Hierarchical Topic Modelling for Knowledge Graphs

Yujia Zhang[1](✉), Marcin Pietrasik[1], Wenjie Xu[1],
and Marek Reformat[1,2]

[1] University of Alberta, 9211-116 Street, Edmonton, Canada
{yujia10,pietrasi,wx4,reformat}@ualberta.ca
[2] University of Social Sciences, 90-113 Łódź, Poland

Abstract. Recent years have demonstrated the rise of knowledge graphs as a powerful medium for storing data, showing their utility in academia and industry alike. This in turn has motivated substantial effort into modelling knowledge graphs in ways that reveal latent structures contained within them. In this paper, we propose a non-parametric hierarchical generative model for knowledge graphs that draws inspiration from probabilistic methods used in topic modelling. Our model discovers the latent probability distributions of a knowledge graph and organizes its elements in a tree of abstract topics. In doing so, it provides a hierarchical clustering of knowledge graph subjects as well as membership distributions of predicates and entities to topics. The main draw of such an approach is that it does not require any a priori assumptions about the structure of the tree other than its depth. In addition to presenting the generative model, we introduce an efficient Gibbs sampling scheme which leverages the Multinomial-Dirichlet conjugacy to integrate out latent variables, making the posterior inference process adaptable to large datasets. We quantitatively evaluate our model on three common datasets and show that it is comparable to existing hierarchical clustering techniques. Furthermore, we present a qualitative assessment of the induced hierarchy and topics.

Keywords: Knowledge graphs · Hierarchical clustering · Non-parametric model · Generative model

1 Introduction

Knowledge bases have received considerable research attention in recent years, demonstrating their utility in areas ranging from question answering [8,11] to knowledge generation [9,12,29] to recommender systems [4]. These knowledge bases are underpinned by graph structures called knowledge graphs which describe facts as a collection of triples that relate two entities via a predicate. Advances in artificial intelligence have spurred on the need to find representations of knowledge graphs which can be easily and accurately reasoned with

P. Groth et al. (Eds.): ESWC 2022, LNCS 13261, pp. 270–286, 2022.
https://doi.org/10.1007/978-3-031-06981-9_16

by machines. One aspect of this is the increased research attention devoted to generative models for knowledge graphs which learn the latent probability distributions of a graph. These models work by decomposing the knowledge graph to a set of probability distributions that, when sampled together, generate its relations. The learning process, therefore, amounts to inferring the posterior distribution conditioned on the data.

Probabilistic topic models are types of generative models that have received considerable attention in the field of natural language processing. The aim of these models is to build abstract word topics from a corpus of documents and their words. In this sense, topics may be viewed as clusters of words. Most topic models operate under the intuition that words which co-occur in the same documents are likely to have similar semantics and therefore belong to the same topics. Hierarchical topic models extend this principle and organize the induced topics into a topic hierarchy whereby each ancestor topic represents a conceptually coarser version of its descendant topics.

In this paper, we present a model for generating a topic hierarchy from knowledge graphs which extends on existing topic models. In our model, topics are collections of entities and predicates, and are organized hierarchically in the form of a rooted tree. In generating these topics, our model also implicitly hierarchically clusters subjects by sampling a corresponding tree path. Furthermore, we employ a non-parametric prior over the tree, allowing our model to be free of any a priori assumptions about its structure other than its depth. We present an efficient Gibbs sampling scheme for posterior inference of our model. The approach leverages the Multinomial-Dirichlet conjugacy to integrate out parameters for faster inference. Our evaluation demonstrates our model's ability to induce a coherent topic hierarchy as well as hierarchical subject clustering.

2 Related Works

We divide the discussion of related works into two subsections, each of which our model shares a degree of similarity with: tag hierarchy induction models; and embeddings and clustering algorithms.

2.1 Tag Hierarchy Induction Methods

In the subsequent section, we introduce the concept of knowledge graph tags and how they can be leveraged to construct a topic hierarchy. Such a formulation is similar to that used in tag hierarchy induction methods which construct a hierarchy of tags based on the documents they annotate. One such method, described by Heymann and Garcia-Molina [16], uses the cosine distance to calculate tag similarity and generality. Tags are then added greedily, starting with the most general tag, as the child of the tag already in the hierarchy they are most similar to. Schmitz [30] proposed a method which uses subsumption rules to identify the relations between parents and children in the hierarchy. These rules form a directed graph which is then pruned to create a tree. Recently, *SMICT* [25] applied principles from the aforementioned methods to knowledge graphs to induce a class taxonomy. This approach was extended in [26] to generate cluster hierarchies of knowledge graph subjects, yielding a result similar to our model. Frequency-based methods, like the ones mentioned above, often suffer from a problem where tags that appear more frequently are assumed to be more general. In an attempt to solve this, [2] introduce domain knowledge to the algorithm in [5] and verify the directionality of relations by searching for lexico-syntactic patterns on Wikipedia. This approach improves the quality of the induced hierarchy when compared to the original model. [15] and [34] both use a two phase approach in which a tag hierarchy is first induced using a strictly frequency-based approach and then optimized using domain knowledge in the form of an existing hierarchy.

2.2 Embeddings and Clustering Algorithms

Knowledge graph embedding methods map knowledge graphs from the discrete graph space to a continuous vector space. Such a representation is useful as it allows knowledge graphs to be easily integrated with common machine learning and deep learning methods. In the context of our work, knowledge graph embeddings may be used in conjunction with hierarchical clustering methods, allowing for a benchmark comparison. Perhaps the most canonical of embedding methods, TransE [9], applies the intuition that subject embeddings should be near object embeddings when translated by valid corresponding predicates. Such a formulation provides an objective function which is then optimized via stochastic gradient descent to learn the embeddings. In a related approach, RDF2Vec [28] uses breadth-first graph walks on the skip-gram language model [22] to generate embeddings. Factorization models such as RESCAL [24] and DistMult [35] learn embeddings by factorizing the knowledge graph adjacency tensor into the product of entity embeddings and relation specific translation matrices. ComplEx embeddings [33] extend DistMult to the complex domain to better handle asymmetry in the knowledge graph. ConvE [12] leverages the convolution operator in a neural framework by stacking embeddings as a martix and convolving them in two dimensions.

Having mapped a knowledge graph to a continuous space via embeddings, clustering is trivial since distances between embeddings may be easily calculated. The process is merely choosing the clustering algorithm best suited for the data. K-means [20] is perhaps the most common clustering algorithm used today and works by assigning entities to the cluster with the smallest centre distance before recalculating cluster centre based on the updated memberships. Another common approach, OPTICS [3], uses a density based approach which expands clusters so long as density criteria are being met. Spectral clustering encompasses a wide range of algorithms which operate on the eigenvalues of the input entities. To generate hierarchical clusters, agglomerate clustering builds a hierarchy bottom-up by joining clusters at higher levels in the hierarchy based on linkage criteria. We use the these clustering methods in conjunction with the aforementioned knowledge graph embeddings during our evaluation procedure.

This is similar to ExCut [13] which first generates embeddings before iteratively refining them using rule mining approaches to generate entity clusters.

3 Proposed Model

In this section, we describe our model by positioning it in the context of existing probabilistic topic models from which it draws inspiration. Specifically, we first introduce readers to Latent Dirichlet Allocation (LDA) [7] and Hierarchical Latent Dirichlet Allocation (hLDA) [6] before formalizing our model.

3.1 Problem Formulation

We define a knowledge graph as a collection of triples, $\mathcal{K}$, such that each triple relates a subject entity, s, to an object entity, o, via a predicate, p. Formally, $\mathcal{K} = \{\langle s, p, o\rangle \in \mathbf{S} \times \mathbf{P} \times \mathbf{O}\}$ where $\langle s, p, o\rangle$ is a triple, and $\mathbf{S}$, $\mathbf{P}$, and $\mathbf{O}$ are the sets of subjects, predicates, and objects in $\mathcal{K}$, respectively. We note that knowledge graphs are rarely bipartite in terms of $\mathbf{S}$ and $\mathbf{O}$. In other words, entities can take on the role of both subjects and objects in $\mathcal{K}$, thus $\mathbf{S} \cap \mathbf{O} \neq \emptyset$. Our goal is to find a representation of the knowledge graph in which entities and predicates are hierarchically organized such that entities representing coarse concepts subsume their fine grained counterparts. For instance, the concept Person is a coarser concept than Artist since it encompasses all persons, including artists and non-artists. A natural representation of this paradigm is a directed tree wherein coarse concepts occupy nodes closer to the root node. Nodes are then collections of entities and predicates which share similar semantics. Paths in the tree capture the progressive granularization of a concept.

3.2 Probabilistic Topic Models

Given a collection of documents and their words, $\mathbf{D}$, topic models generate abstract topics on the intuition that words belonging to the same topic are likely to occur in the same documents. Latent Dirichlet Allocation (LDA) [7] is a canonical example of the topic models used today. In this approach, each document, $d_i \in \mathbf{D}$, is a mixture of topics and each topic is a distribution of words. To generate a document, the number of document words, W_i, the document's topic mixture, θ_i, and each topic's word distributions, β_k, are sampled. For each document word, $w_{i,j}$, first a topic indicator $z_{i,j}$ is sampled according to θ_i then the word is generated from z_j's word distribution, β_{z_j}. This generative procedure is formally defined as follows:

- for each document; $d_i \in \mathbf{D}$
  - $W_i \sim \text{Poisson}(\xi)$
  - $\theta_i \sim \text{Dirichlet}(\alpha)$
- for each topic; $k \in 1, 2, ..., K$
  - $\beta_k \sim \text{Dirichlet}(\eta)$

- for each document; $d_i \in \mathbf{D}$
  - for each word in document; $w_{i,j} \in d_i$
    - * $z_{i,j} \sim \text{Multinomial}(\theta_i)$
    - * $w_{i,j} \sim \text{Multinomial}(\beta_{z_{i,j}})$

Learning the distributions which generate the documents amounts to inferring the posterior distribution. Although this problem is intractable for exact inference, it can be approximated with algorithms such as Variational Bayes [7] or Collapsed Gibbs Sampling [14]. We refer readers to the original papers for the full inference procedure.

LDA has been extended to generate a hierarchy of topics in Hierarchical Latent Dirichlet Allocation (hLDA) [6]. The foundation of hLDA is the nested Chinese restaurant process (nCRP) which is an extension of the Chinese restaurant process (CRP) [1]. The CRP is a recursively defined stochastic process which gets its name from the analogy of seating patrons at a Chinese restaurant. In this restaurant, there are an infinite number of tables and each table can seat an infinite number of guests. When a guest enters, the probability of him being seated at a table is proportional to the number of patrons already seated at the table. Formally, when seating guest g_i at a restaurant that has M non-empty tables, the probability of seating the guest at table m is:

$$P(g_i = m | g_{i-1}, ..., g_1) = \begin{cases} \dfrac{|n_m^i|}{i-1+\gamma} & m \leq M \\ \dfrac{\gamma}{i-1+\gamma} & m = M+1 \\ 0 & M+1 < m \end{cases}$$

where $|n_m^i|$ is the number of patrons sitting at table m when guest g_i arrives and γ is a hyperparameter which controls the probability that an incoming guest will be seated at an empty table.

The nCRP is used in hLDA as an infinitely deep and infinitely branching prior over a tree structure. In this process, a tree is generated by sampling a path, c_i, at each level in the tree via the CRP. Each node in a tree, $n_k \in \mathbf{N}$, has its own CRP and being seated at a table is analogous to taking a specific branch in the path down the tree. As before, the probability of taking a path is proportional to the amount of times the path has been taken before. When arriving at a node n_k with children $\mathbf{M}_k$ on the $(l-1)^{\text{th}}$ level in the tree, the probability of selecting an existing branch, $c_i[l] \in \mathbf{M}_l$ or creating a new branch, $c_i[l] = M_k^*$, is:

$$P(c_i[l] = m | c_{i-1:\,1}, c_i[l-1:\,1]) = \begin{cases} \dfrac{|n_m^i|}{|n_k^i| + \gamma} & m \in \mathbf{M}_k \\ \dfrac{\gamma}{|n_k^i| + \gamma} & m = M_k^* \end{cases}$$

where $\mathbf{c}_i[l]$ is the node on the path of d_i at level l, $M_k^* = \min(\mathbb{Z}^+ \setminus \mathbf{M}_k)$ is the smallest positive integer not in $\mathbf{M}_k$, and $|n_k^i|$ is the number of entities that have gone through node n_k when entity i arrived, $|n_k^i| = |\{j \in \mathbb{Z}^+ : j < i \wedge \mathbf{c}_j[l] = n_k\}|$.

Putting everything together, hLDA uses the nCRP to generate a tree of topics. The tree is bounded to a maximum depth of L and each node in the tree is associated with a topic β_k. Each document d_i samples a path through L nodes in the tree, c_i, and a topic distribution over levels in the tree analogous to the topic mixture in LDA, θ_i. For each word $w_{i,j}$ in d_i, a topic $z_{i,j}$ is sampled from θ_i and a word is generated from that topic. The generative process is summarized as follows:

- for each node in the tree; $n_k \in \mathbf{N}$
  - $\beta_k \sim \text{Dirichlet}(\eta)$
- for each document; $d_i \in \mathbf{D}$
  - $c_i \sim \text{nCRP}(\gamma)$
  - $\theta_i \sim \text{GEM}(\rho, \pi)$
  - for each word in document; $w_{i,j} \in d_i$
    * $z_{i,j} \sim \text{Multinomial}(\theta_i)$
    * $w_{i,j} \sim \beta_{c_i[z_{i,j}]}$

where GEM(ρ, π) stands for the stick-breaking process [27] and functions as the prior for topic levels. As with LDA, we refer the readers to the original papers for model inference.

3.3 Model Description

We present our model as an extension of hLDA which has been adapted to knowledge graphs. As such, we adopt the previously introduced concepts and notation, and focus on highlighting the differences.

The first difference is the departure from the domain of documents and words to that of subjects, predicates, and objects. We can think of a predicate-object pair as a *tag* which describes a subject in a way that is analogous to how a word describes a document. In this view, a tag, t, is defined as $\langle p, o \rangle$ and belongs to a subject such that $t_{i,j} \in \mathbf{T}_i$ denotes that tag $t_{i,j}$ belongs to subject s_i. This formulation is leveraged in our model by assigning a tag topic distribution, β^t, for each node in the tree. Furthermore, to capture the distributions of predicates in each cluster, we mix in a predicate specific topic, β^p. Predicates share their level indicators, $z_{i,j}$, with their corresponding tags. As such, the number of predicates belonging to a subject has to equal its tag count. We define the multiset of predicates which belong to subject s_i as $p_{i,j} \in \mathbf{P}_i$ such that $|\mathbf{P}_i| = |\mathbf{T}_i|$. Thus, each node is a collection of two topics whose elements span the domain of $\mathbf{T} \cup \mathbf{P}$.

Each subject s_i samples a path, c_i, through the tree using the nCRP as well as a level distribution, θ_i. A further departure from the original hLDA model is the replacement of the stick-breaking process as the prior of the level distribution with the Dirichlet distribution. This formulation is a return to the prior used in LDA and was chosen for two reasons. The first is that the Dirichlet distribution introduces only one hyperparameter in contrast to the stick-breaking process' two. This makes our model easier to apply a priori since hyperparameter sensitivity and selection present challenges in non-parametric models. The second is that the inference scheme is simpler when using the Dirichlet prior. Finally, the theoretical benefits of the stick-breaking prior are not justified in a practical context since the infinite distribution would get bounded in our model by the tree depth, L.

As mentioned previously, level indicators, $z_{i,j}$, are shared among corresponding predicates and tags. Thus, we sample one level indicator for each tag analogously to hLDA. This indicator is used in conjunction with the subject path to determine the node whose topics will be sampled from. Unlike hLDA which only samples words, our model samples predicates and tags from the selected node's predicate and tag topic distributions, $\beta^p[c_i[z_{i,j}]]$ and $\beta^t[c_i[z_{i,j}]]$, respectively. We use the notation $\beta^p[c_i[z_{i,j}]]$ and $\beta^t[c_i[z_{i,j}]]$ to denote the predicate and tag topic distributions of the node at level $z_{i,j}$ on path c_i. The generative process is defined as follows:

- for each node in the tree; $n_k \in \mathbf{N}$
  - $\beta^p \sim \text{Dirichlet}(\eta_p)$
  - $\beta^t \sim \text{Dirichlet}(\eta_t)$
- for each subject; $s_i \in \mathbf{S}$
  - $c_i \sim \text{nCRP}(\gamma)$
  - $\theta_i \sim \text{Dirichlet}(\alpha)$
  - for each tag in subject; $t_{i,j} \in \mathbf{T}_i$
    - $z_{i,j} \sim \text{Multinomial}(\theta_i)$
  - for each predicate in subject; $p_{i,j} \in \mathbf{P}_i$
    - $p_{i,j} \sim \text{Multinomial}(\beta^p[c_i[z_{i,j}]])$
  - for each tag in subject; $t_{i,j} \in \mathbf{T}_i$
    - $t_{i,j} \sim \text{Multinomial}(\beta^t[c_i[z_{i,j}]])$

η_p and η_t are hyperparameters of our model which control the sparsity of the topics such that lower η values result in sparser topics which are more dissimilar from one another. Furthermore, the ratio between η_p and η_t controls the relative importance of predicates to tags when calculating the likelihood functions. γ is a hyperparameter of the nCRP and controls the probability of creating a new path in the tree such that higher γ values will generate trees with a higher average branching factor. Finally, α is the topic level hyperparameter. We provide a graphical representation of our model using plate notation in Fig. 1.

3.4 Inference

Our model is intractable for exact inference, thus we approximate it using collapsed Gibbs sampling for posterior inference. The goal of the sampling scheme is to generate the subject paths, $\mathbf{c}$, and level indicators, $\mathbf{z}$, by inferring the latent parameters. For faster mixing, we integrate out the topic distributions, β^p and β^t, as well as the level distributions, θ, by leveraging the Multinomial-Dirichlet conjugacy. This reduces our inference scheme to simply sampling paths and levels alternately until the parameters of the model are learned, at which point we can collect samples to estimate the true posterior.

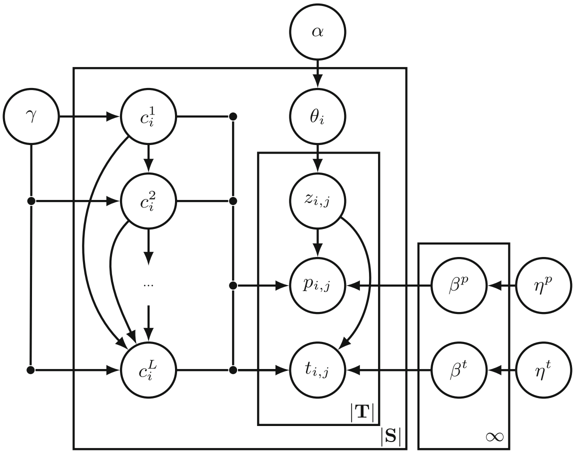

Fig. 1. Plate diagram for our model.

Sampling Paths. The posterior distribution of c_i, the path for subject s_i, conditioned on all other variables is:

$$\mathbb{P}(c_i|\mathbf{c}_{-i}, \mathbf{z}_i, \mathbf{P}_i, \mathbf{T}_i, \gamma, \eta_p, \eta_t) \propto \mathbb{P}(c_i|\mathbf{c}_{-i}, \gamma)\mathbb{P}(\mathbf{P}_i|c_i, \mathbf{P}_{-i}, \mathbf{z}_i, \eta_p) \\ \mathbb{P}(\mathbf{T}_i|c_i, \mathbf{T}_{-i}, \mathbf{z}_i, \eta_t) \tag{1}$$

where $\mathbf{c}_{-i}$ denotes all paths in the tree excluding the path taken by subject s_i. Likewise, $\mathbf{P}_{-i}$ and $\mathbf{T}_{-i}$ denote the predicates and tags on the tree leaving out those belonging to to subject s_i. This expression is merely an application of Bayes' theorem which states the posterior is proportional to the likelihood times the prior. The first term, $\mathbb{P}(c_i|\mathbf{c}_{-i}, \gamma)$, is the nCRP prior and is calculated as outlined earlier in the paper. The second term, $\mathbb{P}(\mathbf{P}_i|c_i, \mathbf{P}_{-i}, \mathbf{z}_i, \eta_p)$, is the predicate likelihood given the choice of paths. In other words, it is the probability of observing the predicate data if subject s_i were to take path c_i. The calculation of this term is defined as follows:

$$\begin{aligned}\mathbb{P}(&\mathbf{P}_i|c_i, \mathbf{P}_{-i}, \mathbf{z}_i, \eta_p) \\ &= \prod_{l=1}^{L} \frac{\Gamma\Big(\sum_{p_{i,j} \in \mathbf{P}_{-i}} \#[\mathbf{z}_{-i} = l, \mathbf{c}_{-i,l} = c_{i,l}, \mathbf{P}_{-i} = p_{i,j}] + \eta_p|\mathbf{P}|\Big)}{\sum_{p_{i,j} \in \mathbf{P}_{-i}} \Gamma\Big(\#[\mathbf{z}_{-i} = l, \mathbf{c}_{-i,l} = c_{i,l}, \mathbf{P}_{-i} = p_{i,j}] + \eta_p\Big)} \\ &\prod_{l=1}^{L} \frac{\prod_{p_{i,j} \in \mathbf{P}_i} \Gamma\Big(\#[\mathbf{z}_i = l, \mathbf{c}_{i,l} = c_{i,l}, \mathbf{P}_i = p_{i,j}] + \eta_p\Big)}{\Gamma\Big(\prod_{p_{i,j} \in \mathbf{P}_i} \#[\mathbf{z}_i = l, \mathbf{c}_{i,l} = c_{i,l}, \mathbf{P}_i = p_{i,j}] + \eta_p|\mathbf{P}|\Big)}\end{aligned} \tag{2}$$

where $\Gamma(.)$ is the gamma function and $\#[.]$ indicates the number of elements that satisfy the given conditions. Finally, the third term, $\mathbb{P}(\mathbf{T}_i|c_i, \mathbf{T}_{-i}, \mathbf{z}_i, \eta_t)$, is the tag likelihood given the choice of paths and is calculated analogously to the predicate likelihood:

$$\begin{aligned}
&\mathbb{P}(\mathbf{T}_i|c_i,\mathbf{T}_{-i},\mathbf{z}_i,\eta_t)\\
&\quad=\prod_{l=1}^{L}\frac{\mathit{\Gamma}\Big(\sum_{t_{i,j}\in\mathbf{T}_{-i}}\#[\mathbf{z}_{-i}=l,\mathbf{c}_{-i,l}=c_{i,l},\mathbf{T}_{-i}=t_{i,j}]+\eta_t|\mathbf{T}|\Big)}{\prod_{t_{i,j}\in\mathbf{T}_{-i}}\mathit{\Gamma}\Big(\#[\mathbf{z}_{-i}=l,\mathbf{c}_{-i,l}=c_{i,l},\mathbf{T}_{-i}=t_{i,j}]+\eta_t\Big)}\\
&\quad\prod_{l=1}^{L}\frac{\prod_{t_{i,j}\in\mathbf{T}_i}\mathit{\Gamma}\Big(\#[\mathbf{z}_i=l,\mathbf{c}_{i,l}=c_{i,l},\mathbf{T}_i=t_{i,j}]+\eta_t\Big)}{\mathit{\Gamma}\Big(\sum_{t_{i,j}\in\mathbf{T}_i}\#[\mathbf{z}_i=l,\mathbf{c}_{i,l}=c_{i,l},\mathbf{T}_i=t_{i,j}]+\eta_t|\mathbf{T}|\Big)} \qquad (3)
\end{aligned}$$

The time complexity of sampling a single path, c_i, is $\mathcal{O}(|\mathbf{N}|(|\mathbf{S}|+|\mathbf{T}|))$, thus sampling all the paths in one iteration of the Gibbs sampler is $\mathcal{O}(|\mathbf{S}||\mathbf{N}|(|\mathbf{S}|+|\mathbf{T}|))$.

Sampling Levels. The posterior distribution of $z_{i,j}$, the level indicator for the j^{th} tag in subject s_i is as follows:

$$\begin{aligned}
\mathbb{P}(z_{i,j}|\mathbf{z}_{i,-j},\mathbf{P}_{i,-j},{}_{i,-j},\mathbf{c},\eta_p,\eta_t,\alpha)\propto{}&\mathbb{P}(z_{i,j}|\mathbf{z}_{i,-j},\alpha)\mathbb{P}(p_{i,j}|\mathbf{P}_{i,-j},\mathbf{c},\mathbf{z}_i,\eta_p)\\
&\mathbb{P}(t_{i,j}|\mathbf{T}_{i,-j},\mathbf{c},\mathbf{z}_i,\eta_t) \qquad (4)
\end{aligned}$$

where $\mathbf{z}_{i,-j}$ are all the level indicators in subject s_i excluding $z_{i,j}$, the indicator for tag $t_{i,j}$. The prior for level indicators, $\mathbb{P}(z_{i,-j}|\mathbf{z}_{i,-j},\alpha)$, is obtained by integrating out the Multinomial distribution via the Multinomial-Dirichlet conjugacy and calculating the Dirichlet prior as follows:

$$\begin{aligned}
\mathbb{P}(z_{i,j}|\mathbf{z}_{i,-j},\alpha)&=\mathbb{E}(z_{i,j}|\mathbf{z}_{i,-j},\alpha)\\
&=\mathbb{E}\Big(\mathbb{E}(z_{i,j}=l)|\theta_1,\theta_2,...,\theta_L,\mathbf{z}_{i,-j},\alpha\Big)\\
&\propto\#[\mathbf{z}_{i,-j}=l]+\alpha \qquad (5)
\end{aligned}$$

The predicate likelihood, $\mathbb{P}(p_{i,j}|\mathbf{P}_{i,-j},c_i,\mathbf{z}_i,\eta_p)$, is calculated by counting the total number of predicates at the node specified by $z_{i,j}$ on path c_i that are the same as $p_{i,j}$:

$$\begin{aligned}
\mathbb{P}(p_{i,j}|\mathbf{P}_{i,-j},c_i,\mathbf{z}_i,\eta_p)&=\mathbb{E}(p_{i,j}|\mathbf{z}_i,c_i,\eta_p)\\
&\propto\#[\mathbf{z}_{-(i,j)}=z_{i,j},\mathbf{c}_{z_{i,j}}=c_{i,z_{i,j}},\mathbf{P}_{-(i,j)}=p_{i,j}]+\eta_p \qquad (6)
\end{aligned}$$

The tag likelihood, $\mathbb{P}(t_{i,j}|\mathbf{T}_{i,-j},\mathbf{c},\mathbf{z}_i,\eta_t)$, is calculated analogously:

$$\begin{aligned}
\mathbb{P}(t_{i,j}|\mathbf{T}_{i,-j},c_i,\mathbf{z}_i,\eta_t)&=\mathbb{E}(p_{i,j}|\mathbf{z}_i,c_i,\eta_t)\\
&\propto\#[\mathbf{z}_{-(i,j)}=z_{i,j},\mathbf{c}_{z_{i,j}}=c_{i,z_{i,j}},\mathbf{T}_{-(i,j)}=t_{i,j}]+\eta_t \qquad (7)
\end{aligned}$$

The time complexity of sampling a single topic, $z_{i,j}$, is $\mathcal{O}(L)$ and meaning that sampling all levels is $\mathcal{O}(|\mathbf{S}||\mathbf{T}||L)$.

Collapsed Gibbs Sampling. As mentioned previously, the collapsed Gibbs sampling process samples paths and levels alternately, as summarized in Algorithm 1. This approach creates a Markov chain which iteratively approaches its stationary distribution. As such, it is necessary to burn-in a fixed number of samples before samples approximating the posterior distribution may be obtained. Although Gibbs sampling is guaranteed to converge in the infinite case, the speed with which it does so is highly variable and difficult to predict a priori. Monitoring the likelihood of the model is therefore important in determining whether sufficient training has taken place. Furthermore, due to the non-parametric nature of our model, the selection of hyperparameters is critically important. Recall, for instance, that the tree's structure and size changes every time it is sampled. Thus, high γ values may induce trees with branching factors too high to feasibly perform inference on.

Algorithm 1. Gibbs Sampling Procedure

Input: Knowledge graph, $\mathcal{K}$; nCRP hyperparmeter, γ; topic hyperparameters, η^p and η^t; level hyperparameter α; Number of iterations, *iters*
Output: Hierarchical topic model for $\mathcal{K}$ defined by $\mathbf{c}$ and $\mathbf{z}$

```
Obtain $\mathbf{S}$, $\mathbf{P}$, and $\mathbf{T}$ from $\mathcal{K}$
for $iter = \{1, 2, ..., iters\}$ do
    for $i \in \{1, 2, ..., |\mathbf{S}|$ do
        Sample $c_i$ using Equation 1
        for $j \in \{1, 2, ..., |\mathbf{T}|$ do
            Sample $z_{i,j}$ using Equation 4
        end for
    end for
end for
```

4 Evaluation

We split the evaluation of our model into two parts: quantitative and qualitative. In our quantitative evaluation, we train our model to obtain a hierarchical clustering of subject entities. This clustering is then evaluated by comparing against ground truth labels and calculating metrics of clustering performance. This gives insight into the quality of induced tree and allocation of subjects to leaf nodes. To assess the quality of the inferred topic clusters, we perform a qualitative evaluation by analyzing the membership distributions of predicates and tags to selected topics. What follows is a summary of our evaluation procedure and discussion of the results. The source code for our model along with the datasets used may be found on GitHub[1].

[1] https://github.com/yujia0223/hkg.

Table 1. Summary of ground truth classes used to derive clustering evaluation datasets.

	FB15k-237	YAGO3-10	DBpedia
Level 1	Person, Organization, Location, Event	Person, Organization, Body of Water	Person, Place
Level 2	Artist, Politician, Scientist, Officeholder, Writer, Musical Organization, Party, Enterprise, Nongovernmental Organization, County, Town, City, Mountain, Movie, Entertainment, Game, Contest	Artist, Politician, Scientist, Officeholder, Writer, Musical Organization, Party, Enterprise, Nongovernmental Organization, Stream, Lake, Ocean, Bay, Sea	Artist, Athlete, PopulatedPlace, NaturalPlace
Level 3	-	-	Actor, MusicalArtist, Painter, SoccerPlayer, GridironFootballPlayer, WinterSportPlayer, Swimmer, BodyOfWater, Mountain, Settlement, Island, Country
Level 4	-	-	AmericanFootballPlayer, IceHockeyPlayer, Lake, City, Town

4.1 Datasets

We use three real-world datasets in our evaluation: FB15k-237, YAGO3-10, and DBpedia. The datasets were chosen based on their ubiquity in existing literature and to highlight the scalability of our sampling scheme on large datasets. What follows is a brief description of each dataset.

FB15k-237. The FB15k-237 dataset [32] was constructed from the FB15k dataset [9] by removing redundant and inverse triples. It contains data queried from a version of Freebase that existed around 2013. Specifically, it is comprised of 272115 triples, 14541 entities, and 237 predicates. For our hierarchical clustering analysis, we followed a similar approach to generating a ground truth subset of the data as [18]. Namely, we first mapped entities to the WordNet taxonomy [23] through the *sameAs* predicate, which relates Freebase entities to YAGO entities. We then extracted triples containing subjects with labels on second level in the taxonomy from the sets provided in Table 1. This process yielded a dataset with 5301 subjects, 103550 triples, 10018 entities, and 190 predicates.

Table 2. Method results (mean ± standard deviation) on the FB15k-237, YAGO3-10, and DBpedia datasets. Underscore denotes significance at alpha value of 0.05 compared against our model as per t-test.

	FB15k-237		**YAGO3-10**		**DBpedia**	
Method	ARI	NMI	ARI	NMI	ARI	NMI
RDF2VEC						
K-means	.308 ± .012	.567 ± .007	.070 ± .019	.199 ± .017	.223 ± .005	.416 ± .005
OPTICS	.087 ± .000	.283 ± .000	.009 ± .000	.172 ± .000	.001 ± .000	.311 ± .000
Agglom	.455 ± .000	.601 ± .000	.038 ± .000	.174 ± .000	.236 ± .000	.414 ± .000
Spectral	.539 ± .000	.678 ± .000	.071 ± .000	.218 ± .000	.218 ± .000	.410 ± .000
TransE						
K-means	.405 ± .049	.632 ± .009	.263 ± .009	.367 ± .003	.247 ± .029	.389 ± .024
OPTICS	.031 ± .000	.253 ± .000	.049 ± .000	.150 ± .000	.001 ± .000	.198 ± .000
Agglom	.491 ± .000	.599 ± .000	.226 ± .000	.337 ± .000	.198 ± .000	.383 ± .000
Spectral	**.658** ± .000	.684 ± .000	**.270** ± .000	.345 ± .000	.057 ± .000	.321 ± .000
DistMult						
K-means	.269 ± .011	.559 ± .013	.174 ± .012	.326 ± .015	.400 ± .008	.587 ± .010
OPTICS	.016 ± .000	.189 ± .000	.029 ± .000	.175 ± .000	.002 ± .000	.184 ± .000
Agglom	.379 ± .000	.621 ± .000	.202 ± .000	**.382** ± .000	.389 ± .000	.594 ± .000
Spectral	.505 ± .000	.600 ± .000	.035 ± .000	.124 ± .000	.150 ± .000	.478 ± .000
ComplEx						
K-means	.271 ± .020	.562 ± .016	.137 ± .012	.342 ± .009	.462 ± .013	.630 ± .015
OPTICS	.019 ± .000	.202 ± .000	.017 ± .000	.152 ± .000	.002 ± .000	.235 ± .000
Agglom	.385 ± .000	.630 ± .000	.181 ± .000	.299 ± .000	.442 ± .000	.628 ± .000
Spectral	.563 ± .000	.613 ± .000	.016 ± .000	.204 ± .000	.203 ± .000	.550 ± .000
ConvE						
K-means	.332 ± .031	.619 ± .013	.004 ± .003	.004 ± .001	**.474** ± .019	.612 ± .013
OPTICS	.040 ± .000	.254 ± .000	.012 ± .000	.088 ± .000	.002 ± .000	.238 ± .000
Agglom	.384 ± .000	.630 ± .000	.003 ± .000	.005 ± .000	.458 ± .000	.614 ± .000
Spectral	.556 ± .000	**.703** ± .000	.002 ± .000	.006 ± .000	.439 ± .000	**.639** ± .000
ExCut	.343 ± .011	.651 ± .002	.130 ± .007	.322 ± .011	.380 ± .016	.595 ± .005
Our Method	.656 ± .005	.669 ± .021	.044 ± .006	.218 ± .002	.406 ± .042	.582 ± .022

YAGO3-10. The YAGO3-10 dataset was derived from the YAGO3 database [21] which is a knowledge graph derived from Wikipedia and follows the hierarchical class structure of WordNet. As with FB15k-237, we mapped entities to the WordNet taxonomy before selecting the subset defined by classes in Table 1. This resulted in a dataset with 11954 subject, 84382 triples, 27572 entities, and 28 relations.

DBpedia. The DBpedia dataset was generated by querying DBpedia [19] for random entities belonging to classes on levels 4 and 5 as specified in Table 1. Specifically, 75 entities were extracted for each of these classes. Triples where these entities take on the subject role were then queried for, filtering out triples which indicate class membership. This process resulted in 908 subjects, 57191 triples, 31202 entities, and 345 predicates. The impetus for this dataset was to

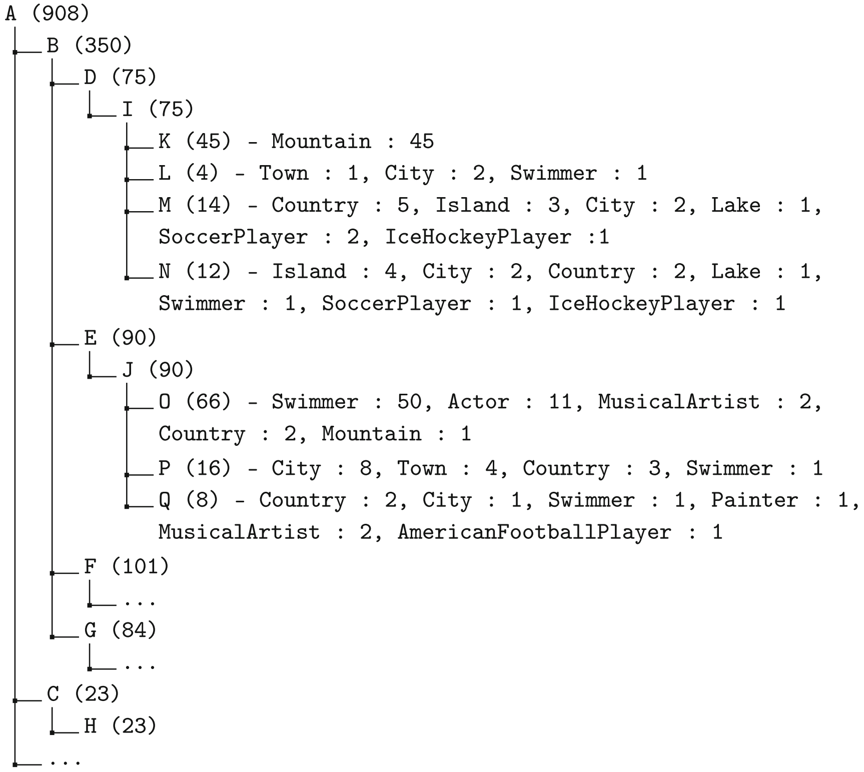
A (908)
B (350)
D (75)
I (75)
K (45) - Mountain : 45
L (4) - Town : 1, City : 2, Swimmer : 1
M (14) - Country : 5, Island : 3, City : 2, Lake : 1, SoccerPlayer : 2, IceHockeyPlayer :1
N (12) - Island : 4, City : 2, Country : 2, Lake : 1, Swimmer : 1, SoccerPlayer : 1, IceHockeyPlayer : 1
E (90)
J (90)
O (66) - Swimmer : 50, Actor : 11, MusicalArtist : 2, Country : 2, Mountain : 1
P (16) - City : 8, Town : 4, Country : 3, Swimmer : 1
Q (8) - Country : 2, City : 1, Swimmer : 1, Painter : 1, MusicalArtist : 2, AmericanFootballPlayer : 1
F (101)
...
G (84)
...
C (23)
H (23)
...

Fig. 2. Excerpt of our induced tree on the DBpedia dataset. Numbers in brackets indicate number of subjects which visited the cluster on its path.

evaluate our model on a hierarchy not rooted in the WordNet taxonomy. The hierarchical relations between DBpedia classes were obtained from the DBpedia ontology mapping which may be found on the DBpeida website[2]. All querying to generate the dataset and ground truth clusters was performed in November of 2021.

4.2 Quantitative Evaluation

To quantitatively evaluate our model, we examined the hierarchical clustering of subjects in our induced topic hierarchy. This type of evaluation jointly assesses the quality of the tree structure as well as the allocation of paths along it. Specifically, we ran our model five times on each of the aforementioned datasets using 100 burn-in samples. We then sampled from our learned distributions to obtain a topic hierarchy. We evaluated the quality of the clustering using the Adjusted Rand Index (ARI) [17] and Normalized Mutual Information (NMI)

[2] http://mappings.dbpedia.org/server/ontology/classes/.

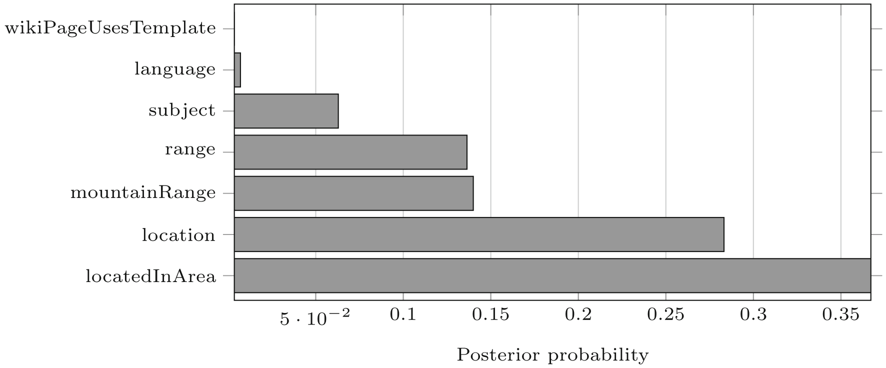

Fig. 3. Predicates and their posterior distribution for cluster K on the DBpedia tree as displayed in Fig. 2.

[31] as in previous works [18]. We compared our model against embedding based methods described in the related works section. Pretrained embeddings for these models were obtained from LibKGE[3] [10]. The mean and standard deviations of five runs are summarized in Table 2.

Our results indicate that our model is comparable with embedding based approaches. Indeed, the performance of all methods is highly variable with no method clearly outperforming the other. We note our model's underperformance on the YAGO3-10 dataset relative to other methods. We hypothesize that this is due to the high ratio of subjects to triples in this dataset. Such a characteristic results in a low amount of predicates and tags for each subject compared to other datasets. This in turn hinders our model's ability to approximate the true likelihood when calculating the posterior, resulting in lesser performance. Nevertheless, our model is still significantly better than many of the other methods as measured by a t-test. We conclude, therefore, that our model is capable of inducing coherent topic hierarchies on real world knowledge graphs.

4.3 Qualitative Evaluation

Cluster allocation is driven by the interaction of predicates and tags. Specifically, each cluster has predicate and tag membership distributions. This allows us to draw interesting observations in that we can describe a cluster by its predicate and tag distributions. This gives us insight into the composition of a cluster. Figure 2 provides an excerpt of our induced tree on the DBpedia dataset. On the other hand in Fig. 3, we provide an example of cluster K's predicate distribution from the DBpedia dataset. We note that this predicate distribution is consistent with the subjects whose path ends at this cluster. Namely, the predicates are consistent with these subjects, i.e., mountains. Furthermore, we can also analyze the distribution of objects to which the predicates are connected to. We highlight

[3] https://github.com/uma-pi1/kge.

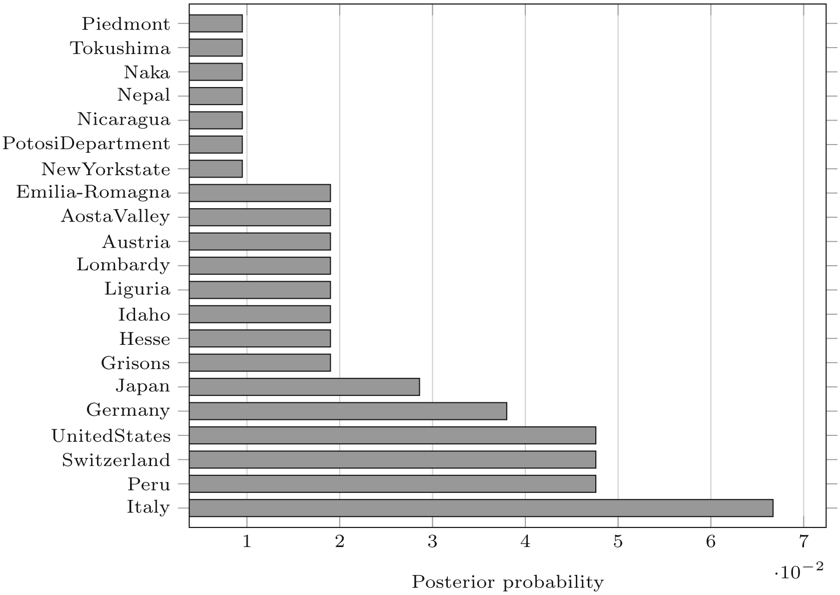

Fig. 4. Objects' posterior distribution for predicate locatedInArea

this in Fig. 4 which shows the object distribution for the predicate *locatedInArea* for cluster K. Based on the data that we used, the mountains in cluster K are most probably located in Italy, Peru, Switzerland, and United States.

5 Conclusion

In this paper we propose a model for discovering underlying hierarchical structures in knowledge graphs. For this purpose we adapt a hierarchical topic model used in natural language processing, namely hLDA, to the domain of knowledge graphs. Our model extends hLDA by introducing separate predicate and tag (predicate-object pair) topics, yielding a topic hierarchy consisting of predicate and tag distributions. Knowledge graph subjects take paths through this hierarchy which may be seen as an implicit hierarchical clustering of knowledge graph subjects. This formulation has the added benefit in that it is non-parametric, therefore does not require a priori assumptions about the tree structure other than its depth. To infer our model, we present an efficient Gibbs sampling scheme which leverages the Multinomial-Dirichlet conjugate to integrate out latent probability distributions allowing our model to scale to large datasets. We evaluate our model on three real world datasets and compare against benchmark methods. Our results demonstrate our model's ability to induce coherent topic hierarchies with high quality subject clusterings and explainable topic predicate and tag memberships.

References

1. Aldous, D.J.: Exchangeability and related topics. In: Hennequin, P.L. (ed.) École d'Été de Probabilités de Saint-Flour XIII — 1983. LNM, vol. 1117, pp. 1–198. Springer, Heidelberg (1985). https://doi.org/10.1007/BFb0099421
2. Almoqhim, F., Millard, D.E., Shadbolt, N.: Improving on popularity as a proxy for generality when building tag hierarchies from folksonomies. In: Aiello, L.M., McFarland, D. (eds.) SocInfo 2014. LNCS, vol. 8851, pp. 95–111. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-13734-6_7
3. Ankerst, M., Breunig, M.M., Kriegel, H.P., Sander, J.: Optics: ordering points to identify the clustering structure. ACM Sigmod Rec. **28**(2), 49–60 (1999)
4. Bellini, V., Schiavone, A., Di Noia, T., Ragone, A., Di Sciascio, E.: Knowledge-aware autoencoders for explainable recommender systems. In: Proceedings of the 3rd Workshop on Deep Learning for Recommender Systems (2018)
5. Benz, D., Hotho, A., Stützer, S., Stumme, G.: Semantics made by you and me: self-emerging ontologies can capture the diversity of shared knowledge (2010)
6. Blei, D.M., Griffiths, T.L., Jordan, M.I.: The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies. J. ACM (JACM) **57**(2), 7 (2010)
7. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
8. Bordes, A., Usunier, N., Chopra, S., Weston, J.: Large-scale simple question answering with memory networks. arXiv preprint (2015). arXiv:1506.02075
9. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: Advances in Neural Information Processing Systems 26 (2013)
10. Broscheit, S., Ruffinelli, D., Kochsiek, A., Betz, P., Gemulla, R.: Libkge-a knowledge graph embedding library for reproducible research. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pp. 165–174 (2020)
11. Das, R., et al.: Go for a walk and arrive at the answer: reasoning over paths in knowledge bases using reinforcement learning. arXiv preprint (2017). arXiv:1711.05851
12. Dettmers, T., Minervini, P., Stenetorp, P., Riedel, S.: Convolutional 2d knowledge graph embeddings. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)
13. Gad-Elrab, M.H., Stepanova, D., Tran, T.-K., Adel, H., Weikum, G.: ExCut: explainable embedding-based clustering over knowledge graphs. In: Pan, J.Z. (ed.) ISWC 2020. LNCS, vol. 12506, pp. 218–237. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-62419-4_13
14. Griffiths, T.L., Steyvers, M.: Finding scientific topics. Proc. Natl. Acad. Sci. **101**(suppl 1), 5228–5235 (2004)
15. Gu, C., Yin, G., Wang, T., Yang, C., Wang, H.: A supervised approach for tag hierarchy construction in open source communities. In: Proceedings of the 7th Asia-Pacific Symposium on Internetware, pp. 148–152. ACM (2015)
16. Heymann, P., Garcia-Molina, H.: Collaborative creation of communal hierarchical taxonomies in social tagging systems. Technical report (2006)
17. Hubert, L., Arabie, P.: Comparing partitions. J. Classif. **2**(1), 193–218 (1985). https://doi.org/10.1007/BF01908075

18. Jain, N., Kalo, J.-C., Balke, W.-T., Krestel, R.: Do embeddings actually capture knowledge graph semantics? In: Verborgh, R. (ed.) ESWC 2021. LNCS, vol. 12731, pp. 143–159. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-77385-4_9
19. Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P.N., Hellmann, S., Morsey, M., Van Kleef, P., Auer, S., et al.: Dbpedia-a large-scale, multilingual knowledge base extracted from wikipedia. Semant. Web **6**(2), 167–195 (2015)
20. MacQueen, J., et al.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, pp. 281–297. Oakland, CA, USA (1967)
21. Mahdisoltani, F., Biega, J., Suchanek, F.: Yago3: a knowledge base from multilingual wikipedias. In: 7th Biennial Conference on Innovative Data Systems Research. CIDR Conference (2014)
22. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. Adv. Neural Inf. Process. Syst. **26**, 3111–3119 (2013)
23. Miller, G.A.: WordNet: an electronic lexical database (1998). MIT press
24. Nickel, M., Tresp, V., Kriegel, H.P.: A three-way model for collective learning on multi-relational data (2011)
25. Pietrasik, M., Reformat, M.: A simple method for inducing class taxonomies in knowledge graphs. In: Harth, A. (ed.) ESWC 2020. LNCS, vol. 12123, pp. 53–68. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-49461-2_4
26. Pietrasik, M., Reformat, M.: Path based hierarchical clustering on knowledge graphs. arXiv preprint (2021). arXiv:2109.13178
27. Pitman, J.: Combinatorial stochastic processes. Technical report 621, Dept. Statistics, UC Berkeley, 2002. Lecture notes for St. Flour course, 2002 (2002)
28. Ristoski, P., Paulheim, H.: RDF2Vec: RDF graph embeddings for data mining. In: Groth, P. (ed.) ISWC 2016. LNCS, vol. 9981, pp. 498–514. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46523-4_30
29. Schlichtkrull, M., Kipf, T.N., Bloem, P., van den Berg, R., Titov, I., Welling, M.: Modeling relational data with graph convolutional networks. In: Gangemi, A. (ed.) ESWC 2018. LNCS, vol. 10843, pp. 593–607. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93417-4_38
30. Schmitz, P.: Inducing ontology from flickr tags. In: Collaborative Web Tagging Workshop at WWW2006, Edinburgh, Scotland, vol. 50, p. 39 (2006)
31. Shannon, C.E.: A mathematical theory of communication. Bell Syst. Techn. J. **27**(3), 379–423 (1948)
32. Toutanova, K., Chen, D.: Observed versus latent features for knowledge base and text inference. In: Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality, pp. 57–66 (2015)
33. Trouillon, T., Welbl, J., Riedel, S., Gaussier, É., Bouchard, G.: Complex embeddings for simple link prediction. In: International Conference on Machine Learning, pp. 2071–2080. PMLR (2016)
34. Wang, S., Wang, T., Mao, X., Yin, G., Yu, Y.: A hybrid approach for tag hierarchy construction. In: Capilla, R., Gallina, B., Cetina, C. (eds.) ICSR 2018. LNCS, vol. 10826, pp. 59–75. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-90421-4_4
35. Yang, B., Yih, W.T., He, X., Gao, J., Deng, L.: Embedding entities and relations for learning and inference in knowledge bases. arXiv preprint (2014). arXiv:1412.6575