TaxoTrans: Taxonomy-Guided Entity Translation

Zhuliu Li
LinkedIn
Mountain View, California, USA
zhuli@linkedin.com

Yiming Wang
LinkedIn
Mountain View, California, USA
yimwang@linkedin.com

Xiao Yan
LinkedIn
Mountain View, California, USA
xyan@linkedin.com

Weizhi Meng
LinkedIn
Mountain View, California, USA
wmeng@linkedin.com

Yanen Li
LinkedIn
Mountain View, California, USA
yali@linkedin.com

Jaewon Yang
LinkedIn
Mountain View, California, USA
jeyang@linkedin.com

ABSTRACT

Taxonomies describe the definitions of entities, entities' attributes and the relations among the entities, and thus play an important role in building a knowledge graph. In this paper, we tackle the task of taxonomy entity translation, which is to translate the names of taxonomy entities in a source language to a target language. The translations then can be utilized to build a knowledge graph in the target language. Despite its importance, taxonomy entity translation remains a hard problem for AI models due to two major challenges. One challenge is understanding the semantic context in very short entity names. Another challenge is having deep understanding for the domain where the knowledge graph is built.

We present TaxoTrans, a novel method for taxonomy entity translation that can capture the context in entity names and the domain knowledge in taxonomy. To achieve this, TaxoTrans creates a heterogeneous graph to connect entities, and formulates the entity name translation problem as link prediction in the heterogeneous graph: given a pair of entity names across two languages, TaxoTrans applies a graph neural network to determine whether they form a translation pair or not. Because of this graph, TaxoTrans can capture both the semantic context and the domain knowledge. Our offline experiments on LinkedIn's skill and title taxonomies show that by modeling semantic information and domain knowledge in the heterogeneous graph, TaxoTrans outperforms the state-of-the-art translation methods by ∽ 10%. Human annotation and A/B test results further demonstrate that the accurately translated entities significantly improves user engagements and advertising revenue at LinkedIn.

CCS CONCEPTS

• **Information systems** → **Data mining**; • **Computing methodologies** → *Knowledge representation and reasoning*.

KDD '22, August 14–18, 2022, Washington, DC, USA.

ACM ISBN 978-1-4503-9385-0/22/08…$15.00
https://doi.org/10.1145/3534678.3539188

KEYWORDS

Taxonomy, Entity Translation, Knowledge Graph, Graph Neural Network

ACM Reference Format:
Zhuliu Li, Yiming Wang, Xiao Yan, Weizhi Meng, Yanen Li, and Jaewon Yang. 2022. TaxoTrans: Taxonomy-Guided Entity Translation. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '22), August 14–18, 2022, Washington, DC, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3534678.3539188

1 INTRODUCTION

Knowledge graphs are graphs representing knowledge in terms of the relation between entities. By using a standard format of (source entity, relation, destination entity), knowledge graphs provide a unified way to organize knowledge. For this reason, knowledge graphs are used in many online places to curate and represent useful knowledge to users. Google built the Google Knowledge Graph [1], Microsoft built Microsoft Satori [18] and LinkedIn built LinkedIn Economic Graph [2].

Knowledge graphs contain *taxonomies*. A taxonomy is a dictionary of a set of important entities where we describe for each entity its name, definition and relations to other entities. Taxonomies are used as a source of truth for defining the entities, categorizing them, and classifying their relations [24, 33, 53]. For example, Commerce companies such as Amazon use product category taxonomy [32] to categorize different products; professional networks such as LinkedIn use taxonomies to organize professional titles, skills and organizations [29].

In this paper, we aim to study translating taxonomy entities across different spoken languages. Given a taxonomy in one language where we have entity names and entity relations, we aim to translate the entity names to other languages (as illustrated by Figure 1 (a)). Such taxonomy entity translation is one of the most important steps when we aim to build a knowledge graph for a new language. First, the translated taxonomies can be used to guide user inputs for the users in the new locale. For example, if we translate a taxonomy from English to Chinese, we can use it to create a typeahead for Chinese users. Second, we can use the translated taxonomy to better understand the user input. For example, if our goal is to understand the position description that a Chinese user wrote, we can use the Chinese title taxonomy translated from English to identify the best matching title.

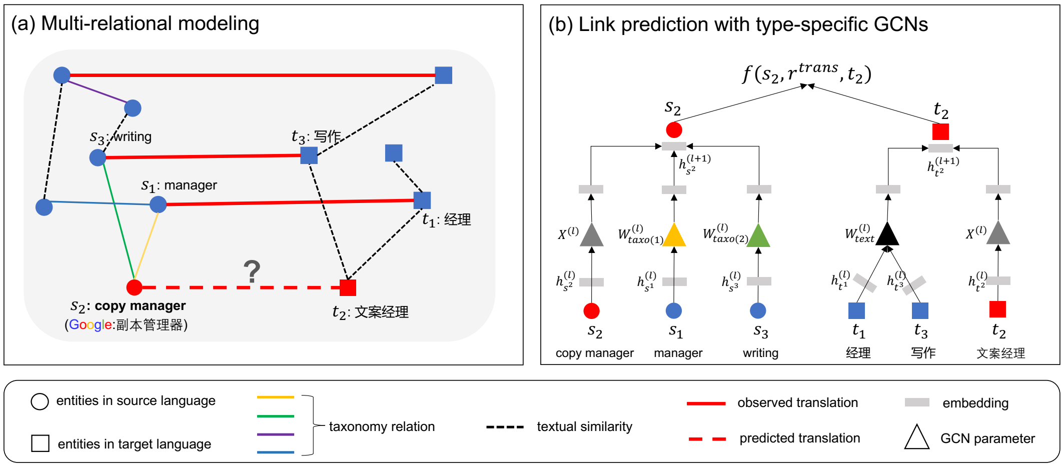


Figure 1: Schema of TaxoTrans. (a) The entity name translation task is modeled as a link prediction problem in a heterogeneous graph which integrates the domain knowledge from taxonomy relations (colored edges) and the textual information from entity names (black edges). The bilingual contextual information is encoded as node features. Information is also allowed to propagate across languages through the known translations (red edges). (b) The link prediction problem is solved by a GNN architecture composed with type-specific GCN branches. Different taxonomy edges have different message passing parameters (colored triangles). The textual similarity edges in the source and target languages share the same parameters (black triangles).

Translating entity names in a taxonomy is a very challenging task. First challenge is scalability: there are usually hundreds of thousands or millions of entities to translate, and new entities keep emerging. This challenge makes human translation infeasible. Second challenge is domain knowledge: taxonomies are created for a particular domain (e.g., medical domain, professional domain) and translation requires deep understanding for the domain. Third one is understanding the context: the context is very important but it is very hard to capture due to the short length of input. For example, "copy" in a title "copy manager" means writing creative content (copyright), not an act of replicating. However, there is no contextual information to infer this meaning in the title name.

We note that existing methods have shortcomings to tackle the problem of translating taxonomy entities. First category of existing methods are neural machine translation methods, such as Google Translate [47]. They are trained with long text in a large corpus, thus it is difficult for them to capture the domain-specific context in short entity names. For example, Google Translate translates the word "copy" in title "copy manager" to "副本" (replicate) in Chinese. Second kind of existing methods are cross-lingual word embeddings where one learns word embedding for multiple languages in the same vector space. These methods can be applied to a domain specific text corpus, but their limitation is that word embeddings can only give us word-to-word translation. They would not understand that "copy" in "copy manager" and "copy clerk" mean different things. Third kind of existing methods are cross-lingual knowledge graph alignment methods. They can be applied to domain specific taxonomies, and they can capture the domain-specific context from the knowledge graph. For example, they can understand that title "copy manager" has a different meaning for "copy" by understanding that "creative writing" is a related skill. Their limitation is that they require a high quality knowledge graph in a target language. If the knowledge graph in the target language is not available or incomplete, these methods will have limited efficacy. Given that we want to translate to a target language to *build* a knowledge graph in that language, this is a serious drawback.

In this paper, we propose a novel method to translate entity names in the taxonomy with capturing the domain knowledge and the context. Given a taxonomy in a source language, we aim to translate entity names to a target language. We formulate a link prediction problem where we create one heterogeneous graph with entities in the source and target languages, and then predict the links between entities across the languages. In the source language, we create the graph by using the taxonomy, to represent relations among entities. In the target language, we extract entities from a domain-specific text corpus and create edges if they are likely to co-occur in the same sentence. To match entities in source and target languages, we use Graph Neural Networks that is inspired by the previous R-GCN [36] architecture. Similar to R-GCN, our method can encode different kinds of relations in the taxonomy.

Our offline experiments on translating LinkedIn's skill and title taxonomy entities show that TaxoTrans achieves superior performance compared to the state-of-the-art competitors which are widely used in the related fields: machine translation, cross-lingual word embedding and cross-lingual graph alignment. The online crowdsourcing experiments and A/B testing further confirm the accuracy of the domain-specific translations generated by TaxoTrans.

Contributions: 1) We introduce an entity translation method TaxoTrans, which can capture the domain knowledge in taxonomy and semantic relations among entities. 2) We model multi-aspect information from both source and target languages as one heterogeneous graph, and solve the entity translation problem as link prediction using GNN. 3) We comprehensively evaluate and analyze the performance of TaxoTrans, and demonstrate its capability of producing accurate translations of taxonomy entities.

2 RELATED WORK

There are three categories of work that also applicable to translate taxonomy entities.

Neural machine translation (NMT) converts text from a source language to a target language based on the encoder-decoder [13, 26, 41] neural architecture, in which an encoder network converts the source text to a fixed-length representation called "context vector", and a decoder network generates the target text from that representation. Attention mechanism [5, 8, 31, 47] is introduced between the encoder and decoder to tackle the drawback that the single context vector from the encoder network is not enough to represent a long sentence with complicated semantic information. Recently, more flexible and scalable self-attention-based models [11, 37, 42, 43] by fully relying on attention mechanisms become a dominant in NMT. However, none of these methods are able to capture the domain-specific context in the short entity names due to two reasons: 2) these methods do not use the domain knowledge in taxonomy; 1) these methods are trained with long text in a large corpus. An example will be shown in Section 5.5 is that the word "runner" in entity "food runner " is translated to "亚军" (runner-up) in Chinese by NMT due to the lack of domain-specific context.

Cross-lingual word embedding represents the words from the source and target languages in a shared vector space, and has been used for bilingual lexicon induction [16, 19, 22] and cross-lingual information retrieval [14, 30, 30, 45]. The most widely adopted approach is to align the pre-trained monolingual word embeddings in different language in the shared space, based on the bilingual supervision such as dictionaries or parallel corpora [3, 17, 38, 48]. Recently, methods [4, 12, 14, 20, 23] alleviating the bilingual supervision received great attention. One representative work is MUSE [14] which proposes to learn a mapping from a source to a target space through adversarial training. All these methods are applicable to learn cross-lingual entity embeddings by combining the word embeddings. However, they are only trained for word-level alignment rather that entity-level alignment, and are also not aware of the structural information encoded in taxonomy.

Cross-lingual knowledge graph alignment aims to align entities across different monolingual knowledge graphs. Over the last few years, two categories of embedding-based approaches become popular for entity alignment. One category of methods [10, 21, 28, 39, 40, 54] are built on top of TransE [7]. Given two KGs with some known matches, these methods project all the entities and relations onto a shared vector space with the relational information preserved. The second category of methods [9, 35, 46, 49, 52] are based on graph neural network which show superior performance to the TransE based methods. The pioneering work is [46], where the entity embeddings in different graphs are generated by different GCNs; the embeddings in the last GCN-layers are aligned by minimizing a loss function. Though all of these methods are applicable to translate entities in a source graph given a complete target graph, their efficacy are severely limited in case the target graph is incomplete or non-existent. For example, we only have one taxonomy graph in the source language while lacking curated relations among the target entities.

3 PROBLEM DEFINITION

In this section, we define the problem of translating taxonomy entities. Given a taxonomy in a source language, the goal is to translate the names of taxonomy entities to a target language with respect to the domain knowledge in taxonomy and the contextual information in the entity names. The input and output of the learning task are summarized as follows.

- **Input:** 1) Taxonomy in a source language, which contains a set $\mathcal{S}$ of entities, as well as relations between the entities. 2) Set $\psi = \{(s,t)|s \in \mathcal{S}_l, t \in \mathcal{T}\}$ of known translations where $\mathcal{S}_l \subset \mathcal{S}$, and $\mathcal{T}$ is the entity set in a target language.
- **Output:** Set $\phi = \{(s,t)|s \in \mathcal{S}_u, t \in \mathcal{T}\}$ of learned translations where $\mathcal{S}_u = \mathcal{S} \backslash \mathcal{S}_l$.

4 PROPOSED METHOD

In this section, we introduce our model TaxoTrans: Taxonomy-Guided Entity Translation, which effectively utilizes multi-aspect knowledge from the input data. We first explain how the domain knowledge in taxonomy and bilingual contextual information in entity names are integrated as a heterogeneous graph with multi-relational edges, and how the domain-specific translation is guaranteed. Next, we propose to formulate the entity translation problem as link prediction in the heterogeneous graph, which is solved with a graph neural network (GNN) inspired by the R-GCN [36] architecture.

4.1 Multi-relational Modeling

To integrate the prior knowledge from the taxonomy as well as the contextual information in the entity names, we propose to combine the sets $\mathcal{S}$, $\mathcal{T}$ and ψ defined in Section 3 as a heterogeneous graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{R})$ where $\mathcal{V}$, $\mathcal{E}$ and $\mathcal{R}$ are the sets of entities, edges and edge types respectively. The heterogeneous $\mathcal{G}$ is illustrated by Figure 1 (a) and elaborated as follows.

Entities (nodes): The entity set $\mathcal{V}$ is composed with set $\mathcal{S}$ of taxonomy entities in the source language, and set $\mathcal{T}$ of candidate entities in the target language. In some application domains, target entities can be easily obtained. For example, if $\mathcal{S}$ is a set of LinkedIn taxonomy entities, set $\mathcal{T}$ can be constructed with LinkedIn user-inputs written in a target language on member profiles. Alternatively, for application domains where target entities do not exist, named-entity recognition (NER) can be applied to tag entities from the text corpus in a target language. As will be shown in the experiments, our method outperforms state-of-the-art competitors by training with either user-input target entities or NER-tagged target entities.

Links (Edges): We encode multi-aspect dependencies between the entities in the edge set $\mathcal{E}$. A pair of entities $v_i, v_j \in \mathcal{V}$ can be connected by multiple types of edges. Each edge $e \in \mathcal{E}$ has a type $r \in \mathcal{R}$ which is in one of the three categories: 1) r^{taxo}: taxonomy relations between entities with $r^{taxo(k)} \in r^{taxo}$ as its k-th subtype (denoted as one edge color in Figure 1 (a)). 2) r^{text}: textual similarity of entity names. For a pair of entities $v_i, v_j \in \mathcal{V}$ in the same language, we first average the fastText [6, 25] embeddings of all the words in the entity names as entity embeddings $\mathbf{h}_i$ and $\mathbf{h}_j$, and then draw an edge with type r^{text} between v_i and v_j if and only if $\frac{\mathbf{h}_i^T \mathbf{h}_j}{||\mathbf{h}_i|| \, ||\mathbf{h}_j||} > \epsilon$ where ϵ is a thresholding hyperparameter. 3) r^{trans}: observed entity translations between source and target languages.

Entity features: To boost the translation performance, we also extract the contextual information from entity names as entity features. We expect the features of the source and target entities to be informative to identify the source-target pairs. One possibility is to apply the exiting cross-lingual word embedding methods to project the words in the source and target entities onto a shared embedding space, then merge these word embeddings as entity feature vectors. However, an obvious flaw is that these cross-lingual word embedding methods are only aware of the word-level or character-level information, while do not utilize the contextual information in the entity names. Instead, to incorporate the contextual information in the entity features, we first translate all the entities in $\mathcal{T}$ to the source language using Google Translate [47] which is supposed to capture the contextual information in majority of the entities. Next, for every entity $v \in \mathcal{V}$, we use the fastText embedding of its names in the source language as the feature vector.

With the multi-aspect knowledge base represented as a heterogeneous graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{R})$, we propose to model the entity translation problem as link prediction in $\mathcal{G}$. Given a source entity $s \in \mathcal{S}_u$, the goal is to predict the scores in $\{f(s, r^{trans}, t) | \forall t \in \mathcal{T}\}$, where $f(s, r^{trans}, t)$ represents how likely t is a translation of s. Finally, the pair (s, t^*) is added into the output set ϕ where

$$t^* = \arg\max_{t \in \mathcal{T}} f(s, r^{trans}, t). \tag{1}$$

In Section 4.2, we propose to use graph neural network to learn the scoring function $f(\cdot)$.

4.2 Link Prediction with Graph Neural Network

Encouraged by the superior performance of GNN-based methods [9, 35, 46, 49, 52] over the traditional TransE-based graph embedding methods [10, 21, 39, 54] on the cross-lingual entity alignment problem, we also apply GNN to solve the link prediction for entity translation.

The message passing in the l-th layer of a Graph Convolutional Network (GCN) [27] can be summarized as the following two steps: 1) entities aggregate information from their neighbors to produce the intermediate representations; 2) apply both linear and non-linear transformations on the aggregation as the updated representations in the $(l+1)$-th layer. Given the adjacency matrix A and degree matrix D of a graph $\mathcal{G}$ with n entities, the convolution operation in the l-th layer is as follows

$$H^{(l+1)} = \Phi((I + D^{-\frac{1}{2}} A D^{-\frac{1}{2}}) H^{(l)} W^{(l)}), \tag{2}$$

where $H^{(l)} \in \mathbb{R}^{n \times d^{(l)}}$ contains the length $d^{(l)}$ entity embeddings in the l-th layer; $W^{(l)} \in \mathbb{R}^{d^{(l)} \times d^{(l+1)}}$ is the parameter matrix; I is the identity matrix; $\Phi(\cdot)$ is the non-linear activation function. The role of Equation (2) is to aggregate information from neighbors and also retain the entity embeddings in the current layer.

Though existing entity alignment methods have achieved promising results with the classical GCN model, their performances are hindered by several limitations as summarized below.

- They do not directly model the edge type information in the message passing stage of the GCN. Instead, some of these methods propose to indirectly consider the edge type information. For example, [52] represents the edge types as N-hot vectors as entity features, and [46] introduces the edge type information by re-weighting the adjacency matrix, which are not effective strategies to utilize the edge type information.
- They use two independent GCNs to handle the graphs in source and target languages respectively, and both GCNs are only interacted through the loss function in the output layer. Such setup might not be sufficient to allow information to propagate across different languages.
- They are only trained to predict the translation edges between the languages without predicting other types of edges in the heterogeneous graph. In the scenario that labeled translation edges are scarce, highly likely such training strategy can only reach to some local sub-networks surrounding the training entities, while can not utilize the global information of the graph.

To address the aforementioned issues, we propose to directly model the edge type information in the GNN architecture, which also allows information to effectively propagate across different languages and maximizes the utility of the global graph topology. We follow the R-GCN [36] idea to use a type-specific GCN for the message passing between the entity pairs which are connected by a specific edge type; then combine all the type-specific GCNs together. As illustrated by Figure 1 (b), to generate the hidden representation $\mathbf{h}_v^{(l+1)}$ of an entity $v \in \mathcal{V}$ in the $(l+1)$-th layer, we first create a computation graph composed with multiple GCN branches corresponding to multiple edge types. Each branch contains one type-specific GCN with a parameter matrix $W_r^{(l)}$ propagating the messages from the neighbors of v through the edge type r. The hidden representation $\mathbf{h}_v^l$ of entity v in the l-th layer is also retained by an additional GCN branch with parameter matrix $X^{(l)}$. The convolution operation in the GNN is formulated as

$$\mathbf{h}_i^{(l+1)} = \Phi(X^{(l)} \mathbf{h}_i^{(l)} + \sum_{r \in \mathcal{R}} \sum_{j \in \mathcal{N}_r^i} \alpha_r^{ij} W_r^{(l)} \mathbf{h}_j^{(l)}),$$

where $\mathcal{N}_r^i$ denotes the neighbor set of entity v_i under edge type r, and α_r^{ij} is a scalar which quantifies the importance of the massage passed from entity v_j to v_i via edge type r. We set α_r^{ij} uniformly as $\frac{1}{|\mathcal{N}_r^i|}$ since we concentrate on the multi-relational modeling, which however can be more sophisticate and learnable via introducing attention mechanisms [44]. In contrast to the featureless setup in the original R-GCN paper, we encode the input feature vector of entity v_i as $\mathbf{h}_i^{(0)}$. We find that explicitly providing entity features to graph neural network allows it to leverage contextual information in entity names in addition to the topological information in the

Table 1: Statistics of Experimental Data

	Offline Experiment	Online Experiment			
	EN -> ZH	EN -> FR	EN -> ES	EN -> PT	EN -> JA
Taxonomy Edge	60k+	70k+	70k+	70k+	70k+
Textual Edge	100k+	500k+	400k+	500k+	200k+
Translation Edge	20k+	40k+	20k+	30k+	20k+
Entity	50k+	300k+	200k+	300k+	100k+

graph, which results in better performance and prevents overfitting as will be shown in Section 5.3.

With the length d embeddings $\mathbf{h}_i$ and $\mathbf{h}_j$ of an entity pair $v_i,\ v_j \in \mathcal{V}$ in the last GNN layer, the likelihood of entities v_i and v_j are connected by an edge with type r is computed as $f(v_i, r, v_j) = \sigma(\mathbf{h}_i^T D_r \mathbf{h}_j)$ where $D_r \in \mathbb{R}^{d\times d}$ is a diagonal parameter matrix which increases the model freedom with respect to edge types. $\sigma(\cdot)$ denotes the sigmoid function. The model is trained through minimizing the cross-entropy loss function with negative sampling as follows

$$\mathcal{L} = \sum_{(v_i,r,v_j)\in\mathcal{G}} (-\log(f(v_i,r,v_j) - \sum_{t=1}^{k} \mathbb{E}_{v_t \sim P_r(j)} \log(1 - f(v_i,r,v_t)), \tag{3}$$

to encourage the score of the observed edge (v_i, r, v_j) to be higher than the score of a random pair (v_i, v_t) sampled from a noise distribution P_r [34]. With the learned scoring function $f(\cdot)$, the translations from source entities to target entities can be inferred by ranking their association scores according to Equation (1).

5 EXPERIMENTS

In the experiments, the performance of TaxoTrans is evaluated using the in-house dataset from LinkedIn. Section 5.1 describes the preparation of taxonomy data and the extraction of target entities from the text corpus in professional domain. Section 5.2 introduces the compared baseline methods. Section 5.3 shows the offline comparison between TaxoTrans and the baselines on translating taxonomy entities. Section 5.4 analyzes TaxoTrans from different angles to justify its novelties. Section 5.5 conducts the case study where TaxoTrans outperforms neural machine translation by a large margin on some difficult entities. Finally, Section 5.6 further validates the translations generated by TaxoTrans with online experiments in LinkedIn.

5.1 Data Preparation

We construct the heterogeneous graph described in Section 4.1 using LinkedIn's in-house data.

Taxonomy: LinkedIn's skill and title taxonomies are fundamental components of LinkedIn's economic graph, which power multiple LinkedIn's core products like job recommendations, search typeahead, course recommendations, etc [50, 51]. With the skill and title taxonomies, we construct multiple types of taxonomy edges between the entities (titles and skills in English) based on their classifications and semantic relations in taxonomy.

Target entities: As described in Section 4.1, we adopt two strategies to collect target entities: 1) We collect the member-input titles and skills as entities in the target languages, which are logged in member profiles, job postings, uploaded resumes, and search records in LinkedIn. 2) We applied SpaCy[1] library to extract named-entities from the profile pages of LinkedIn members. *Unless otherwise mentioned in the experiments, target entities are collected from member-input as default.*

Ground truth labels: The training and test source-target entity pairs (translations) are generated by internal annotators in LinkedIn.

According to the resource availability, we conduct the offline and online experiments on different datasets at different scales. For the offline comparisons, we translate both the skill and title taxonomy entities in English (EN) to Chinese (ZH). For the online comparisons, we validate the translations of one type (title or skill) of taxonomy entities. The skill entities are translated from English to French (FR) and Spanish (ES), and the title entities are translated from English to Portuguese (PT) and Japanese (JA). The statistics of the heterogeneous graphs in the offline and online experiments are summarized in Table 1. In the online A/B testing, a broader range of i18n languages are tested.

5.2 Baseline Methods

In the offline experiments, we compare TaxoTrans with five baseline methods which are widely used in the closely related areas including machine translation, cross-lingual word embedding and cross-lingual graph alignment. The methods are summarized as follows.

- **GCN:** The classical GCN model defined by Equation (2), which uses the same entity features as in TaxoTrans while ignores the edge type information. In [46] GCN was was first used for cross-lingual knowledge graph alignment.
- **Multi-Aspect GCN:** In [52], GCN is applied for cross-lingual alignment of multi-relational knowledge graphs. The idea is to encode edge types and node indices as N-hot multi-aspect entity features. Following this idea, we create a baseline by concatenating the N-hot edge type vector and the entity features of TaxoTrans, and using GCN for message passing.
- **Featureless R-GCN:** As proposed in the previous R-GCN paper [36], the entity features are removed from TaxoTrans. Instead, the input layer $H^{(0)}$ is replaced with a trainable embedding layer.

[1] https://spacy.io/

Table 2: Results using member-input target entities.

Method	*MRR*	*Hits*@1	*Hits*@10
TaxoTrans	**+19.0%**	**+19.7%**	**+16.2%**
GCN	+12.5%	+10.1%	+15.1%
Multi-Aspect GCN	+10.4%	+7.8%	+13.6%
MUSE	-35.4%	-38.5%	-28.5%
Featureless R-GCN	-43.9%	-48.5%	-34.6%
NMT+fastText	- -	- -	- -

Table 3: Results using NER-tagged target entities.

Method	*MRR*	*Hits*@1	*Hits*@10
TaxoTrans	**+12.4%**	**+9.6%**	**+15.8%**
GCN	+4.2%	-1.2%	+12.0%
Multi-Aspect GCN	-4.3%	-11.6%	+7.9%
Featureless R-GCN	-60.1%	-58.9%	-71.2%
NMT+fastText	- -	- -	- -

- **NMT+fastText:** The association scores defined in Equation (1) are computed by only using the input features of TaxoTrans, which are generated by neural machine translation (Google Translate) and fastText as explained in Section 4.1.
- **MUSE:** A multi-lingual word embedding library [14] which supports both supervised training based on iterative Procrustes alignment and unsupervised adversarial training. We apply MUSE to obtain the word-level embeddings which are then averaged as the entity embeddings.

For fair comparison, we let all the GNN-based methods share the same neural architecture and hyper-parameters with 2 convolutional layers, 300 hidden units in each layer, and a dropout rate of 0.1. All the GNN models are optimized with Adam. We implement MUSE using the its own Python package[2] with fastText word embeddings and a bilingual dictionary as input. The Chinese entities are segmented to match with the vocabulary in MUSE. Since we observed the predictions of unsupervised MUSE are close to random, we only show the results of its supervised version.

5.3 Offline Performance Comparison

To compare TaxoTrans and the five baseline methods, we randomly choose 80% of the observed translation edges for training, and measure the performances on the rest 20% of edges. We use $Hits@k$ and MRR (mean reciprocal rank) as the performance metrics to assess the translation accuracy, both of which are commonly used for measuring the ranking performance. Denoting $\mathcal{S}_{test}$ as the set of test entities in the source language, for every $s_i \in \mathcal{S}_{test}$ we find the rank of its true translation (s_i, r^{trans}, t^*) in the scoring set $\{f(s_i, r^{trans}, t) | \forall t \in \mathcal{T}_l\}$ as rank(s_i), where $\mathcal{T}_l \subset \mathcal{T}$ is the set of labeled target entities. The two metrics are then computed as

$$Hits@k = \frac{1}{|\mathcal{S}_{test}|} \sum_{s_i \in \mathcal{S}_{test}} \mathbb{1}(\text{rank}(s_i) \leq k),$$
$$MRR = \frac{1}{|\mathcal{S}_{test}|} \sum_{s_i \in \mathcal{S}_{test}} \frac{1}{\text{rank}(s_i)}.$$

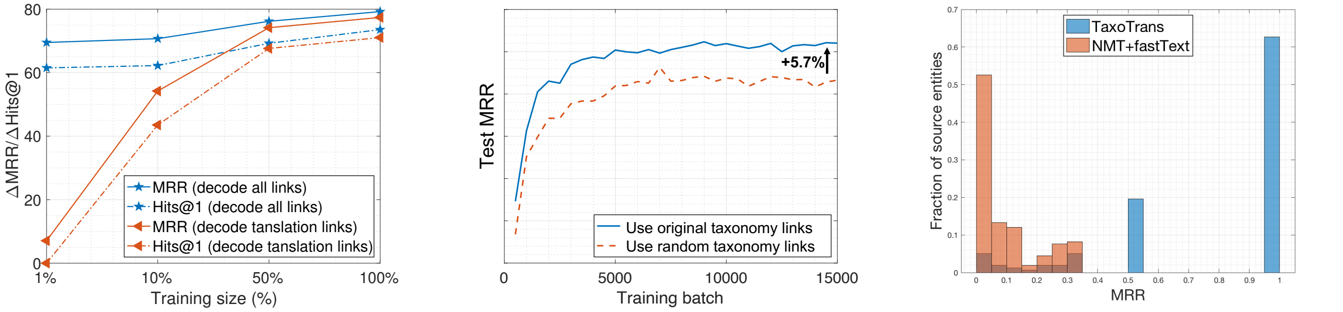
In all the results of this paper, we convert $Hits@k$ and MRR to percentage. The results of all the methods are compared in Table 2 (The relative amount of improvement over NMT+fastText are shown due to proprietary reasons.), from which we observe that TaxoTrans consistently outperforms all the baseline methods with higher MRR and $Hits@k$ scores. The comparisons justify several key advantages regarding the multi-relational modeling and GNN architecture in TaxoTrans: 1) the comparison of TaxoTrans, NMT+fastText and MUSE confirms that the heterogeneous graph integrating domain knowledge in taxonomy as well as textual information in entity names is helpful to optimize the cross-lingual entity embeddings; 2) the comparison of TaxoTrans, GCN and Multi-Aspect GCN implies that the neural architecture constructed with edge type specific GCN branches more effectively utilizes the relational information in the knowledge graph; 3) the comparison of TaxoTrans and Featureless R-GCN suggests that the entity features contain rich bilingual contextual information remarkably boost the accuracy of entity translation. We also compared the performances of all the GNN-based methods trained with NER-tagged target entities. TaxoTrans still achieves superior performance as shown in Table 3. Note that since target entities found by NER tagger have a smaller coverage of titles and skills compared to member-input entities, we observe smaller overall performance lift in Table 3 than in Table 2; The performances of NMT+fastText and MUSE are invariant to the target extraction strategies since they do not use graph information.

5.4 Analysis

In this section, we analyze the robustness of TaxoTrans in the label-scarce scenario, and also justify the importance of taxonomy edges in the heterogeneous graph.

5.4.1 Performance on Label-Scarce Scenario. We randomly sample 1%, 10%, 50% and 100% of translation edges from the training set and measure the performance on the same test set as in Section 5.3. We compare the performances of TaxoTrans using two different decoding ideas: 1) decoding all the edges in the heterogeneous graph as proposed in Equation (3); 2) only decoding the translation edges between the languages as adopted by existing graph alignment methods [35, 46, 52]. In Figure 2 (a), TaxoTrans with two different decoding strategies are compared by varying the training size (ΔMRR and $\Delta Hits@k$ to the first data point in the "$Hits$@1 - decode translation links group" are shown in the y-axis for proprietary reasons). We observe that the performance of TaxoTrans with the proposed decoding strategy is stable across all the training percentages. TaxoTrans still achieves an acceptable accuracy even in the label-scarce situation with only 1% of training translations. In contrast, the performance with the second decoding strategy is deteriorated significantly as the amount of available

[2] https://ai.facebook.com/tools/muse/



(a) Performance w.r.t. label scarcity. **(b) Justification of taxonomy edges** **(c) Performance on difficult cases.**

Figure 2: Analysis of TaxoTrans. (a) Two different decoding strategies are compared by varying the training size. (b) Comparison between using original and random taxonomy edges. (c) Performance distribution on entities that are difficult to translate.

training edges decreases. Such observations validate our assumptions that decoding all the graph edges allows better knowledge sharing between two different languages, and optimizes the utility of the global information in the knowledge graph.

5.4.2 Justification of Taxonomy Edges. We justify the importance of taxonomy edges in the heterogeneous graph by measuring the performance of TaxoTrans on random taxonomy edges. To generate the random taxonomy edges, we swap the edges among the taxonomy entities randomly without changing the node degrees. Such strategy ensures the degree distribution in the original graph preserved in the random graph. The test *MRR* scores in the first 15,000 training batches using 1% of training translations are shown in Figure 2 (b). We observe that TaxoTrans using the original taxonomy edges achieves ~ 6% performance lift compared to using the random edges. This observation confirms that the taxonomy edges encode domain knowledge and rich semantic relations between the entities are crucial for the improvement of translation accuracy.

5.5 Case Study

Table 2 has shown that TaxoTrans outperforms NMT+fastText by ~ 20% in *MRR* and *Hits*@1. Now we zoom in on the hard cases where NMT+fastText fails with *MRR*<50%. Figure 2 (c) compares the performance distributions of both methods on the hard cases. We observe that TaxoTrans correctly translates majority of the difficult entities with significantly higher overall *MRR* scores.

Next, we also visualize the embeddings of five source-target entity pairs in the hard cases as examples. We apply Multidimensional Scaling (MDS) [15] dimension reduction to project the pairwise similarity matrix onto two dimensions. For a pair of source and target entities (s, t), their similarity by TaxoTrans is computed as $f(s, r_{trans}, t)$ as in Equation (1) using embeddings from the output layer of the GNN; their similarity by NMT+fastText is computed as the cosine similarity between their feature vectors. Figure 3(a) shows that TaxoTrans embeds the matched entity pair closely and divides the unmatched entity pair widely from each other. In contrast, Figure 3 (b) shows that the embeddings generated by NMT+fastText do not reflect the mapping relations between the source and target entities. The reason is because Google Translate wrongly translates the source entities to phrases in non-professional domains due to the lack of domain-specific context on these short entities.

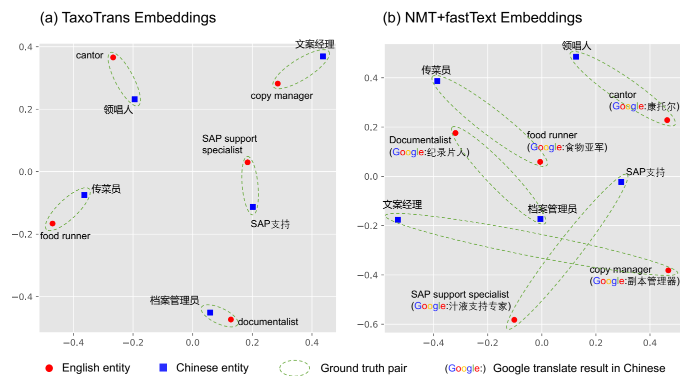


Figure 3: Visualization of the difficult cases. (a) The entity embeddings generated by TaxoTrans. (b) The entity embeddings generated by NMT+fastText.

5.6 Online Experiments

To test the performance of TaxoTrans, we build a two-step pipeline to translate the LinkedIn taxonomy in English to other i18n languages. Step 1 is to apply TaxoTrans to generate translation scores $\{f(s, r^{trans}, t) | \forall s \in \mathcal{S}, \forall t \in \mathcal{T}\}$, and output the top-ranked translations as candidates; Step 2 is to identify the correct translations among the candidates from step 1, which is achieved by human annotation from both external and internal linguists.

The human annotation workflow is summarized as follows: For each entity and candidate translations in target languages, we send them as a query to the Appen[3] platform for external linguists to select the correct translations. The linguists can choose the option of "No Match" if none of these translations are accurate. The selected translations are further examined by the internal linguists, to confirm if the translated phrases agree with the semantic concepts in the professional domain.

[3] https://appen.com/

Table 4: Human annotation results on title and skill entities. Each entry presents a tuple of three performance metrics: *Precision*, *Coverage*, *Throughput*.

	Skill Translation		Title Translation	
	EN -> FR	EN -> ES	EN -> PT	EN -> JA
TaxoTrans	+3.0%, +2.0%, +15.3%	+8.0%, +1.1%, +3.6%	+7.0%, +2.9%, +5.0%	+2.0%, +3.3%, +3.4%

Table 5: Human-validated novel prediction. The numbers of newly identified target aliases, and the percentage of one-to-many translations are shown.

	Skill Translation		Title Translation	
	EN -> FR	EN -> ES	EN -> PT	EN -> JA
New Alias	7k+	14k+	13k+	2k+
One-to-Many (%)	54.7	53.2	48.1	37.6

5.6.1 Comparison with Current Product. The performance of the new pipeline is compared with our current product where in step 1 TaxoTrans is replaced with Google Translate and Microsoft Bing Translator. We hold out 20% of translated source entities as the test set. The human annotation results are evaluated by following metrics.

- *Throughput*: The percentage of queries where Appen finds a match.
- *Precision*: The percentage of translations selected by Appen that are accepted by the internal linguists.
- *Coverage*: The percentage of total taxonomy entities that appear in the translations accepted by the internal linguists.

In the comparison, we assess the performances on translating the English skill entities to French and Spanish, as well as translating the English title entities to Portuguese and Japanese, according to the available resources. The remarkably improved *Throughput*, *Precision* and *Coverage* scores of TaxoTrans shown in Table 4 validate that comparing with NMT-based translators, TaxoTrans can more accurately translate a larger size of taxonomy entities to the professional domain.

5.6.2 Novel Translations from TaxoTrans. We further demonstrate the value of our new product by counting the newly identified title and skill aliases that are not found by the current product. Remarkably, the numbers in Table 5 show that the new product relying on TaxoTrans successfully finds 20k+ new skill aliases and 15k+ new title aliases from the four target languages.

Another interesting observation made by us is the capability of TaxoTrans to generate one-to-many translations; this is an advantage over the existing translators which only support one-to-one translations. Table 5 shows the percentage of source entities that are matched to multiple crowdsourcing-validated target key phrases, which demonstrates that one-to-many translations of the taxonomy entities can be derived by ranking the translation scores from TaxoTrans.

Table 6: Online A/B testing results.

	Title Coverage	Profile View	Message Sent	Ad Revenue
TaxoTrans	+2.3%	+3.1%	+3.9%	+1.1%

5.6.3 Online A/B testing. New title alias found by TaxoTrans are injected into the current LinkedIn title taxonomy. We observe widespread impacts on 11M+ LinkedIn members by online A/B testing, which is conducted with 50% of the LinkedIn traffic for 14 days.

The results in Table 6 show that the new title aliases significantly improve the coverage of entities in LinkedIn title taxonomy. For example, the Chinese entity "档案管理员" (Documentalist) is now curated in the new taxonomy. Moreover, with the increased amount of entities covered by title taxonomy, the member profiles become more discoverable (eg. by recruiter search), and more engagement (eg. the number of messages sent) are observed among members. All of these factors jointly lead to a remarkable increase in the advertising revenue of LinkedIn.

6 CONCLUSIONS

Translating the names of taxonomy entities is challenging due to the difficulty of understanding the semantic context in short entity names and the domain where taxonomy is constructed. In this study, we introduced a novel GNN-based method TaxoTrans which can accurately translate taxonomy entities by modeling the domain knowledge in taxonomy, and the bilingual contextual information in entity names as a heterogeneous graph. Our comprehensive offline evaluations show that by effectively utilizing the multi-aspect knowledge in the heterogeneous graph, TaxoTrans can achieve state-of-the-art performance on generating domain-specific entity translations. Our online experiments demonstrate that TaxoTrans can be utilized to expand the taxonomy of a large-scale social network, and has significant impact on the user engagement and Ad revenue.

In the future, we plan to generalize TaxoTrans to the multilingual scenario and which can enable the domain knowledge and semantic relations in taxonomies to be propagated across multiple languages.

REFERENCES

[1] [n.d.]. Google Knowledge Graph. https://en.wikipedia.org/wiki/Google_Knowledge_Graph
[2] [n.d.]. LinkedIn Economic Graph. https://economicgraph.linkedin.com/
[3] Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2016. Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In *EMNLP*. 2289–2294.
[4] Mikel Artetxe and Holger Schwenk. 2019. Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Beyond. *TACL* 7 (2019), 597–610.

[5] Dzmitry Bahdanau, Kyung Hyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

[6] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *TACL* 5 (2017), 135–146.

[7] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Neural Information Processing Systems (NIPS)*. 1–9.

[8] Denny Britz, Anna Goldie, Minh-Thang Luong, and Quoc Le. 2017. Massive Exploration of Neural Machine Translation Architectures. In *EMNLP*. 1442–1451.

[9] Yixin Cao, Zhiyuan Liu, Chengjiang Li, Juanzi Li, and Tat-Seng Chua. 2019. Multi-Channel Graph Neural Network for Entity Alignment. In *ACL*. 1452–1461.

[10] Muhao Chen, Yingtao Tian, Mohan Yang, and Carlo Zaniolo. 2017. Multilingual knowledge graph embeddings for cross-lingual knowledge alignment. In *IJCAI*. 1511–1517.

[11] Mia Xu Chen, Orhan Firat, Ankur Bapna, Melvin Johnson, Wolfgang Macherey, George Foster, Llion Jones, Mike Schuster, Noam Shazeer, Niki Parmar, et al. 2018. The Best of Both Worlds: Combining Recent Advances in Neural Machine Translation. In *ACL*. 76–86.

[12] Xilun Chen and Claire Cardie. 2018. Unsupervised Multilingual Word Embeddings. In *EMNLP*.

[13] Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. In *EMNLP*.

[14] Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. In *ICLR*.

[15] Michael AA Cox and Trevor F Cox. 2008. Multidimensional scaling. In *Handbook of data visualization*. Springer, 315–347.

[16] Long Duong, Hiroshi Kanayama, Tengfei Ma, Steven Bird, and Trevor Cohn. 2016. Learning Crosslingual Word Embeddings without Bilingual Corpora. In *EMNLP*. 1285–1295.

[17] Manaal Faruqui and Chris Dyer. 2014. Improving vector space word representations using multilingual correlation. In *EACL*. 462–471.

[18] Yuqing Gao, Jisheng Liang, Benjamin Han, Mohamed Yakout, and Ahmed Mohamed. 2018. Building a large-scale, accurate and fresh knowledge graph. *KDD-2018, Tutorial* 39 (2018), 1939–1374.

[19] Stephan Gouws, Yoshua Bengio, and Greg Corrado. 2015. Bilbowa: Fast bilingual distributed representations without word alignments. In *ICML*. PMLR, 748–756.

[20] Edouard Grave, Armand Joulin, and Quentin Berthet. 2019. Unsupervised alignment of embeddings with wasserstein procrustes. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 1880–1890.

[21] Yanchao Hao, Yuanzhe Zhang, Shizhu He, Kang Liu, and Jun Zhao. 2016. A joint embedding method for entity alignment of knowledge bases. In *China Conference on Knowledge Graph and Semantic Computing*. Springer, 3–14.

[22] Geert Heyman, Ivan Vulić, and Marie Francine Moens. 2017. Bilingual lexicon induction by learning to combine word-level and character-level representations. In *EACL*. 1085–1095.

[23] Yedid Hoshen and Lior Wolf. 2018. Non-Adversarial Unsupervised Word Translation. In *EMNLP*.

[24] Jin Huang, Zhaochun Ren, Wayne Xin Zhao, Gaole He, Ji-Rong Wen, and Daxiang Dong. 2019. Taxonomy-aware multi-hop reasoning networks for sequential recommendation. In *WSDM*. 573–581.

[25] Armand Joulin, Edouard Grave, and Piotr Bojanowski Tomas Mikolov. 2017. Bag of Tricks for Efficient Text Classification. *EACL* (2017), 427.

[26] Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *EMNLP*. 1700–1709.

[27] Thomas N Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *ICLR*.

[28] Chengjiang Li, Yixin Cao, Lei Hou, Jiaxin Shi, Juanzi Li, and Tat-Seng Chua. 2019. Semi-supervised entity alignment via joint knowledge embedding model and cross-graph model. In *EMNLP-IJCNLP*. 2723–2732.

[29] Shan Li, Baoxu Shi, Jaewon Yang, Ji Yan, Shuai Wang, Fei Chen, and Qi He. 2020. Deep job understanding at linkedin. In *SIGIR*. 2145–2148.

[30] Robert Litschko, Goran Glavaš, Simone Paolo Ponzetto, and Ivan Vulić. 2018. Unsupervised cross-lingual information retrieval using monolingual data only. In *SIGIR*. 1253–1256.

[31] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective Approaches to Attention-based Neural Machine Translation. In *EMNLP*. 1412–1421.

[32] Yuning Mao, Tong Zhao, Andrey Kan, Chenwei Zhang, Xin Luna Dong, Christos Faloutsos, and Jiawei Han. 2020. Octet: Online Catalog Taxonomy Enrichment with Self-Supervision. In *KDD*. 2247–2257.

[33] Yu Meng, Yunyi Zhang, Jiaxin Huang, Yu Zhang, Chao Zhang, and Jiawei Han. 2020. Hierarchical topic mining via joint spherical tree and text embedding. In *KDD*. 1908–1917.

[34] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*. 3111–3119.

[35] Shichao Pei, Lu Yu, Guoxian Yu, and Xiangliang Zhang. 2020. Rea: Robust cross-lingual entity alignment between knowledge graphs. In *KDD*. 2175–2184.

[36] Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *European semantic web conference*. Springer, 593–607.

[37] Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-Attention with Relative Position Representations. In *NAACL-HLT*. 464–468.

[38] Samuel L Smith, David HP Turban, Steven Hamblin, and Nils Y Hammerla. 2017. Offline bilingual word vectors, orthogonal transformations and the inverted softmax. In *ICLR*.

[39] Zequn Sun, Wei Hu, and Chengkai Li. 2017. Cross-lingual entity alignment via joint attribute-preserving embedding. In *International Semantic Web Conference*. Springer, 628–644.

[40] Zequn Sun, Wei Hu, Qingheng Zhang, and Yuzhong Qu. 2018. Bootstrapping Entity Alignment with Knowledge Graph Embedding.. In *IJCAI*, Vol. 18. 4396–4402.

[41] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to Sequence Learning with Neural Networks. *Advances in Neural Information Processing Systems* 27 (2014), 3104–3112.

[42] Gongbo Tang, Mathias Müller, Annette Rios, and Rico Sennrich. 2018. Why Self-Attention? A Targeted Evaluation of Neural Machine Translation Architectures. In *EMNLP*.

[43] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *NIPS*.

[44] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2018. Graph attention networks. In *ICLR*.

[45] Ivan Vulić and Marie-Francine Moens. 2015. Monolingual and cross-lingual information retrieval models based on (bilingual) word embeddings. In *SIGIR*. 363–372.

[46] Zhichun Wang, Qingsong Lv, Xiaohan Lan, and Yu Zhang. 2018. Cross-lingual knowledge graph alignment via graph convolutional networks. In *EMNLP*. 349–357.

[47] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144* (2016).

[48] Chao Xing, Dong Wang, Chao Liu, and Yiye Lin. 2015. Normalized word embedding and orthogonal transform for bilingual word translation. In *NAACL-HLT*. 1006–1011.

[49] Kun Xu, Liwei Wang, Mo Yu, Yansong Feng, Yan Song, Zhiguo Wang, and Dong Yu. 2019. Cross-lingual Knowledge Graph Alignment via Graph Matching Neural Network. In *ACL*. 3156–3161.

[50] Xiao Yan, Ada Ma, Jaewon Yang, Lin Zhu, How Jing, Jacob Bollinger, and Qi He. 2021. Contextual Skill Proficiency via Multi-task Learning at LinkedIn. In *CIKM*. 4273–4282.

[51] Xiao Yan, Jaewon Yang, Mikhail Obukhov, Lin Zhu, Joey Bai, Shiqi Wu, and Qi He. 2019. Social skill validation at LinkedIn. In *KDD*. 2943–2951.

[52] Hsiu-Wei Yang, Yanyan Zou, Peng Shi, Wei Lu, Jimmy Lin, and SUN Xu. 2019. Aligning Cross-Lingual Entities with Multi-Aspect Information. In *EMNLP-IJCNLP*. 4422–4432.

[53] Yuchen Zhang, Amr Ahmed, Vanja Josifovski, and Alexander Smola. 2014. Taxonomy discovery for personalized recommendation. In *WSDM*. 243–252.

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