Type-Less yet Type-Aware Inductive Link Prediction with Pretrained Language Models

Anonymous ACL submission

Abstract

Inductive link prediction is emerging as a key paradigm for real-world knowledge graphs (KGs), where new entities frequently appear and models must generalize to them without retraining. Predicting links in a KG faces the challenge of guessing previously unseen entities by leveraging generalizable node features such as subgraph structure, type annotations, and ontological constraints. However, explicit type information is often lacking or incomplete. Even when available, type information in most KGs is often coarse-grained, sparse, and prone to errors due to human annotation. In this work, we explore the potential of pre-trained language models (PLMs) to enrich node representations with *implicit* type signals. We introduce *TyleR*, a Type-less yet type-awaRe approach for subgraph-based inductive link prediction that leverages PLMs for semantic enrichment. Experiments on standard benchmarks demonstrate that TyleR outperforms state-of-the-art baselines in scenarios with scarce type annotations and sparse graph connectivity. To ensure reproducibility, we share our code at `https://anonymous.4open.science/r/tyler-7C2C/`.

1 Introduction

Knowledge graphs (KGs) represent complex relationships between entities in a structured, graph-based format (Hogan et al., 2021). Their ability to encode semantic information and support reasoning makes them valuable in a variety of applications, such as natural language processing (Peters et al., 2019), recommendation systems (Wang et al., 2024), and biomedical research (Gema et al., 2023). However, KGs are notoriously incomplete: many valid relations are absent, reducing their effectiveness in downstream tasks (Rossi et al., 2021).

Link prediction aims to infer these missing relationships by analyzing the existing graph's structure and patterns. Traditional link prediction methods aim to predict links among entities observed during training. Although effective in static settings, they are limited in dynamic environments where new entities are incrementally introduced. Inductive link prediction (ILP) addresses this challenge by aiming to generalize to previously unseen entities, leveraging transferable features such as structural information and type information.

Prior work has demonstrated that incorporating entity type information can enhance the generalization capability of ILP models. For instance, Zhou et al. (2023) explicitly integrate type annotations and ontological constraints into the learning process. Yet, these methods face a critical bottleneck: the explicit type information available in real-world KGs is often coarse-grained, incomplete, or even erroneous. This limitation is particularly acute when facing structural sparsity. Consider, for instance, the triple ⟨`Lionel Messi`, `playedFor`, `Barcelona FC`⟩. A model might assign similar plausibility to ⟨`Cristiano Ronaldo`, `playedFor`, `Barcelona FC`⟩ if both subject entities (i.e., `Lionel Messi` and `Cristiano Ronaldo`) lack distinct neighborhood information and are categorized only under the broad type "`Footballer`". This highlights a fundamental inadequacy of type-informed ILP approaches when explicit type signals are weak and local graph structure is uninformative.

To address this gap, our idea is to leverage the rich semantic knowledge captured by pre-trained language models (PLMs). We hypothesize that the semantic understanding these models acquire during their extensive pre-training on vast textual corpora (Petroni et al., 2019; Hao et al., 2023) offers a pathway to a more fine-grained representation of entities. This "inner knowledge," encoded within the PLM's parameters, offers a dense representation of diverse semantic facets. For example, prompting a PLM like BERT (Devlin et al., 2019) with "Paris is located in __," generates a hidden representation for the missing token that (ideally) enables it to correctly predict "France," reflecting the model's "*understanding*" of the Paris's geographical location. We aim to utilize the implicit semantic insights of PLMs to derive fine-grained entity representations, overcoming limitations in explicit type information. We start from these two observations: (i) an entity can be described by a set of assertions defining its properties; (ii) the same assertions, when used as prompts for a PLM, can elicit dense, multifaceted representations that implicitly capture a "type-aware" understanding of the entity. This potential led us to ask: *Can PLM-derived entity representations compensate for structural and type sparsity in inductive knowledge graph completion?* To investigate this question, we introduce **TyleR**–***Ty**pe-**le**ss yet type-awa**R**e*–a novel inductive link prediction framework that leverages PLMs to embed implicit type-aware signals within node representations, thus eliminating reliance on explicit type annotations. Our contributions are:

1. We introduce a novel methodology for harnessing PLMs to derive and embed implicit type semantics within an ILP model, thereby enabling nuanced entity representations without relying on explicit type data.

2. We demonstrate TyleR's effectiveness on multiple benchmark datasets, showing its capability to perform competitively, especially in settings with limited or coarse-grained type information and sparse graph structures.

3. We conduct an empirical analysis investigating the interplay between PLM-derived semantic features and varying levels of type and structural sparsity, thereby characterizing the resilience of our approach.

The remainder of the paper is organized as follows: Section 2 introduces the idea behind subgraph-based relational inference; Section 3 details the methodology; Section 4 describes the experimental setup and evaluation; Section 5 presents the results; Section 6 reviews related work; and Section 7 concludes with future directions.

2 Background and Motivation

Inductive link prediction aims to predict the likelihood of triples (h, r, t), where h and t are unseen entities. In practice, this is done by means of a scoring function $f(h, r, t)$. At training time, f is optimized on the triples in a training graph $\mathcal{G}_{train}$. At test time, the same scoring function is used to predict the plausibility of triples (h', r, t') belonging to a test graph $\mathcal{G}_{test}$, based on the triples in an inference graph $\mathcal{G}_{inf}$. Unlike traditional embedding-based approaches, subgraph-based relation prediction methods such as GraIL (Teru et al., 2020) can be viewed as learning logical rules that capture entity-independent relational semantics. For example, one can derive the simple rule:

$$\textit{spouse\_of}(X,Y) \wedge \textit{lives\_in}(Y,Z) \rightarrow \textit{lives\_in}(X,Z). \quad (142)$$

As demonstrated by Zhou et al. (2023), the reasoning capabilities of GraIL can be enhanced by incorporating explicit *type information* about entities. This additional semantic context enables the model to induce more precise and type-aware rules:

$$\begin{aligned}&\textit{Employee}(X) \wedge \textit{Department}(Y) \wedge \textit{Office}(Z) \wedge \\ &\wedge \textit{part\_of}(X,Y) \wedge \textit{located\_in}(Y,Z) \rightarrow \textit{works\_in}(X,Z).\end{aligned} \quad (148)$$

Type-constrained rules enhance both accuracy and interpretability in relational inference by reducing spurious predictions and enforcing semantic validity. However, explicit type information is often incomplete or missing in real-world knowledge graphs. To address this, we propose learning a function τ_{PLM}, parameterized by a pre-trained language model, that maps entities to implicit type representations capturing their latent semantics. These PLM-derived embeddings enable type-aware reasoning without explicit type labels and can be integrated into the logical rule induction process. For example, a type-aware rule may take the form:

$$\begin{aligned}&\tau_{\text{PLM}}(X) \wedge \tau_{\text{PLM}}(Y) \wedge \tau_{\text{PLM}}(Z) \wedge \textit{part\_of}(X,Y) \wedge \\ &\qquad \wedge\ \textit{located\_in}(Y,Z) \rightarrow \textit{works\_in}(X,Z),\end{aligned} \quad (162)$$

with $\tau_{\text{PLM}}(X)$, $\tau_{\text{PLM}}(Z)$, and $\tau_{\text{PLM}}(Z)$ such that

$$\begin{aligned}\tau_{\text{PLM}}(X) &\approx \textit{Employee}(X), \\ \tau_{\text{PLM}}(Y) &\approx \textit{Department}(Y), \\ \tau_{\text{PLM}}(Z) &\approx \textit{Office}(Z),\end{aligned} \quad (164)$$

where $\tau_{\text{PLM}}(\cdot)$ for X is an approximation of the logical statement *Employee*$(\cdot)$ while, for Y and Z, $\tau_{\text{PLM}}(\cdot)$ is an approximation of their types, *Department* and *Office*, respectively (more details in Section 3). This guides the rule induction process towards more meaningful and generalizable patterns, allowing us to infuse latent type semantics into subgraph-based link prediction models, even when explicit type information is absent.

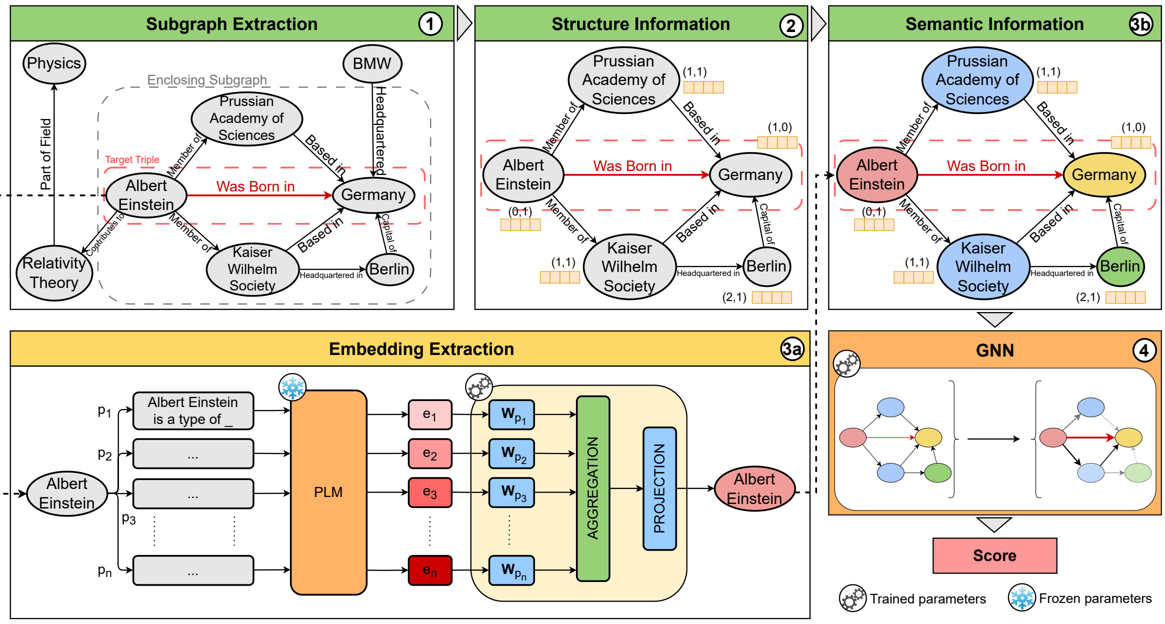

Figure 1: Overview of TyleR. The process begins with ① extracting the enclosing subgraph and ② applying a node labeling strategy. Multi-faceted, semantic representations are then derived using a pre-trained language model ③a, ③b. Finally, a graph neural network ④ integrates structural and semantic information to obtain the final prediction.

3 Methodology

In this section, we introduce **TyleR** (**Ty**pe-**le**ss yet type-awa**R**e inductive link prediction with pre-trained language models). Building on Graph Inductive Learning (Teru et al., 2020), which infers relations from local subgraph patterns, TyleR leverages PLM-derived semantics to enrich node representations. However, integrating PLMs into full-graph models is computationally expensive due to high-dimensional embeddings and large graph size. TyleR adopts a subgraph-reasoning approach, restricting triple scoring to compact and informative subgraphs, making PLM integration tractable.

As illustrated in Figure 1, TyleR 's pipeline consists of four stages: ① extracting an enclosing subgraph, ② structurally labeling nodes (following GraIL (Teru et al., 2020)), ③a, ③b enriching nodes with PLM-based semantic embeddings, and ④ feeding the enhanced subgraph into a GNN architecture from Zhou et al. (2023). The following sections provide further details on each step.

Subgraph Extraction ① Given a target triple (u, r_t, v), we define $\mathcal{N}_k(u)$ and $\mathcal{N}_k(v)$ as the sets of k-hop neighboring nodes of u and v, respectively. We also define a specific distance metric $d(i, u)$ as the shortest path from a node i to u that does not pass through v, and $d(i, v)$ is similarly the shortest path distance from i to v that does not pass through u. The *enclosing subgraph* of triple (u, r_t, v) is computed by (i) forming an initial set of candidate nodes by taking the intersection $\mathcal{N}_k(u) \cap \mathcal{N}_k(v)$ and (ii) pruning nodes that are either isolated (i.e., have no edges connecting it to other nodes within the subgraph after this pruning step) or for which $d(i, u) > k$ or $d(i, v) > k$. The remaining nodes and their edges form the enclosing subgraph.

Subgraph Labeling ② Each node i in the extracted subgraph is labeled with a pair of shortest path distances $(d(i, u), d(i, v))$ to the target nodes u and v, respectively, within the subgraph. This pair captures the relative *position* of node i with respect to the target nodes u and v. The final positional embedding $\mathbf{h}_i^{\text{pos}}$ is:

$$\mathbf{h}_i^{\text{pos}} = \text{one-hot}(d(i, u)) \oplus \text{one-hot}(d(i, v)), \quad (1)$$

where $\oplus$ denotes the concatenation operator and one-hot$(\cdot)$ is the one-hot encoding function. All nodes in the enclosing subgraph are within k hops of u or v, so $\mathbf{h}_i^{\text{pos}} \in \mathbb{R}^{2k+2}$.

Semantic Enrichment ③a③b Semantic enrichment leverages a pre-trained language model (PLM), supporting either masked token prediction or next-token generation. A straightforward approach for encoding entity type semantics involves prompting the PLM with an explicit query to elicit the most plausible type for a given entity. For masked language models (MLMs) (e.g., RoBERTa), this operation results in a prompt such as *"The type of Paris is* `[MASK]`*"*, with the type semantics encoded in the last hidden layer representation of the `[MASK]` token; for causal language models (CLMs) (e.g., Llama), this representation corresponds to the last hidden representation of the final sequence token. However, relying solely on representations derived from such direct type queries can be suboptimal. Prior research has shown that transformer-based representations tend to be highly anisotropic, often concentrated in narrow cones (Ethayarajh, 2019), which can limit their discriminative utility. To address this, we propose to refine type semantics through multiple prompts, designed to extract different semantic aspects. Given an entity i with textual label l_i, we define a set of **assertion prompts** $P = \{p_1, p_2, \dots, p_n\}$, where each p_k targets a semantic facet of the entity (e.g., type, location, membership). Each prompt $p_k(l_i)$ is processed by the PLM (3a) to yield a latent representation:

$$\mathbf{z}_{p_k,i} = \text{Extract}(\text{PLM}(p_k(l_i))). \tag{2}$$

Here, PLM$(\cdot)$ denotes the forward pass of the language model given an input prompt, and Extract$(\cdot)$ selects the relevant hidden state (i.e., the `[MASK]` token's final hidden layer for MLMs or the last token's representation for CLMs). These representations $\mathbf{z}_{p_k,i}$ are refined and projected into a unified space using an assertion-specific projection block:

$$\mathbf{z}^h_{p_k,i} = \mathbf{W}_{p_k}\text{LN}(\mathbf{z}_{p_k,i}) + \mathbf{b}_{p_k}, \tag{3}$$

where $\mathbf{W}_{p_k}$ and $\mathbf{b}_{p_k}$ are specific learnable parameters for each assertion prompt p_k, and LN denotes layer normalization (Ba et al., 2016). We aggregate (AGG$(\cdot)$) the prompt representations with different strategies such as *sum*, *mean* or *concatenation*:

$$\mathbf{z}^{\text{agg}}_i = \text{AGG}_p(\{\mathbf{z}^h_{p_k,i}\}^n_{k=1}). \tag{4}$$

The semantic embedding h^{sem}_i is obtained as:

$$\mathbf{h}^{sem}_i \equiv \tau_{\text{PLM}}(i) = \sigma(\mathbf{W}_o\text{ReLU}(\mathbf{z}^{\text{agg}}_i))), \tag{5}$$

where $\tau_{\text{PLM}}(i)$ is a function capturing the semantics of i by aggregating multiple prompt-based representations via a PLM (as introduced in Section 2), and $\sigma(\cdot)$ is the sigmoid function. Given a node i, we then construct the embedding h^0_i as (3b):

$$\mathbf{h}^0_i = [\mathbf{h}^{pos}_i \oplus \mathbf{h}^{sem}_i]. \tag{6}$$

GNN Scoring 4 As suggested by Zhou et al. (2023), our base GNN follows the R-GCN (Schlichtkrull et al., 2018) architecture. At layer l, the embedding for a node i is computed as:

$$\mathbf{h}^{(l)}_i = \text{ReLU}(\mathbf{W}^{(l)}_0\mathbf{h}^{(l-1)}_i + \mathbf{a}^{(l)}_i), \tag{7}$$

where $\mathbf{W}^{(l)}_0$ is a self-loop learnable matrix and $\mathbf{a}^{(l)}_i$ is the AGGREGATE function, based on edge attention (Teru et al., 2020) and entity-relation composition (Vashishth et al., 2020):

$$\mathbf{a}^{(l)}_i = \sum_{r\in R}\sum_{j\in\mathcal{N}^r(i)} \alpha^{(l)}_{rr_tji}\mathbf{W}^{(l)}_r(\mathbf{h}^{(l-1)}_j - \mathbf{e}^{(l-1)}_r), \tag{8}$$

where $\mathbf{W}^{(l)}_r$ is a relation-specific transformation matrix at layer l, $\mathcal{N}^r(i)$ is the set of outgoing neighboring nodes of node i under relation r. We adopt basis sharing (Schlichtkrull et al., 2018) as regularization for the $\mathbf{W}^{(l)}_r$ transformation matrices, whereas $\mathbf{e}^{(l)}_r$ is the relation embedding at layer l:

$$\mathbf{e}^{(l)}_r = \mathbf{W}^{(l)}_{rel}\mathbf{e}^{(l-1)}_r. \tag{9}$$

The edge attention weight $\alpha^{(l)}_{rr_tji}$ quantifies the importance of an edge (j, r, i) when inferring relation r_t at layer l.

$$\alpha^{(l)}_{rr_tji} = \sigma(\mathbf{W}^{(l)}_\alpha\mathbf{s}^{(l)}_{rr_tji} + b^{(l)}_\alpha), \tag{10}$$

$$\begin{aligned}\mathbf{s}^{(l)}_{rr_tji} = \text{ReLU}(\mathbf{W}^{(l)}_s[&\mathbf{h}^{(l-1)}_j \oplus \mathbf{h}^{(l-1)}_i \\ &\oplus \mathbf{e}^{(l-1)}_r \oplus \mathbf{e}^{(l-1)}_{r_t}] + \mathbf{b}^{(l)}_s).\end{aligned} \tag{11}$$

To obtain the final representation of a node, Teru et al. (2020) suggests adopting JK-Connections (Xu et al., 2018), i.e., by concatenating all the intermediate-layer representations. After the aggregation, the final score is computed as

$$f(u,r_t,v)=\mathbf{W}^T_f\bigoplus_{l=1}^{L}[\mathbf{h}^{(l)}_G(u,r_t,v)\oplus\mathbf{h}^{(l)}_u\oplus\mathbf{h}^{(l)}_v\oplus\mathbf{e}^L_{r_t}], \tag{12}$$

where $\mathbf{h}^{(l)}_G(u, r_t, v)$ is the subgraph representation, obtained via average pooling over all node representations at level l in the subgraph.

Loss Function We adopt a margin-based pairwise loss function, which aims at maximizing the score on positive triples and minimizing the score on randomly sampled negative triples:

$$\mathcal{L}=\sum_{(u,r_t,v)\in G} max(0,f_e(u',r_t,v')-f_e(u,r_t,v)+\gamma), \tag{13}$$

where γ is a margin hyperparameter, (u, r_t, v) is a positive triple and (u', r_t, v') is a negative triple.

4 Experimental Setup

In this section we detail our experimental setup, including datasets, baselines, training and evaluation details. Experiments were conducted with Python 3.8.19 and PyTorch 2.3.0, using an NVIDIA Ampere A100 GPU (64GB VRAM) and CUDA 12.1.

4.1 Datasets

We conduct experiments on YAGO21K-610 (Zhou et al., 2023) and three FB15K-237 (FB237 in short) variants (v1–v3) from Teru et al. (2020). Dataset statistics are in Appendix A, Table 4. For YAGO21K-610, we use the original splits with the provided ontology graph and type links; test entities are unseen during training, while relations are shared. Each FB237 variant contains disjoint train and inductive test graphs with distinct entities but shared relations. For each FB237 variant, we train on its designated training set and evaluate using its corresponding "ind" (inductive) set as the inference graph, with testing performed on its test set. For YAGO21K-610, when evaluating a specific target triple, the inference graph includes all other test triples (excluding the target itself), following Zhou et al. (2023). Since FB237 lacks concept annotations, we build ontology graphs and type links for all variants using Freebase-Wikidata mappings (see Appendix A). Dataset density, defined as $2|T|/|E|$ (Pujara et al., 2017), is the lowest for YAGO21K-610 train (3.67) and increases across FB237 variants (i.e., from 5.33 to 9.80). This pattern also holds for the inference graphs (density ranging from 3.54 for YAGO21K-610 to 5.92 for FB237-V3), allowing us to analyze the impact of type information under varying graph sparsity.

4.2 Metrics

We evaluate models using Mean Reciprocal Rank (MRR) and Hits@K for $K \in \{1, 10\}$, averaging over 5 evaluation runs. Following standard protocol (Teru et al., 2020), each positive test triple is ranked against 50 negative triples generated by randomly corrupting either its head or tail entity.

Tie resolution markedly affects these metrics. While methods like random tie-breaking (Rossi et al., 2021)—which randomly assign ranks among tied entities—are prevalent, they can lead to an overestimation of true model performance. This issue is particularly evident in sparse settings where limited structural or type information leads to frequent ties, an issue amplified by the candidate pool of 50. To address these concerns and provide a more stringent and reliable evaluation, we adopt a **strict tie-breaking strategy**. This approach assigns the positive triple the highest (i.e., worst-case/pessimistic) rank when its score is identical to one or more negative triples.

ID	Aspect	Template
p_1	type	Paris is a type of __
p_2	geographic	Paris is located in __
p_3	membership	Paris is member of __
p_4	equivalence	Paris is equivalent to __
p_5	difference	Paris is different from __
p_6	similarity	Paris is similar to __

Table 1: Assertion prompts (p_1-p_6) used in the semantic enrichment step (Section 3). These templates, with a placeholder for the entity, are fed to the Pre-trained Language Model to elicit representations capturing different semantic aspects (type, geographic context, membership, equivalence, difference, similarity) of the entity.

4.3 Models

To isolate the contribution of our semantic-enrichment module, we focus the comparison on methods that share a similar subgraph-reasoning backbone as Tyler. We evaluate TyleR against GraIL (Teru et al., 2020), a type-agnostic baseline that relies solely on subgraph structure, and the ontology-enhanced method of Zhou et al. (2023), which explicitly incorporates type information via learnable embeddings and ontological constraints, even though its effectiveness is tied to the availability and quality of type annotations and ontology triples. In contrast, Tyler is designed for scenarios where explicit type information is scarce as it is able to infer implicit type semantics from PLMs. Our semantic enrichment strategy, detailed in Section 3, is model-agnostic and compatible with any PLM supporting masked or causal language modeling. We use RoBERTa-Large (Liu et al., 2019) and Llama3-8B (Dubey et al., 2024) without fine-tuning, aggregating representations from six manually crafted assertion prompts (Table 1).

5 Results

Experiments aim to answer three core questions:

RQ1. Does explicit type information improve subgraph-based inductive link prediction?

Inductive LP Model	FB237-V1			FB237-V2			FB237-V3			YAGO21K-610		
	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10
GraIL (Teru et al., 2020)	.456	34.97	64.44	**.618**	**50.46**	82.70	.609	49.94	82.26	.661	**62.76**	68.68
Zhou et al. (2023)	.398	27.85	64.55	.576	44.69	82.45	.554	41.85	81.42	**.673**	60.36	76.56
TyleR-RoBERTa-L (2025)	.470	35.66	69.95	.602	47.51	**83.28**	**.630**	**50.60**	**86.72**	.660	58.26	**79.68**
TyleR-Llama3-8B (2025)	**.481**	**36.88**	**70.63**	.610	49.37	82.01	.620	49.94	84.46	.651	59.70	69.30

Table 2: Link Prediction (LP) evaluation on multiple FB237 variants and YAGO21K-610. Best and second-best scores are in **bold** and underlined, respectively. Evaluation uses the strictest tie-breaking policy (Section 4.2), assigning the highest (worst) possible rank to the positive triple in case of ties.

PLM	Aggregation	MRR	Hits@1	Hits@10
TyleR-RoBERTa-L	TYPE-ONLY	.442	32.93	66.49
TyleR-RoBERTa-L	SUM	**.470**	**35.66**	**69.95**
TyleR-RoBERTa-L	MEAN	.468	35.22	68.05
TyleR-RoBERTa-L	CONCAT	.455	34.10	67.76
TyleR-Llama3-8B	TYPE-ONLY	.477	**37.12**	68.29
TyleR-Llama3-8B	SUM	**.481**	36.88	70.63
TyleR-Llama3-8B	MEAN	.465	35.51	68.73
TyleR-Llama3-8B	CONCAT	.474	35.95	**71.12**

Table 3: Ablation study on the FB15K-237-V1 dataset, evaluating the impact of different Pre-trained Language Models and aggregation functions (Equation 4) for semantic embeddings within TyleR. 'TYPE-ONLY' uses only the representation from the p_1 prompt (Table 1).

RQ2. Can PLMs enhance node representations for subgraph-based inductive link prediction?

RQ3. Can PLMs mitigate type and structural sparsity challenges in inductive link prediction?

5.1 Type Information in Subgraph-Based Inductive Link Prediction (RQ1)

Table 2 presents link prediction results across various models and datasets, emphasizing the role of type information in inductive link prediction. GraIL, which operates without type information, performs competitively overall. It achieves strong results in both MRR and Hits@10, and obtains the highest Hits@1 on the sparse YAGO21K-610 dataset. This suggests its ability to rank correct entities precisely in low-density settings without relying on type cues. When explicit type information is incorporated, as in Zhou et al. (2023), performance patterns shift: while Hits@10 often remain competitive—or even surpass GraIL on sparse datasets like YAGO21K-610 —Hits@1 consistently decline. This indicates that explicit types may mitigate sparsity by providing useful semantic signals, but also introduce complexity that reduces precision in top-ranked predictions. As dataset density increases, the performance gap between GraIL and type-informed models narrows, and in some cases, GraIL even outperforms the latter. This trend suggests that **explicit type information becomes less helpful—and potentially detrimental—in denser graphs**, where structural cues are already sufficient. In contrast, implicit type information, as leveraged by TyleR, generally leads to more robust and consistent improvements. While not always achieving the best Hits@1, models using implicit types (TyleR variants) rank first or second across most datasets for both Hits@10 and Hits@1. These models show particular strength on sparse datasets, such as YAGO21K-610, where the gap in Hits@10 is most pronounced. This suggests that **implicit typing is more robust against topological variations in the data**, exhibiting a higher generalization potential.

*Addressing **RQ1**, the benefit of explicit type information is dataset-dependent. It aids relational inference in sparse graphs lacking rich topology, but can add detrimental complexity in denser graphs. Implicit type signals, however, consistently enhance inference, particularly by mitigating structural sparsity.*

5.2 Usefulness of PLM Representations for Implicit Type Signal (RQ2)

The results in Table 2 provide compelling evidence that PLMs can significantly enhance node representations in subgraph-based link prediction. However, **the impact of PLMs is not uniform across all datasets**. For example, on FB237-V1 and FB237-V3, RoBERTa-L and Llama3-8B models exhibit competitive performance, especially in terms of Hits@1 and MRR, suggesting that PLMs provide a strong inductive bias for relational reasoning. In contrast, models without PLMs, like GraIL, show lower performance on these datasets, particularly regarding the Hits@10 metric. This highlights the ability of PLMs to generalize and make more accurate predictions in larger, more complex graphs, where non-PLM models may struggle. Table 3 shows the impact of different aggregation strategies on the FB237-V1 dataset, with SUM showing

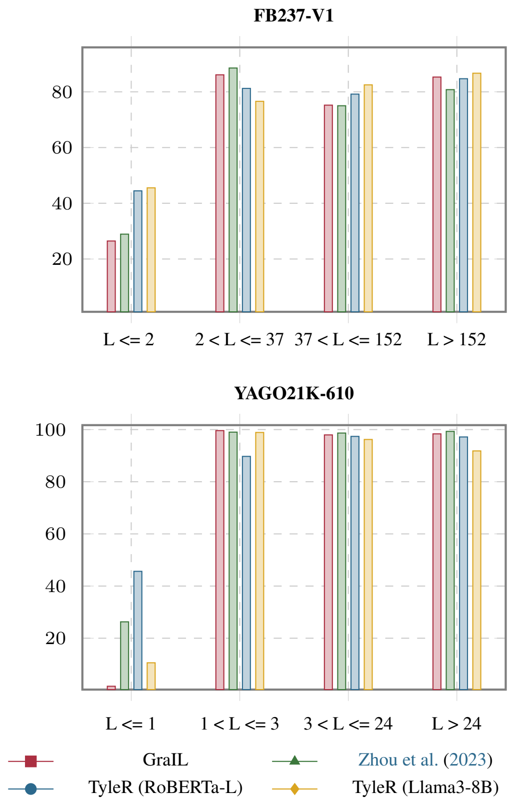

Figure 2: Link Prediction (Hits@10) evaluation under **varying structural sparsity conditions** (i.e., the number of edges L in the enclosing subgraph of the target triple) on FB237-V1 (top) and YAGO21K-610 (bottom).

the most consistent results. For example, Llama3-8B with SUM outperforms GraIL across all metrics. In addition, we compare the results of different aggregation strategies with the scenario where only the "type" prompt is considered (i.e., p_1 in Table 1), showing consistent improvements.

*Regarding **RQ2**, PLMs effectively enhance node representations for subgraph-based inductive link prediction. By providing richer semantic features, models like RoBERTa-L and Llama3-8B improve relational inference. Aggregating diverse PLM-derived semantic embeddings (i.e., from different prompts) boosts representation expressiveness.*

5.3 Effect on Type and Structural Sparsity (RQ3)

To investigate the effect of our approach on *type sparsity*, we categorize entities into four groups based on the number of explicit type annotations they possess. Group 0 consists of entities with *no explicit type*. Group 1 includes entities with *exactly one* explicit type. The remaining entities (those with *more than one* type) are split into two additional groups based on the median number of types in this subset: the lower 50% form the group labeled ">1 (1st)", and the upper 50% form ">1 (2nd)". For each test triple, we determine the group membership of the *known* entity (irrespective of the type information available for the candidate entities) and report the model performance in Figure 3. We report the average Hits@10 across the three FB237 variants for each group. Group 0 represents the most type-sparse setting with entities lacking any explicit type; in this scenario, both variants of TyleR consistently outperform the typeless baseline GraIL across all dataset variants. This reinforces the intuition that **type signals derived from PLMs can enhance inference capabilities in sparse settings**. For instance, on FB237-V1, TyleR (RoBERTa-L) yields a 6.15% relative improvement over the non-PLM baselines, while on FB237-V2, it achieves an even greater gain of 10.75% over GraIL. Interestingly, in the case of entities with multiple types, TyleR continues to outperform the explicit-type-based method of Zhou et al. (2023). This suggests that while explicit type information is useful, its effectiveness may diminish when type annotations are noisy or overly numerous, highlighting the need for better strategies to aggregate multiple type signals.

We further analyze the role of *implicit type information* in addressing *structural sparsity*. For this analysis, we consider the two datasets with the lowest graph density: YAGO21K-610 and FB237-V1. Evaluation triples are grouped into four bins based on the number of edges in their enclosing subgraphs, using percentiles to capture varying levels of sparsity. Figure 2 presents Hits@10 across these structural sparsity conditions. The results indicate that PLM-based approaches, particularly those using RoBERTa-L, demonstrate strong performance in extremely sparse subgraphs. For example, RoBERTa-L performs best in scenarios with one or fewer edges (for YAGO21K-610) and more than 152 edges (for FB237-V1), demonstrating its robustness at both ends of the sparsity spectrum. However, in *moderate sparsity* settings (e.g., $2 < L \leq 37$ in FB237-V1), models such as GraIL and Zhou et al. (2023) perform comparably or better, due to their reliance on structural patterns that are still informative in such contexts.

*Answering **RQ3**, PLM-based approaches such as TyleR address both type and structural sparsity. They consistently outperform baselines in scenarios with minimal explicit type information or sparse*

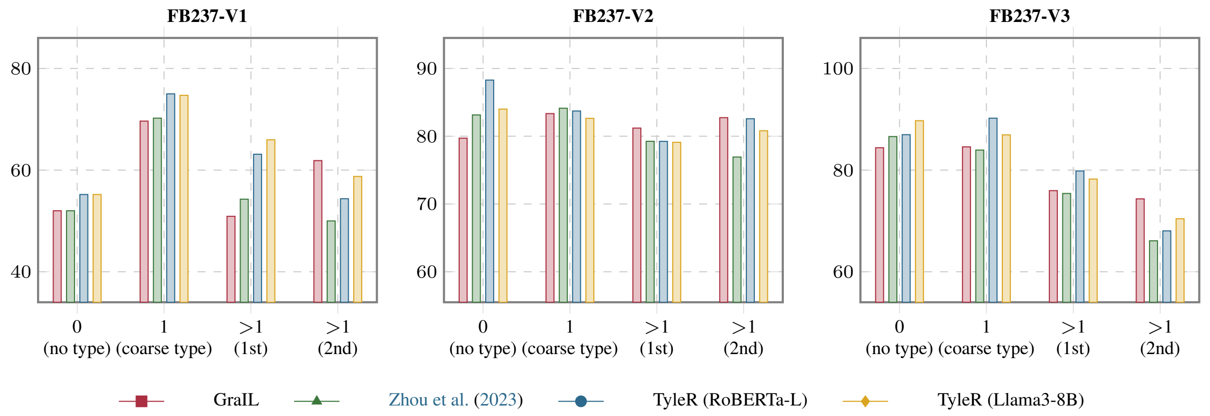

Figure 3: Hits@10 performance across four **type sparsity** groups for three FB237 variants, computed according to the number of explicit types linked to each entity (details in Section 5.3). The groups, from left to right, represent scenarios with an increasing number of explicit types associated with the known entity.

subgraph structures by inferring meaningful semantics from PLMs. Although challenges persist with moderate sparsity and noisy types, PLMs show significant potential.

6 Related Work

Inductive Link Prediction. Inductive Link Prediction (ILP) in Knowledge Graphs (KGs) aims to infer missing links that involve entities unseen during training, thereby enabling models to generalize to evolving KGs. Unlike traditional embedding-based models (Lin et al., 2015; Bordes et al., 2013; Wang et al., 2014), inductive methods explicitly handle unseen entities. Early approaches relied on rule-based reasoning (Yang et al., 2017; Meilicke et al., 2018), but graph neural networks (GNNs) soon became dominant (Hamilton et al., 2017), with GraIL (Teru et al., 2020) leveraging enclosing subgraph structures for relational inference. Extensions include CoMPILE (Mai et al., 2021), emphasizing relational directionality, and TACT (Chen et al., 2021), introducing relation-level reasoning. Zhou et al. (2023) incorporate ontological data, but assume complete type information, an assumption that seldom holds in real-world KGs.

Entity Representation with Language Models. Pre-trained language models (PLMs) capture factual and relational knowledge from large corpora (Petroni et al., 2019; Brown et al., 2020), encoding rich entity semantics (Zhu et al., 2024) and retrieving factual information via prompting (Wei et al., 2023). This makes PLMs well-suited for link prediction because they can enrich entity representations. For example, KGBERT (Yao et al., 2019) verbalizes triples as text and fine-tunes BERT to classify their plausibility. Subsequent methods (Zhang et al., 2020; Daza et al., 2021; Wang et al., 2021) integrate entity descriptions into KG completion to induce embeddings for new entities via PLMs. BERTRL (Zha et al., 2022) exemplifies this trend by injecting GNN-discovered reasoning paths into a BERT-based model. A promising direction involves integrating LLMs with subgraph-based methods to reduce model queries while preserving structural reasoning. Li et al. (2025) propose CATS, a hybrid model that leverages latent type cues and neighbor facts to fine-tune an LLM for triple scoring, combining semantic understanding with explicit subgraph evidence. Unlike prior approaches that fine-tune PLMs, our method extracts semantic knowledge from a frozen PLM, and we investigate how effectively such pre-trained models enable a subgraph-reasoning module to capture the type semantics underlying each relation.

7 Conclusion

We present TyleR, a novel inductive link-prediction approach designed to handle incomplete or noisy type information. By leveraging pre-trained language models (PLMs), TyleR enriches node representations with implicit type signals, overcoming the limitations of methods reliant on explicit annotations. Experiments show that TyleR exhibits competitive performance, particularly when type data are sparse or unreliable. The results underscore the potential of PLMs for semantic enrichment, enabling robust link prediction without complete type supervision. Future work will examine domain-specific PLMs, more embedding-aggregation strategies, and broader applications to graph-based tasks.

Limitations

Our study employs a set of predefined prompts, which, while effective for the scope of our experiments, may not represent the most informative or optimal configurations. More sophisticated strategies for adaptive prompt selection or prompt tuning could potentially enhance model performance. Exploring these approaches is left as a direction for future research. Additionally, the hyperparameters for our models were selected empirically, based on extensive experimentation and informed judgment. While this approach yielded strong results, it may not guarantee optimal configurations. A more systematic or exhaustive hyperparameter search could lead to improved outcomes. Nonetheless, the computational cost and complexity associated with such procedures, particularly given the scale and resource demands of our training setup, render them infeasible within the constraints of this study.

References

Lei Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. 2016. Layer normalization. *CoRR*, abs/1607.06450.

Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *NIPS*, pages 2787–2795.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *NeurIPS*.

Jiajun Chen, Huarui He, Feng Wu, and Jie Wang. 2021. Topology-aware correlations between relations for inductive link prediction in knowledge graphs. In *AAAI*, pages 6271–6278. AAAI Press.

Daniel Daza, Michael Cochez, and Paul Groth. 2021. Inductive entity representations from text via link prediction. In *WWW*, pages 798–808. ACM / IW3C2.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT (1)*, pages 4171–4186. Association for Computational Linguistics.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, and 82 others. 2024. The llama 3 herd of models. *CoRR*, abs/2407.21783.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and GPT-2 embeddings. In *EMNLP/IJCNLP (1)*, pages 55–65. Association for Computational Linguistics.

Aryo Pradipta Gema, Dominik Grabarczyk, Wolf De Wulf, Piyush Borole, Javier Antonio Alfaro, Pasquale Minervini, Antonio Vergari, and Ajitha Rajan. 2023. Knowledge graph embeddings in the biomedical domain: Are they useful? A look at link prediction, rule learning, and downstream polypharmacy tasks. *CoRR*, abs/2305.19979.

William L. Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *NIPS*, pages 1024–1034.

Shibo Hao, Bowen Tan, Kaiwen Tang, Bin Ni, Xiyan Shao, Hengzhe Zhang, Eric P. Xing, and Zhiting Hu. 2023. Bertnet: Harvesting knowledge graphs with arbitrary relations from pretrained language models. In *ACL (Findings)*, pages 5000–5015. Association for Computational Linguistics.

Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard de Melo, Claudio Gutierrez, Sabrina Kirrane, José Emilio Labra Gayo, Roberto Navigli, Sebastian Neumaier, Axel-Cyrille Ngonga Ngomo, Axel Polleres, Sabbir M. Rashid, Anisa Rula, Lukas Schmelzeisen, Juan Sequeda, Steffen Staab, and Antoine Zimmermann. 2021. *Knowledge Graphs*. Synthesis Lectures on Data, Semantics, and Knowledge. Morgan & Claypool Publishers.

Muzhi Li, Cehao Yang, Chengjin Xu, Zixing Song, Xuhui Jiang, Jian Guo, Ho-fung Leung, and Irwin King. 2025. Context-aware inductive knowledge graph completion with latent type constraints and subgraph reasoning. In *AAAI*, pages 12102–12111. AAAI Press.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *AAAI*, pages 2181–2187. AAAI Press.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Sijie Mai, Shuangjia Zheng, Yuedong Yang, and Haifeng Hu. 2021. Communicative message passing for inductive relation reasoning. In *AAAI*, pages 4294–4302. AAAI Press.

Christian Meilicke, Manuel Fink, Yanjie Wang, Daniel Ruffinelli, Rainer Gemulla, and Heiner Stuckenschmidt. 2018. Fine-grained evaluation of rule- and embedding-based systems for knowledge graph completion. In *ISWC (1)*, volume 11136 of *Lecture Notes in Computer Science*, pages 3–20. Springer.

Matthew E. Peters, Mark Neumann, Robert L. Logan IV, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. 2019. Knowledge enhanced contextual word representations. In *EMNLP/IJCNLP (1)*, pages 43–54. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander H. Miller. 2019. Language models as knowledge bases? In *EMNLP/IJCNLP (1)*, pages 2463–2473. Association for Computational Linguistics.

Jay Pujara, Eriq Augustine, and Lise Getoor. 2017. Sparsity and noise: Where knowledge graph embeddings fall short. In *EMNLP*, pages 1751–1756. Association for Computational Linguistics.

Andrea Rossi, Denilson Barbosa, Donatella Firmani, Antonio Matinata, and Paolo Merialdo. 2021. Knowledge graph embedding for link prediction: A comparative analysis. *ACM Trans. Knowl. Discov. Data*, 15(2):14:1–14:49.

Michael Sejr Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *ESWC*, volume 10843 of *Lecture Notes in Computer Science*, pages 593–607. Springer.

Komal K. Teru, Etienne G. Denis, and William L. Hamilton. 2020. Inductive relation prediction by subgraph reasoning. In *ICML*, volume 119 of *Proceedings of Machine Learning Research*, pages 9448–9457. PMLR.

Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, and Partha P. Talukdar. 2020. Composition-based multi-relational graph convolutional networks. In *ICLR*. OpenReview.net.

Shuyao Wang, Yongduo Sui, Chao Wang, and Hui Xiong. 2024. Unleashing the power of knowledge graph for recommendation via invariant learning. In *WWW*, pages 3745–3755. ACM.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2021. KEPLER: A unified model for knowledge embedding and pre-trained language representation. *Trans. Assoc. Comput. Linguistics*, 9:176–194.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *AAAI*, pages 1112–1119. AAAI Press.

Yanbin Wei, Qiushi Huang, Yu Zhang, and James T. Kwok. 2023. KICGPT: large language model with knowledge in context for knowledge graph completion. In *EMNLP (Findings)*, pages 8667–8683. Association for Computational Linguistics.

Keyulu Xu, Chengtao Li, Yonglong Tian, Tomohiro Sonobe, Ken-ichi Kawarabayashi, and Stefanie Jegelka. 2018. Representation learning on graphs with jumping knowledge networks. In *ICML*, volume 80 of *Proceedings of Machine Learning Research*, pages 5449–5458. PMLR.

Fan Yang, Zhilin Yang, and William W. Cohen. 2017. Differentiable learning of logical rules for knowledge base reasoning. In *NIPS*, pages 2319–2328.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. KG-BERT: BERT for knowledge graph completion. *CoRR*, abs/1909.03193.

Hanwen Zha, Zhiyu Chen, and Xifeng Yan. 2022. Inductive relation prediction by BERT. In *AAAI*, pages 5923–5931. AAAI Press.

Zhiyuan Zhang, Xiaoqian Liu, Yi Zhang, Qi Su, Xu Sun, and Bin He. 2020. Pretrain-kge: Learning knowledge representation from pretrained language models. In *EMNLP (Findings)*, volume EMNLP 2020 of *Findings of ACL*, pages 259–266. Association for Computational Linguistics.

Wentao Zhou, Jun Zhao, Tao Gui, Qi Zhang, and Xuanjing Huang. 2023. Inductive relation inference of knowledge graph enhanced by ontology information. In *EMNLP (Findings)*, pages 6491–6502. Association for Computational Linguistics.

Yuqi Zhu, Xiaohan Wang, Jing Chen, Shuofei Qiao, Yixin Ou, Yunzhi Yao, Shumin Deng, Huajun Chen, and Ningyu Zhang. 2024. Llms for knowledge graph construction and reasoning: recent capabilities and future opportunities. *World Wide Web (WWW)*, 27(5):58.

Appendix

This appendix provides supplementary material to support the main paper. It is organized as follows:

- **Dataset Details (Appendix A)**: Provides a summary table (Table 4) and further information on the datasets used in our experiments. The section includes the procedure for extracting ontology graphs and entity-type links for the FB237 variants, which initially lack such annotations. It also details the methodology for splitting ontology triples into training, validation, and test sets.

- **Hyperparameter Details (Appendix B)**: Outlines the hyperparameter settings employed for training our proposed model, TyleR, as well as the baseline models. Key parameters such as learning rates, number of hops for subgraph extraction, embedding dimensions, and early stopping criteria are specified to ensure reproducibility.

- **Examples of Predictions (Appendix C)**: Presents a qualitative example (Table 5) comparing the link predictions made by TyleR and baseline models for a specific target triple, particularly in a scenario with a sparse enclosing subgraph. This section illustrates how different models rank candidate entities and highlights the impact of the strict tie-breaking strategy.

- **Embedding Visualization (Appendix D)**: Includes a 2D visualization of entity embeddings (obtained via PCA). This offers a qualitative insight into the learned representations and their spatial distribution for a sample set of entities (Figure 4 and Figure 5).

A Dataset Details

Table 4 provides a statistical overview of the datasets utilized in our experiments, detailing their key characteristics, including the number of entities, relations, triples, types, meta-relations, ontology triples, type links, and textual labels.

The model from Zhou et al. (2023) relies on explicit entity-type pairs and an ontology graph for training. FB237 initially lacks these annotations. Therefore, we processed the FB237 variants to extract the necessary type information and construct a corresponding ontology using the following procedure.

To construct the ontology graph for our experiments we mapped all the freebase entities appearing in the dataset to their Wikidata identifier, using the publicly available Freebase-Wikidata mappings [1]. Using the public Wikidata API [2], we then retrieved for every mapped entity its respective textual label and the values associated with its "*instance of*" property, which indicates the type(s) an entity is associated to. With the set of relevant concepts established, we constructed the schema-level ontology. For each concept identified in the previous step, its full set of concepts was fetched from Wikidata.

A schema-level triple $\langle$Concept$_1$ PropertyLabel Concept$_2\rangle$ was generated and added to our onthology graph if, and only if, the target value of a Concept ($Concept_2$) was itself one of the recognized concepts.

In the entity triples, the entities in the test set do not appear in the train set and valid set, while the relations in both the test set and valid set are included in the train set. We train on the train graph and test on the test graph. In addition, to achieve ontology training, we randomly divide the ontology triples into a train set, a valid set, and a test set using hold-out splitting in the ratio of 80%, 10%, 10%, respectively.

B Hyperparameter Details

Baselines are trained using the hyperparameter settings reported in their original papers. For our model, we adopt the configuration from Zhou et al. (2023) to ensure fair comparison, tuning only the learning rate, which we set empirically to 1e-3. All models are trained for 50 epochs with early stopping (patience of 100 iterations) and a batch size of 16. We adopt the Adam optimizer. For all models, the number of hops in the enclosing subgraph is 3. We set the semantic embedding dimension to 24, the layer-0 embedding dimension to 32, and the margin γ in the loss function to 10.

C Examples of Predictions

This section provides a qualitative example to illustrate the behavior of TyleR in comparison to baseline models, particularly in challenging scenarios characterized by extreme structural sparsity. We focus on a specific instance from the YAGO21K-610 dataset where the enclosing subgraph for the

[1] https://developers.google.com/freebase
[2] https://www.wikidata.org/w/api.php

Dataset	Split	Entities	Relations	Triples	Types	Meta Rel.	Onto. Triples	Type Links	Text Labels
	train	1594	180	4245	458	29	680	2163	1516
fb237_v1	valid	567	103	489	124	15	86	756	539
	test	550	102	492	113	13	85	764	517
	train	1093	142	1993	458	29	680	1525	1041
fb237_v1_ind	valid	287	66	206	124	15	86	406	275
	test	301	68	205	113	13	85	434	289
	train	2608	200	9739	575	33	865	3586	2489
fb237_v2	valid	1139	143	1166	160	15	109	1511	1083
	test	1142	140	1180	153	16	108	1515	1094
	train	1660	172	4145	575	33	865	2257	1561
fb237_v2_ind	valid	548	92	469	160	15	109	757	516
	test	562	107	478	153	16	108	745	524
	train	3668	215	17986	732	31	1060	5114	3484
fb237_v3	valid	1882	183	2194	196	17	133	2575	1787
	test	1871	179	2214	192	16	133	2520	1773
	train	2501	183	7406	732	31	1060	3426	2379
fb237_v3_ind	valid	973	120	866	196	17	133	1275	920
	test	981	128	865	192	16	133	1290	924
	train	16357	30	30000	610	24	1983	4861	16357
YAGO21K-610	valid	4388	21	3000	166	14	248	1783	4388
	test	3938	25	6970	159	13	248	1898	3938

Table 4: Statistics of the datasets used in our experiments. The YAGO21K-610 (Zhou et al., 2023) dataset includes ontology triples and entity-type links, while the FB237 dataset variants (Teru et al., 2020) are further processed to extract ontology triples, type links and textual labels.

target triple lacks any connecting edges.

Table 5 presents the top-ranked predictions for the target triple (`Christos Kagiouzis`, `isAffiliatedTo`, `Kastoria F.C.`), where the task is to predict the tail entity (`Kastoria F.C.`). This triple was chosen because its 3-hop enclosing subgraph presents a worst-case scenario for structural reasoning. Specifically, the subgraph contains no path that could link the head entity (`Christos Kagiouzis`) to the correct tail entity (`Kastoria F.C.`), beside the target link. This lack of structural information within the subgraph presents a significant challenge for models that heavily rely on graph patterns. The evaluation follows the standard protocol (Section 4.2), where the correct tail entity is ranked against 50 randomly corrupted negative samples. Crucially, as detailed in Section 4.2, ranking employs the strict tie-breaking strategy, assigning the worst possible rank to the positive triple in case of score ties.

C.1 Analysis

TyleR (RoBERTa-L). Despite the absence of direct structural paths in the enclosing subgraph, TyleR ranks the correct entity (`Kastoria F.C.`) 2nd. This strong performance is attributed to its ability to leverage rich semantic information derived from the PLM (RoBERTa-L). The PLM's understanding of entities and their likely affiliations, learned from vast text corpora, allows TyleR to infer plausible connections even when explicit graph structure is missing. The top-ranked entity, (`Southern United FC`), is also a football club, indicating that TyleR correctly identifies the semantic category of plausible tail entities for the relation (`isAffiliatedTo`) with (`Christos Kagiouzis`) (likely a footballer). The scores assigned by TyleR are relatively distinct, suggesting a higher degree of confidence in its ranking.

GraIL. In contrast, GraIL, which relies purely on subgraph structures for relational inference, performs poorly. It ranks the correct entity (`Kastoria F.C.`) at 50th (last among the 50 candidates considered for ranking this positive triple). The identical scores for all top 50 entities (all -11.888) indicate that GraIL cannot differentiate between the candidates due to the lack of structural cues in the enclosing subgraph. This highlights a key limitation of purely structural methods in extremely sparse settings.

Zhou et al. (2023). This model, which incorporates explicit type information and ontology reasoning, ranks the correct entity 16th. While this is significantly better than GraIL, it falls short of

Rank	Triple	Score
	TyleR (RoBERTa-L)	
1	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Southern United FC	-1.951
2	**Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Kastoria F.C. (*gold*)**	**-1.953**
3	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Deltras F.C.	-1.985
4	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Yunnan Hongta F.C.	-1.998
5	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Chainat Hornbill F.C.	-4.957
6	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Hoàng Anh Gia Lai F.C.	-5.172
7	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Great Britain women's Olympic football team	-5.529
8	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ APEP F.C.	-5.604
9	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Basketball League Belgium	-5.698
10	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Baltimore Blast (1980–92)	-5.706
	GraIL	
41	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Darko Vukić	-11.888
42	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Connecticut Pride	-11.888
43	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Hoàng Anh Gia Lai F.C.	-11.888
44	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Southern United FC	-11.888
45	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Helgi Sigurðsson	-11.888
46	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Conor Powell	-11.888
47	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Samuel Cunningham (footballer)	-11.888
48	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Ferdinand Daučík	-11.888
49	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Ertan Demiri	-11.888
50	**Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Kastoria F.C. (*gold*)**	**-11.888**
	Zhou et al. (2023)	
10	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ SC 07 Bad Neuenahr	4.330
11	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Chicago Power	4.330
12	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Baltimore Blast (1980–92)	4.330
13	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Peristeri B.C.	4.330
14	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Deltras F.C.	4.330
15	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ ADET	4.330
16	**Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Kastoria F.C. (*gold*)**	**4.330**
17	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Łukasz Tumicz	-6.757
18	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Ertan Demiri	-6.757
19	Christos Kagiouzis $\rightarrow$ `isAffiliatedTo` $\rightarrow$ Ferdinand Daučík	-6.757

Table 5: Example of ranking predictions on the YAGO21K-610 dataset for the target triple (`Christos Kagiouzis, isAffiliatedTo, Kastoria F.C.`), when the tail is to be predicted. In this case, the target triple has no links in the associated enclosing subgraph. As discussed in Section 4.2, ranking is done using the strict tie-breaking strategy.

TyleR's performance. The explicit type information likely provides some signal ("(`Kastoria F.C.`) is a *Club*"). However, this explicit information might be coarser-grained or less directly informative for this specific prediction compared to the nuanced semantic representations captured by TyleR. The presence of many ties in the scores (e.g., ranks 10-16 all have score 4.330) suggests that while types help narrow down possibilities, they do not offer the same fine-grained discriminative power as TyleR's PLM-based semantic enrichment in this particular sparse scenario.

This example underscores the advantage of TyleR's approach, particularly its semantic enrichment stage using PLMs. By infusing node representations with implicit type-aware signals, TyleR can effectively reason about entity relationships even when the local graph structure is uninformative, thereby mitigating the challenges posed by structural sparsity.

D Embedding Visualization

This section provides a qualitative analysis of entity embeddings through 2D visualization to illustrate how different models represent candidate entities in a challenging link prediction task characterized by structural sparsity. We utilize Principal Component Analysis (PCA) to project the final-layer GNN embeddings $\mathbf{h}_v^L$ of 50 candidate tail entities onto a 2D plane. The specific task visualized is predicting the missing tail entity for the triple `<Andrei Gashkin, playsFor, ?>` from the YAGO21K-610 dataset. Notably, this example is chosen for its extreme structural sparsity. The enclosing subgraph constructed around the head entity `Andrei Gashkin` and the correct tail entity

FC KAMAZ Naberezhnye Chelny is very sparse. Furthermore, for many of the 49 negative candidate entities considered alongside the correct tail, their respective enclosing subgraphs (when considered with the head Andrei Gashkin) also lack rich structural information, making it difficult for models relying heavily on graph patterns to make accurate distinctions. We compare the embeddings generated by:

- The ontology-enhanced model from Zhou et al. (2023), which leverages explicit type information (Figure 4).
- Our proposed model, TyleR (RoBERTa-L), which uses PLM-derived implicit type signals (Figure 5).

D.1 Analysis

Figure 4 visualizes the PCA-projected embeddings from the model by Zhou et al. (2023). In this visualization:

- The correct tail entity, FC KAMAZ Naberezhnye Chelny (highlighted or labeled distinctly if possible in the actual figure), is positioned among a cluster of other football clubs and sports-related entities. For instance, it might be spatially close to other entities like SV Grödig or Egri FC if they were among the candidates.
- The embeddings of many semantically similar entities (e.g., various football clubs) are tightly clustered. This suggests that while the explicit type information used by this model (e.g., "Football Club" type) helps group entities by their broad category, it may not provide sufficient fine-grained discriminative power in this structurally sparse scenario.
- The model appears to struggle to clearly distinguish FC KAMAZ Naberezhnye Chelny from other plausible (same-type) but incorrect candidate entities based solely on the explicit type signals and the limited structural information available in the sparse subgraph. The representation reflects a general categorical understanding rather than a nuanced, context-specific one for the playsFor relation with Andrei Gashkin.

Figure 5 displays the PCA-projected embeddings from our TyleR-RoBERTa-L model for the same set of 50 candidate entities.

- The correct tail entity, FC KAMAZ Naberezhnye Chelny, is noticeably more separated in the embedding space compared to its representation in Figure 4. While it would still likely be in a region associated with sports entities, its position relative to other incorrect candidate football clubs is more distinct.
- This improved separation suggests that TyleR's semantic enrichment, derived from RoBERTa-L, provides more nuanced and discriminative features. The model benefits from the implicit propagation of semantic information related to the head entity Andrei Gashkin (a known footballer) through the PLM's understanding.
- The PLM's pre-trained knowledge helps infer a more fine-grained "type-awareness" and contextual understanding for the playsFor relation. Even with sparse explicit graph structure, TyleR can leverage the rich semantics encoded by the PLM (and potentially GNN mechanisms like self-loop connections that reinforce entity identity) to better characterize and differentiate the correct tail entity.

This visual comparison underscores the benefit of TyleR 's approach in handling structurally sparse scenarios. The ontology-enhanced model (Zhou et al. (2023)), while utilizing explicit types, produces less distinguishable embeddings for semantically similar entities when graph structure is poor. In contrast, TyleR, by incorporating rich implicit type signals from a pre-trained language model, achieves a more fine-grained characterization and better separation of the correct entity in the embedding space. This highlights the potential of PLM-derived semantic enrichment to compensate for deficiencies in explicit type annotations and structural connectivity, leading to more robust inductive link prediction. This supports our paper's argument that implicit type signals enable a more nuanced understanding, particularly crucial in sparse settings.

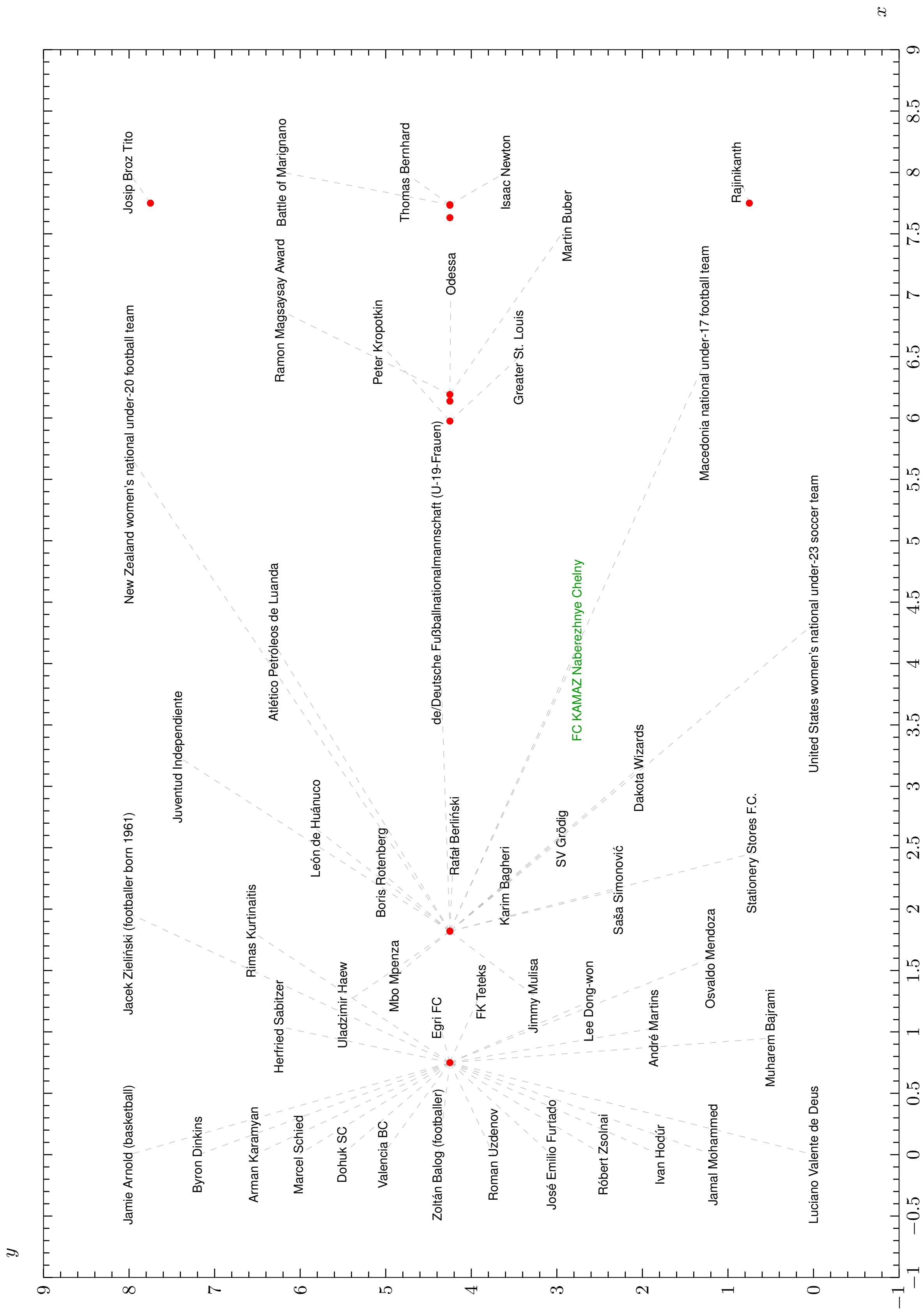

Figure 4: Visualization of last layer embeddings (using PCA) for the ontology-enhanced model of Zhou et al. (2023) for 50 candidate entities when predicting the missing tail for triple `<Andrei Gashkin, playsFor, ?>`. For all the 50 candidates, there is no enclosing subgraph.

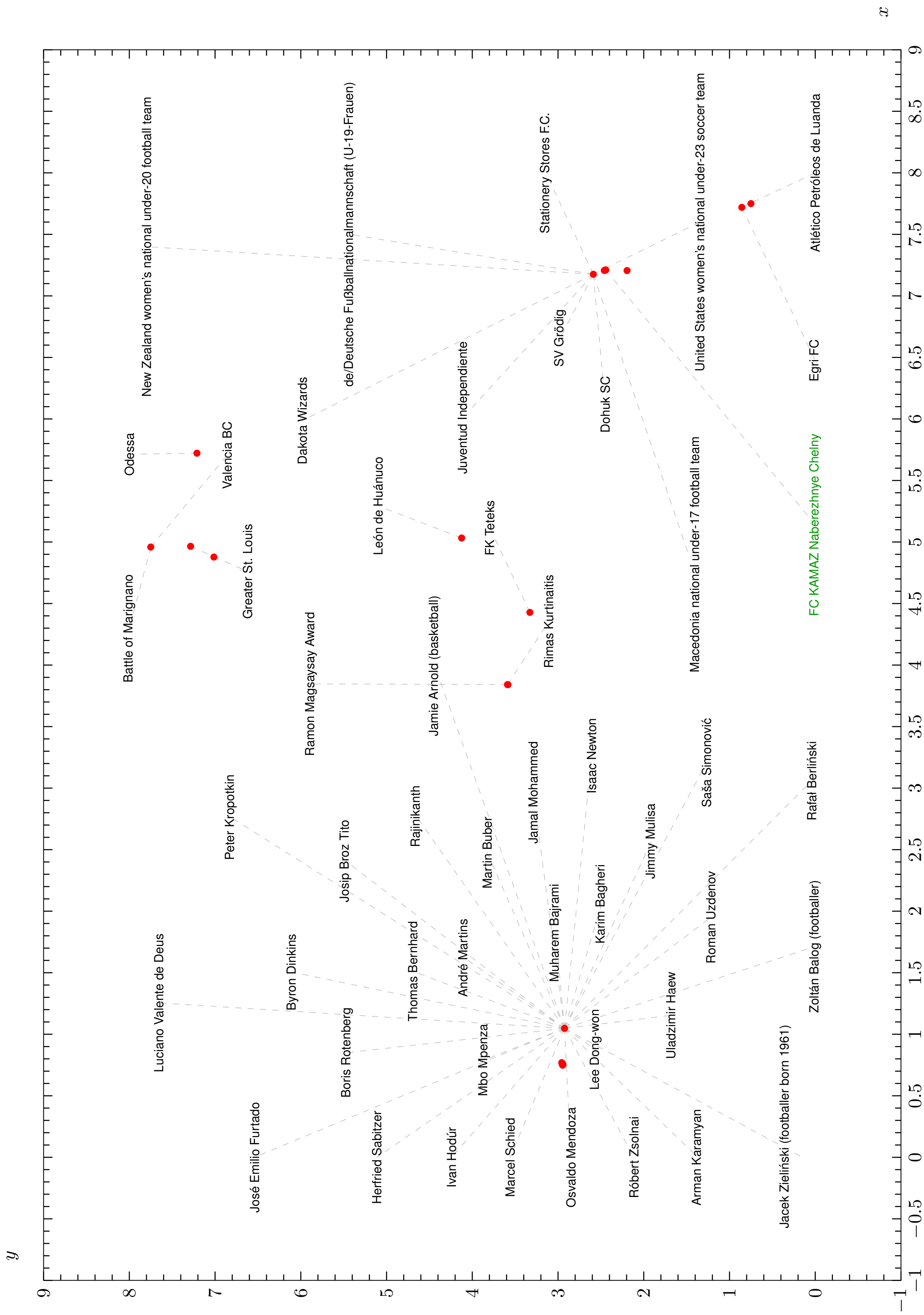

Figure 5: Visualization of last layer embeddings (using PCA) for TyleR (RoBERTa-L) for 50 candidate entities when predicting the missing tail for triple <Andrei Gashkin, playsFor, ?>. For all the 50 candidates, there is no enclosing subgraph.