0.669	0.692	0.699	0.752
Class Space	48	0.043	0	0.015	0.030	0.045
Dynamic Embedding	69	0.091	0	0.090	0.150	0.316
Geographic & Name	<u>4</u>	<u>0.769</u>	<u>0.714</u>	0.767	0.842	0.887
Geographic & Class	9	0.355	0.143	0.489	0.654	0.842
Geographic & Dynamic Embedding	18	0.151	0	0.173	0.278	0.489
Name & Class	13	0.695	0.662	0.669	0.729	0.797
Name & Dynamic Embedding	10	0.627	0.511	0.707	0.774	0.835
Class & Dynamic Embedding	51	0.128	0.023	0.143	0.195	0.383
USLP w/ Dynamic Embedding	**3**	0.701	0.571	<u>0.797</u>	<u>0.887</u>	**0.962**
USLP	**3**	**0.832**	**0.774**	**0.857**	**0.902**	<u>0.939</u>

Link Prediction Using Spatial Information. Huang et al. [11] propose a method containing an enhancer, encoder, and decoder where the enhancer converts relations to relation expressions using lexical, spatial, structural, and attribute similarity networks, followed by the encoder to obtain vector representations and a decoder to perform relation prediction. SE-KGE by Mai et al. [15] is a location-aware embedding model designed for geographic question answering. The method encodes spatial footprints (coordinates and bounding boxes) using a location-encoder based on Space2Vec and positional encoding of the transformer model. Qiu et al. [19] perform geographic link prediction by encoding geospatial distance restriction as a weighing term based on Euclidean distance in the objective function of translational embedding models. In contrast, Wang et al. [26] add geographic constraints such as inclusion, adjacency, and intersection for optimizing existing models such as TransE and RESCAL. The link prediction approaches using spatial information involve the construction of multiple similarity networks [11] or geographic constraints such as intersection [26], making them infeasible on our datasets containing a large number of entities and datatype properties.

7 Conclusion

In this paper, we proposed two novel approaches, SSLP and USLP, for transductive and inductive spatial link prediction, respectively. Our approaches address a crucial gap in the state-of-the-art by considering literal values and spatial semantics of geographic entities. SSLP and USLP outperform all baselines in terms of Hits@k. Our results demonstrate that effective fusion of spatial and literal semantics in knowledge graphs can facilitate the completion of sparse KGs that lack connectivity, including knowledge graphs from the geographic domain.

In future research, we would like to explore the adoption of our proposed spatial and semantic embeddings for answering complex semantic spatial queries.

Supplemental Material Statement: Our source code, experimental data, and instructions for repeating all experiments are available at GitHub[2].

Acknowledgements. This work was partially funded by the DFG, German Research Foundation ("WorldKG", 424985896), the Federal Ministry for Economic Affairs and Climate Action (BMWK), Germany ("ATTENTION!", 01MJ22012C), and DAAD/BMBF, Germany ("KOALA", 57600865).

References

1. Ahmadi, N., Huynh, V., Meduri, V.V., Ortona, S., Papotti, P.: Mining expressive rules in knowledge graphs. ACM J. Data Inf. Qual. **12**(2), 8:1–8:27 (2020)
2. Ali, M., et al.: Improving inductive link prediction using hyper-relational facts. In: Hotho, A., et al. (eds.) ISWC 2021. LNCS, vol. 12922, pp. 74–92. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-88361-4_5
3. Balažević, I., Allen, C., Hospedales, T.: TuckER: tensor factorization for knowledge graph completion. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pp. 5185–5194 (2019)
4. Biswas, R., Sack, H., Alam, M.: MADLINK: attentive multihop and entity descriptions for link prediction in knowledge graphs. Semantic Web J. 2960–4174 (2022)
5. Blum, M., Ell, B., Cimiano, P.: Exploring the impact of literal transformations within knowledge graphs for link prediction. In: Proceedings of the 11th International Joint Conference on Knowledge Graphs, pp. 48–54. ACM (2022)
6. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. Trans. Assoc. Comput. Linguist. **5**, 135–146 (2017)
7. Bordes, A., Usunier, N., García-Durán, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: Proceedings of the 27th Annual Conference on Neural Information Processing Systems, pp. 2787–2795 (2013)
8. Dsouza, A., Tempelmeier, N., Yu, R., Gottschalk, S., Demidova, E.: WorldKG: a world-scale geographic knowledge graph. In: Proceeding of the 30th ACM International Conference on Information and Knowledge Management, pp. 4475–4484. ACM (2021)
9. Ganti, R.K., Srivatsa, M., Agrawal, D., Zerfos, P., Ortiz, J.: MP-trie: fast spatial queries on moving objects. In: Proceedings of the Industrial Track of the 17th International Middleware Conference, p. 1. ACM (2016)
10. Gesese, G.A., Biswas, R., Alam, M., Sack, H.: A survey on knowledge graph embeddings with literals: which model links better literal-ly? Semantic Web **12**(4), 617–647 (2021)
11. Huang, Z., Qiu, P., Yu, L., Lu, F.: MSEN-GRP: a geographic relations prediction model based on multi-layer similarity enhanced networks for geographic relations completion. ISPRS Int. J. Geo Inf. **11**(9), 493 (2022)

[2] https://github.com/gkmn21/SSLPandUSLP.

12. Kristiadi, A., Khan, M.A., Lukovnikov, D., Lehmann, J., Fischer, A.: Incorporating literals into knowledge graph embeddings. In: Ghidini, C., et al. (eds.) ISWC 2019. LNCS, vol. 11778, pp. 347–363. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-30793-6_20
13. Lao, N., Mitchell, T.M., Cohen, W.W.: Random walk inference and learning in A large scale knowledge base. In: Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, pp. 529–539. ACL (2011)
14. Li, M., Gao, N., Tu, C., Peng, J., Li, M.: Incorporating attributes semantics into knowledge graph embeddings. In: Proceedings of the 24th International Conference on Computer Supported Cooperative Work in Design, pp. 620–625. IEEE (2021)
15. Mai, G., et al.: SE-KGE: a location-aware knowledge graph embedding model for geographic question answering and spatial semantic lifting. Trans. GIS **24**(3), 623–655 (2020)
16. Mai, G., Janowicz, K., Yan, B., Zhu, R., Cai, L., Lao, N.: Multi-scale representation learning for spatial feature distributions using grid cells. In: Proceedings of the 8th International Conference on Learning Representations. OpenReview.net (2020)
17. Nguyen, D.Q., Nguyen, D.Q., Nguyen, T.D., Phung, D.: A convolutional neural network-based model for knowledge base completion and its application to search personalization. Semantic Web **10**(5), 947–960 (2019)
18. Nickel, M., Tresp, V., Kriegel, H.: A three-way model for collective learning on multi-relational data. In: Proceedings of the 28th International Conference on Machine Learning, pp. 809–816. Omnipress (2011)
19. Qiu, P., Gao, J., Yu, L., Lu, F.: Knowledge embedding with geospatial distance restriction for geographic knowledge graph completion. ISPRS Int. J. Geo Inf. **8**(6), 254 (2019)
20. Reimers, N., Gurevych, I.: Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pp. 3980–3990. ACM (2019)
21. Rossi, A., Barbosa, D., Firmani, D., Matinata, A., Merialdo, P.: Knowledge graph embedding for link prediction: a comparative analysis. ACM Trans. Knowl. Discov. Data **15**(2), 14:1–14:49 (2021)
22. Sun, Z., Deng, Z., Nie, J., Tang, J.: Rotate: knowledge graph embedding by relational rotation in complex space. In: Proceedings of the 7th International Conference on Learning Representations, OpenReview.net (2019)
23. Tay, Y., Tuan, L.A., Phan, M.C., Hui, S.C.: Multi-task neural network for non-discrete attribute prediction in knowledge graphs. In: Proceedings of the Conference on Information and Knowledge Management, pp. 1029–1038. ACM (2017)
24. Trouillon, T., Welbl, J., Riedel, S., Gaussier, É., Bouchard, G.: Complex embeddings for simple link prediction. In: Proceedings of the 33rd International Conference on Machine Learning, pp. 2071–2080. JMLR.org (2016)
25. Vashishth, S., Sanyal, S., Nitin, V., Talukdar, P.P.: Composition-based multi-relational graph convolutional networks. In: Proceedings of the 8th International Conference on Learning Representations, OpenReview.net (2020)
26. Wang, Y., Zhang, H., Xie, H.: Geography-enhanced link prediction framework for knowledge graph completion. In: Zhu, X., Qin, B., Zhu, X., Liu, M., Qian, L. (eds.) CCKS 2019. CCIS, vol. 1134, pp. 198–210. Springer, Singapore (2019). https://doi.org/10.1007/978-981-15-1956-7_18
27. Wu, Y., Wang, Z.: Knowledge graph embedding with numeric attributes of entities. In: Proceedings of The Third Workshop on Representation Learning for NLP, pp. 132–136. ACM (2018)

28. Xue, B., Li, Y., Zou, L.: Introducing semantic information for numerical attribute prediction over knowledge graphs. In: Proceedings of the 21st International Semantic Web Conference, pp. 3–21. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-19433-7_1
29. Yang, B., Yih, W., He, X., Gao, J., Deng, L.: Embedding entities and relations for learning and inference in knowledge bases. In: Proceedings of the 3rd International Conference on Learning Representations (2015)
30. Zhang, Z., Cai, J., Zhang, Y., Wang, J.: Learning hierarchy-aware knowledge graph embeddings for link prediction. In: Proceedings of the Thirty-Fourth Conference on Artificial Intelligence, pp. 3065–3072. AAAI Press (2020)