Semantically Smooth Knowledge Graph Embedding

Shu Guo[†], **Quan Wang**[†*], **Bin Wang**[†], **Lihong Wang**[‡], **Li Guo**[†]
[†]Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China
{guoshu,wangquan,wangbin,guoli}@iie.ac.cn
[‡]National Computer Network Emergency Response Technical Team
Coordination Center of China, Beijing 100029, China
wlh@isc.org.cn

Abstract

This paper considers the problem of embedding Knowledge Graphs (KGs) consisting of entities and relations into low-dimensional vector spaces. Most of the existing methods perform this task based solely on observed facts. The only requirement is that the learned embeddings should be compatible within each individual fact. In this paper, aiming at further discovering the intrinsic geometric structure of the embedding space, we propose *Semantically Smooth Embedding* (SSE). The key idea of SSE is to take full advantage of additional semantic information and enforce the embedding space to be semantically smooth, i.e., entities belonging to the same semantic category will lie close to each other in the embedding space. Two manifold learning algorithms Laplacian Eigenmaps and Locally Linear Embedding are used to model the smoothness assumption. Both are formulated as geometrically based regularization terms to constrain the embedding task. We empirically evaluate SSE in two benchmark tasks of link prediction and triple classification, and achieve significant and consistent improvements over state-of-the-art methods. Furthermore, SSE is a general framework. The smoothness assumption can be imposed to a wide variety of embedding models, and it can also be constructed using other information besides entities' semantic categories.

1 Introduction

Knowledge Graphs (KGs) like WordNet (Miller, 1995), Freebase (Bollacker et al., 2008), and DBpedia (Lehmann et al., 2014) have become extremely useful resources for many NLP related applications, such as word sense disambiguation (Agirre et al., 2014), named entity recognition (Magnini et al., 2002), and information extraction (Hoffmann et al., 2011). A KG is a multi-relational directed graph composed of entities as nodes and relations as edges. Each edge is represented as a triple of fact $\langle e_i, r_k, e_j \rangle$, indicating that head entity e_i and tail entity e_j are connected by relation r_k. Although powerful in representing structured data, the underlying symbolic nature makes KGs hard to manipulate.

Recently a new research direction called knowledge graph embedding has attracted much attention (Socher et al., 2013; Bordes et al., 2013; Bordes et al., 2014; Lin et al., 2015). It attempts to embed components of a KG into continuous vector spaces, so as to simplify the manipulation while preserving the inherent structure of the original graph. Specifically, given a KG, entities and relations are first represented in a low-dimensional vector space, and for each triple, a scoring function is defined to measure its plausibility in that space. Then the representations of entities and relations (i.e. embeddings) are learned by maximizing the total plausibility of observed triples. The learned embeddings can further be used to benefit all kinds of tasks, such as KG completion (Socher et al., 2013; Bordes et al., 2013), relation extraction (Riedel et al., 2013; Weston et al., 2013), and entity resolution (Bordes et al., 2014).

To our knowledge, most of existing KG embedding methods perform the embedding task based solely on observed facts. The only requirement is that the learned embeddings should be compatible within each individual fact. In this paper we propose *Semantically Smooth Embedding* (SSE), a new approach which further imposes constraints on the geometric structure of the embedding space. The key idea of SSE is to make ful-

[*] Corresponding author: Quan Wang.

Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics
and the 7th International Joint Conference on Natural Language Processing, pages 84–94,
Beijing, China, July 26-31, 2015. ©2015 Association for Computational Linguistics

l use of additional semantic information (i.e. semantic categories of entities) and enforce the embedding space to be semantically smooth—entities belonging to the same semantic category should lie close to each other in the embedding space. This smoothness assumption is closely related to the local invariance assumption exploited in manifold learning theory, which requires nearby points to have similar embeddings or labels (Belkin and Niyogi, 2001). Thus we employ two manifold learning algorithms Laplacian Eigenmaps (Belkin and Niyogi, 2001) and Locally Linear Embedding (Roweis and Saul, 2000) to model the smoothness assumption. The former requires an entity to lie close to every other entity in the same category, while the latter represents that entity as a linear combination of its nearest neighbors (i.e. entities within the same category). Both are formulated as manifold regularization terms to constrain the KG embedding objective function. As such, SSE obtains an embedding space which is semantically smooth and at the same time compatible with observed facts.

The advantages of SSE are two-fold: 1) By imposing the smoothness assumption, SSE successfully captures the semantic correlation between entities, which exists intrinsically but is overlooked in previous work on KG embedding. 2) KGs are typically very sparse, containing a relatively small number of facts compared to the large number of entities and relations. SSE can effectively deal with data sparsity by leveraging additional semantic information. Both aspects lead to more accurate embeddings in SSE. Moreover, our approach is quite general. The smoothness assumption can actually be imposed to a wide variety of KG embedding models. Besides semantic categories, other information (e.g. entity similarities specified by users or derived from auxiliary data sources) can also be used to construct the manifold regularization terms. And besides KG embedding, similar smoothness assumptions can also be applied in other embedding tasks (e.g. word embedding and sentence embedding).

Our main contributions can be summarized as follows. First, we devise a novel KG embedding framework that naturally requires the embedding space to be semantically smooth. As far as we know, it is the first work that imposes constraints on the geometric structure of the embedding space during KG embedding. By leveraging additional semantic information, our approach can also deal with the data sparsity issue that commonly exists in typical KGs. Second, we evaluate our approach in two benchmark tasks of link prediction and triple classification, and achieve significant and consistent improvements over state-of-the-art models.

In the remainder of this paper, we first provide a brief review of existing KG embedding models in Section 2, and then detail the proposed SSE framework in Section 3. Experiments and results are reported in Section 4. Then in Section 5 we discuss related work, followed by the conclusion and future work in Section 6.

2 A Brief Review of KG Embedding

KG embedding aims to embed entities and relations into a continuous vector space and model the plausibility of each fact in that space. In general, it consists of three steps: 1) representing entities and relations, 2) specifying a scoring function, and 3) learning the latent representations. In the first step, given a KG, entities are represented as points (i.e. vectors) in a continuous vector space, and relations as operators in that space, which can be characterized by vectors (Bordes et al., 2013; Bordes et al., 2014; Wang et al., 2014b), matrices (Bordes et al., 2011; Jenatton et al., 2012), or tensors (Socher et al., 2013). In the second step, for each candidate fact $\langle e_i, r_k, e_j \rangle$, an energy function $f(e_i, r_k, e_j)$ is further defined to measure its plausibility, with the corresponding entity and relation representations as variables. Plausible triples are assumed to have low energies. Then in the third step, to obtain the entity and relation representations, a margin-based ranking loss, i.e.,

$$\mathcal{L}=\sum_{t^+\in\mathcal{O}}\sum_{t^-\in\mathcal{N}_{t^+}}\Big[\gamma+f(e_i,r_k,e_j)-f(e'_i,r_k,e'_j)\Big]_+, \quad (1)$$

is minimized. Here, $\mathcal{O}$ is the set of observed (i.e. positive) triples, and $t^+ = \langle e_i, r_k, e_j \rangle \in \mathcal{O}$; $\mathcal{N}_{t^+}$ denotes the set of negative triples constructed by replacing entities in t^+, and $t^- = \langle e'_i, r_k, e'_j \rangle \in \mathcal{N}_{t^+}$; $\gamma > 0$ is a margin separating positive and negative triples; and $[x]_+ = \max(0, x)$. The ranking loss favors lower energies for positive triples than for negative ones. Stochastic gradient descent (in mini-batch mode) is adopted to solve the minimization problem. For details please refer to (Bordes et al., 2013) and references therein.

Different embedding models differ in the first two steps: entity/relation representation and energy

Method	Entity/Relation embeddings	Energy function
TransE (Bordes et al., 2013)	$\mathbf{e}, \mathbf{r} \in \mathbb{R}^d$	$f(e_i, r_k, e_j) = \|\mathbf{e}_i + \mathbf{r}_k - \mathbf{e}_j\|_{\ell_1/\ell_2}$
SME (lin) (Bordes et al., 2014)	$\mathbf{e}, \mathbf{r} \in \mathbb{R}^d$	$f(e_i, r_k, e_j) = (\mathbf{W}_{u1}\mathbf{r}_k + \mathbf{W}_{u2}\mathbf{e}_i + \mathbf{b}_u)^T \left(\mathbf{W}_{v1}\mathbf{r}_k + \mathbf{W}_{v2}\mathbf{e}_j + \mathbf{b}_v\right)$
SME (bilin) (Bordes et al., 2014)	$\mathbf{e}, \mathbf{r} \in \mathbb{R}^d$	$f(e_i, r_k, e_j) = \left(\left(\underline{\mathbf{W}}_u \bar{\times}_3 \mathbf{r}_k\right)\mathbf{e}_i + \mathbf{b}_u\right)^T \left(\left(\underline{\mathbf{W}}_v \bar{\times}_3 \mathbf{r}_k\right)\mathbf{e}_j + \mathbf{b}_v\right)$
SE (Bordes et al., 2011)	$\mathbf{e} \in \mathbb{R}^d$, $\mathbf{R}^u, \mathbf{R}^v \in \mathbb{R}^{d\times d}$	$f(e_i, r_k, e_j) = \|\mathbf{R}_k^u\mathbf{e}_i - \mathbf{R}_k^v\mathbf{e}_j\|_{\ell_1}$

Table 1: Existing KG embedding models.

function definition. Three state-of-the-art embedding models, namely TransE (Bordes et al., 2013), SME (Bordes et al., 2014), and SE (Bordes et al., 2011), are detailed below. Please refer to (Jenatton et al., 2012; Socher et al., 2013; Wang et al., 2014b; Lin et al., 2015) for other methods.

TransE (Bordes et al., 2013) represents both entities and relations as vectors in the embedding space. For a given triple $\langle e_i, r_k, e_j \rangle$, the relation is interpreted as a translation vector $\mathbf{r}_k$ so that the embedded entities $\mathbf{e}_i$ and $\mathbf{e}_j$ can be connected by $\mathbf{r}_k$ with low error. The energy function is defined as $f(e_i, r_k, e_j) = \|\mathbf{e}_i + \mathbf{r}_k - \mathbf{e}_j\|_{\ell_1/\ell_2}$, where $\|\cdot\|_{\ell_1/\ell_2}$ denotes the ℓ_1-norm or ℓ_2-norm.

SME (Bordes et al., 2014) also represents entities and relations as vectors, but models triples in a more expressive way. Given a triple $\langle e_i, r_k, e_j \rangle$, it first employs a function $g_u(\cdot, \cdot)$ to combine $\mathbf{r}_k$ and $\mathbf{e}_i$, and $g_v(\cdot, \cdot)$ to combine $\mathbf{r}_k$ and $\mathbf{e}_j$. Then, the energy function is defined as matching $g_u(\cdot, \cdot)$ and $g_v(\cdot, \cdot)$ by their dot product, i.e., $f(e_i, r_k, e_j) = g_u(\mathbf{r}_k, \mathbf{e}_i)^T g_v(\mathbf{r}_k, \mathbf{e}_j)$. There are two versions of SME, linear and bilinear (denoted as SME (lin) and SME (bilin) respectively), obtained by defining different $g_u(\cdot, \cdot)$ and $g_v(\cdot, \cdot)$.

SE (Bordes et al., 2011) represents entities as vectors but relations as matrices. Each relation is modeled by a left matrix $\mathbf{R}_k^u$ and a right matrix $\mathbf{R}_k^v$, acting as independent projections to head and tail entities respectively. If a triple $\langle e_i, r_k, e_j \rangle$ holds, $\mathbf{R}_k^u\mathbf{e}_i$ and $\mathbf{R}_k^v\mathbf{e}_j$ should be close to each other. The energy function is $f(e_i, r_k, e_j) = \|\mathbf{R}_k^u\mathbf{e}_i - \mathbf{R}_k^v\mathbf{e}_j\|_{\ell_1}$. Table 1 summarizes the entity/relation representations and energy functions used in these models.

3 Semantically Smooth Embedding

The methods introduced above perform the embedding task based solely on observed facts. The only requirement is that the learned embeddings should be compatible within each individual fact. However, they fail to discover the intrinsic geometric structure of the embedding space. To deal with this limitation, we introduce *Semantically Smooth Embedding* (SSE) which constrains the embedding task by incorporating geometrically based regularization terms, constructed by using additional semantic categories of entities.

3.1 Problem Formulation

Suppose we are given a KG consisting of n entities and m relations. The facts observed are stored as a set of triples $O = \left\{\langle e_i, r_k, e_j \rangle\right\}$. A triple $\langle e_i, r_k, e_j \rangle$ indicates that entity e_i and entity e_j are connected by relation r_k. In addition, the entities are classified into multiple semantic categories. Each entity e is associated with a label c_e indicating the category to which it belongs. SSE aims to embed the entities and relations into a continuous vector space which is compatible with the observed facts, and at the same time semantically smooth.

To make the embedding space compatible with the observed facts, we make use of the triple set O and follow the same strategy adopted in previous methods. That is, we define an energy function on each candidate triple (e.g. the energy functions listed in Table 1), and require observed triples to have lower energies than unobserved ones (i.e. the margin-based ranking loss defined in Eq. (1)).

To make the embedding space semantically smooth, we further leverage the entity category information $\{c_e\}$, and assume that entities within the same semantic category should lie close to each other in the embedding space. This smoothness assumption is similar to the local invariance assumption exploited in manifold learning theory (i.e. nearby points are likely to have similar embeddings or labels). So we employ two manifold learning algorithms Laplacian Eigenmaps (Belkin and Niyogi, 2001) and Locally Linear Embedding (Roweis and Saul, 2000) to model such semantic smoothness, termed as LE and LLE for short respectively.

3.2 Modeling Semantic Smoothness by LE

Laplacian Eigenmaps (LE) is a manifold learning algorithm that preserves local invariance between each two data points (Belkin and Niyogi, 2001). We borrow the idea of LE and enforce semantic smoothness by assuming:

Smoothness Assumption 1 *If two entities e_i and e_j belong to the same semantic category, they will have embeddings $\mathbf{e}_i$ and $\mathbf{e}_j$ close to each other.*

To encode the semantic information, we construct an adjacency matrix $\mathbf{W}_1 \in \mathbb{R}^{n \times n}$ among the entities, with the ij-th entry defined as:

$$w_{ij}^{(1)} = \begin{cases} 1, & \text{if } c_{e_i} = c_{e_j}, \\ 0, & \text{otherwise,} \end{cases}$$

where c_{e_i}/c_{e_j} is the category label of entity e_i/e_j. Then, we use the following term to measure the smoothness of the embedding space:

$$\mathcal{R}_1 = \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \|\mathbf{e}_i - \mathbf{e}_j\|_2^2 w_{ij}^{(1)},$$

where $\mathbf{e}_i$ and $\mathbf{e}_j$ are the embeddings of entities e_i and e_j respectively. By minimizing $\mathcal{R}_1$, we expect Smoothness Assumption 1: if two entities e_i and e_j belong to the same semantic category (i.e. $w_{ij}^{(1)} = 1$), the distance between $\mathbf{e}_i$ and $\mathbf{e}_j$ (i.e. $\|\mathbf{e}_i - \mathbf{e}_j\|_2^2$) should be small.

We further incorporate $\mathcal{R}_1$ as a regularization term into the margin-based ranking loss (i.e. Eq. (1)) adopted in previous KG embedding methods, and propose our first SSE model. The new model performs the embedding task by minimizing the following objective function:

$$\mathcal{L}_1 = \frac{1}{N} \sum_{t^+ \in \mathcal{O}} \sum_{t^- \in \mathcal{N}_{t^+}} \ell(t^+, t^-) + \frac{\lambda_1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \|\mathbf{e}_i - \mathbf{e}_j\|_2^2 w_{ij}^{(1)},$$

where $\ell(t^+, t^-) = \left[\gamma + f(e_i, r_k, e_j) - f(e_i', r_k, e_j')\right]_+$ is the ranking loss on the positive-negative triple pair (t^+, t^-), and N is the total number of such triple pairs. The first term in $\mathcal{L}_1$ enforces the resultant embedding space compatible with all the observed triples, and the second term further requires that space to be semantically smooth. Hyperparameter λ_1 makes a trade-off between the two cases.

The minimization is carried out by stochastic gradient descent. Given a randomly sampled positive triple $t^+ = \langle e_i, r_k, e_j \rangle$ and the associated negative triple $t^- = \langle e_i', r_k, e_j' \rangle$,[1] the stochastic gradient w.r.t. $\mathbf{e}_s$ ($s \in \{i, j, i', j'\}$) can be calculated as:

$$\nabla_{\mathbf{e}_s} \mathcal{L}_1 = \nabla_{\mathbf{e}_s} \ell(t^+, t^-) + 2\lambda_1 \mathbf{E}(\mathbf{D} - \mathbf{W}_1) \mathbf{1}_s,$$

where $\mathbf{E} = [\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_n] \in \mathbb{R}^{d \times n}$ is a matrix consisting of entity embeddings; $\mathbf{D} \in \mathbb{R}^{n \times n}$ is a diagonal matrix with the i-th entry on the diagonal being $d_{ii} = \sum_{j=1}^{n} w_{ij}^{(1)}$; and $\mathbf{1}_s \in \mathbb{R}^n$ is a column vector where the s-th entry is 1 and the others are 0. Other parameters are not included in $\mathcal{R}_1$, and their gradients remain the same as defined in previous work.

3.3 Modeling Semantic Smoothness by LLE

As opposed to LE which preserves local invariance within data pairs, Locally Linear Embedding (LLE) expects each data point to be roughly reconstructed by a linear combination of its nearest neighbors (Roweis and Saul, 2000). We borrow the idea of LLE and enforce semantic smoothness by assuming:

Smoothness Assumption 2 *Each entity e_i can be roughly reconstructed by a linear combination of its nearest neighbors in the embedding space, i.e., $\mathbf{e}_i \approx \sum_{e_j \in \mathcal{N}(e_i)} \alpha_j \mathbf{e}_j$. Here nearest neighbors refer to entities belonging to the same semantic category with e_i.*

To model this assumption, for each entity e_i, we randomly sample K entities uniformly from the category to which e_i belongs, denoted as the nearest neighbor set $\mathcal{N}(e_i)$. We construct a weight matrix $\mathbf{W}_2 \in \mathbb{R}^{n \times n}$ by defining:

$$w_{ij}^{(2)} = \begin{cases} 1, & \text{if } e_j \in \mathcal{N}(e_i), \\ 0, & \text{otherwise,} \end{cases}$$

and normalize the rows so that $\sum_{j=1}^{n} w_{ij}^{(2)} = 1$ for each row i. Note that $\mathbf{W}_2$ is no longer a symmetric matrix. The smoothness of the embedding space can be measured by the reconstruction error:

$$\mathcal{R}_2 = \sum_{i=1}^{n} \left\| \mathbf{e}_i - \sum_{e_j \in \mathcal{N}(e_i)} w_{ij}^{(2)} \mathbf{e}_j \right\|_2^2 .$$

Minimizing $\mathcal{R}_2$ results in Smoothness Assumption 2: each entity can be linearly reconstructed from its nearest neighbors with low error.

By incorporating $\mathcal{R}_2$ as a regularization term into the margin-based ranking loss defined in Eq. (1), we obtain our second SSE model, which performs the embedding task by minimizing:

$$\mathcal{L}_2 = \frac{1}{N} \sum_{t^+ \in \mathcal{O}} \sum_{t^- \in \mathcal{N}_{t^+}} \ell(t^+, t^-) + \lambda_2 \sum_{i=1}^{n} \left\| \mathbf{e}_i - \sum_{e_j \in \mathcal{N}(e_i)} w_{ij}^{(2)} \mathbf{e}_j \right\|_2^2 .$$

[1] The negative triple is constructed by replacing one of the entities in the positive triple.

The resultant embedding space is also semantically smooth and compatible with the observed triples. Hyperparameter λ_2 makes a trade-off between the two cases.

Similar to the first model, stochastic gradient descent is used to solve the minimization problem. Given a positive triple $t^+ = \langle e_i, r_k, e_j \rangle$ and the associated negative triple $t^- = \langle e'_i, r_k, e'_j \rangle$, the gradient w.r.t. $\mathbf{e}_s$ $(s \in \{i, j, i', j'\})$ is calculated as:

$$\nabla_{\mathbf{e}_s}\mathcal{L}_2 = \nabla_{\mathbf{e}_s}\ell\left(t^+, t^-\right)+2\lambda_2\mathbf{E}\left(\mathbf{I}-\mathbf{W}_2\right)^T\left(\mathbf{I}-\mathbf{W}_2\right)\mathbf{1}_s,$$

where $\mathbf{I} \in \mathbb{R}^{n\times n}$ is the identity matrix. Other parameters are not included in $\mathcal{R}_2$, and their gradients remain the same as defined in previous work. To better capture the cohesion within each category, during each stochastic step we resample the nearest neighbors for each entity, uniformly from the category to which it belongs.

3.4 Advantages and Extensions

The advantages of our approach can be summarized as follows: 1) By incorporating geometrically based regularization terms, the SSE models are able to capture the semantic correlation between entities, which exists intrinsically but is overlooked in previous work. 2) By leveraging additional entity category information, the SSE models can deal with the data sparsity issue that commonly exists in most KGs. Both aspects lead to more accurate embeddings.

Entity category information has also been investigated in (Nickel et al., 2012; Chang et al., 2014; Wang et al., 2015), but in different manners. Nickel et al. (2012) take categories as pseudo entities and introduce a specific relation to link entities to categories. Chang et al. (2014) and Wang et al. (2015) use entity categories to specify relations' argument expectations, removing invalid triples during training and reasoning respectively. None of them considers the intrinsic geometric structure of the embedding space.

Actually, our approach is quite general. 1) The smoothness assumptions can be imposed to a wide variety of KG embedding models, not only the ones introduced in Section 2, but also those based on matrix/tensor factorization (Nickel et al., 2011; Chang et al., 2013). 2) Besides semantic categories, other information (e.g. entity similarities specified by users or derived from auxiliary data sources) can also be used to construct the manifold regularization terms. 3) Besides KG embedding, similar smoothness assumptions can also be applied in other embedding tasks (e.g. word embedding and sentence embedding).

L	S
CityCapitalOfCountry	AthleteLedSportTeam
CityLocatedInCountry	AthletePlaysForTeam
CityLocatedInGeopoliticallocation	AthletePlaysInLeague
CityLocatedInState	AthletePlaysSport
CountryLocatedInGeopoliticallocation	CoachesInLeague
StateHasCapital	CoachesTeam
StateLocatedInCountry	TeamPlaysInLeague
StateLocatedInGeopoliticallocation	TeamPlaysSport

Table 2: Relations in L and S .

4 Experiments

We empirically evaluate the proposed SSE models in two tasks: link prediction (Bordes et al., 2013) and triple classification (Socher et al., 2013).

4.1 Data Sets

We create three data sets with different sizes using NELL (Carlson et al., 2010): L , S , and N 186. L and S are two small-scale data sets, both containing 8 relations on the topics of "location" and "sport" respectively. The corresponding relations are listed in Table 2. N 186 is a larger data set containing the most frequent 186 relations. On all the data sets, entities appearing only once are removed. We extract the entity category information from a specific relation called `Generalization`, and keep non-overlapping categories.[2] Categories containing less than 5 entities on L and S as well as categories containing less than 50 entities on N 186 are further removed. Table 3 gives some statistics of the three data sets, where # Rel./# Ent./# Trip./# Cat. denotes the number of relations/entities/observed triples/categories respectively, and # c-Ent. denotes the number of entities that have category labels. Note that our SSE models do not require every entity to have a category label. From the statistics, we can see that all the three data sets suffer from the data sparsity issue, containing a relatively small number of observed triples compared to the number of entities.

On the two small-scale data sets L and S , triples are split into training/validation/test sets, with the ratio of 3:1:1. The first set is used for modeling training, the second for hyperparameter tuning, and the third for evaluation. All experiments are repeated 5 times by drawing new

[2] If two categories overlap, the smaller one is discarded.

	# Rel.	# Ent.	# Trip.	# Cat.	# c-Ent.
L	8	380	718	5	358
S	8	1,520	3,826	4	1,506
N 186	186	14,463	41,134	35	8,590

Table 3: Statistics of data sets.

training/validation/test splits, and results averaged over the 5 rounds are reported. On N 186 experiments are conducted only once, using a training/validation/test split with 31,134/5,000/5,000 triples respectively. We will release the data upon request.

4.2 Link Prediction

This task is to complete a triple $\langle e_i, r_k, e_j \rangle$ with e_i or e_j missing, i.e., predict e_i given (r_k, e_j) or predict e_j given (e_i, r_k).

Baseline methods. We take TransE, SME (lin), SME (bilin), and SE as our baselines. We then incorporate manifold regularization terms into these methods to obtain the SSE models. A model with the LE/LLE regularization term is denoted as TransE-LE/TransE-LLE for example. We further compare our SSE models with the setting proposed by Nickel et al. (2012), which also takes into account the entity category information, but in a more direct manner. That is, given an entity e with its category label c_e, we create a new triple $\langle e, \texttt{Generalization}, c_e \rangle$ and add it into the training set. Such a method is denoted as TransE-Cat for example.

Evaluation protocol. For evaluation, we adopt the same ranking procedure proposed by Bordes et al. (2013). For each test triple $\langle e_i, r_k, e_j \rangle$, the head entity e_i is replaced by every entity e'_i in the KG, and the energy is calculated for the corrupted triple $\langle e'_i, r_k, e_j \rangle$. Ranking the energies in ascending order, we get the rank of the correct entity e_i. Similarly, we can get another rank by corrupting the tail entity e_j. Aggregated over all test triples, we report three metrics: 1) the averaged rank, denoted as Mean (the smaller, the better); 2) the median of the ranks, denoted as Median (the smaller, the better); and 3) the proportion of ranks no larger than 10, denoted as Hits@10 (the higher, the better).

Implementation details. We implement the methods based on the code provided by Bordes et al. (2013)[3]. For all the methods, we create 100 mini-batches on each data set. On L and S , the dimension of the embedding space d is set in the range of $\{10, 20, 50, 100\}$, the margin γ is set in the range of $\{1, 2, 5, 10\}$, and the learning rate is fixed to 0.1. On N 186, the hyperparameters d and γ are fixed to 50 and 1 respectively, and the learning rate is fixed to 10. In LE and LLE, the regularization hyperparameters λ_1 and λ_2 are tuned in $\{10^{-4}, 10^{-5}, 10^{-6}, 10^{-7}, 10^{-8}\}$. And the number of nearest neighbors K in LLE is tuned in $\{5, 10, 15, 20\}$. The best model is selected by early stopping on the validation sets (by monitoring Mean), with a total of at most 1000 iterations over the training sets.

Results. Table 4 reports the results on the test sets of L , S , and N 186. From the results, we can see that: 1) SSE (regularized via either LE or LLE) outperforms all the baselines on all the data sets and with all the metrics. The improvements are usually quite significant. The metric Mean drops by about 10% to 65%, Median drops by about 5% to 75%, and Hits@10 rises by about 5% to 190%. This observation demonstrates the superiority and generality of our approach. 2) Even if encoded in a direct way (e.g. TransE-Cat), the entity category information can still help the baseline methods in the link prediction task. This observation indicates that leveraging additional information is indeed useful in dealing with the data sparsity issue and hence leads to better performance. 3) Compared to the strategy which incorporates the entity category information directly, formulating such information as manifold regularization terms results in better and more stable results. The *-Cat models sometimes perform even worse than the baselines (e.g. TransE-Cat on S data), while the SSE models consistently achieve better results. This observation further demonstrates the superiority of constraining the geometric structure of the embedding space.

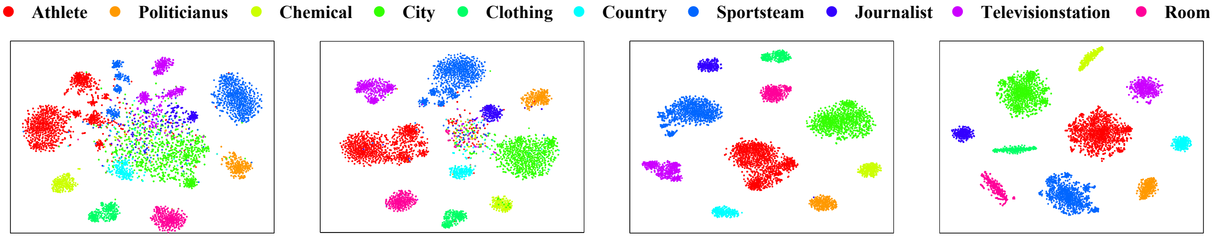
We further visualize and compare the geometric structures of the embedding spaces learned by traditional embedding and semantically smooth embedding. We select the 10 largest semantic categories in N 186 (specified in Figure 1) and the 5,740 entities therein. We take the embeddings of these entities learned by TransE, TransE-Cat, TransE-LE, and TransE-LLE, with the optimal hyperparameter settings determined in the link prediction task. Then we create 2D plots using t-SNE (Van der Maaten and Hinton, 2008)[4]. The results are shown in Figure 1, where a different

[3] https://github.com/glorotxa/SME

[4] http://lvdmaaten.github.io/tsne/

		L			S			N 186	
	Mean	Median	Hits@10 (%)	Mean	Median	Hits@10 (%)	Mean	Median	Hits@10 (%)
TransE	30.94	10.70	50.56	362.66	62.90	43.86	924.37	94.00	16.95
TransE-Cat	28.48	**8.90**	52.43	320.30	86.40	37.46	657.53	80.50	19.14
TransE-LE	28.59	**8.90**	**53.06**	**183.10**	**23.20**	**45.83**	573.55	**79.00**	**20.26**
TransE-LLE	**28.03**	9.20	52.36	231.67	52.40	43.18	**535.32**	95.00	20.02
SME (lin)	63.01	24.10	40.90	266.50	87.10	32.34	427.86	26.00	35.97
SME (lin)-Cat	41.12	18.30	42.43	263.88	70.80	35.03	309.60	**25.00**	36.22
SME (lin)-LE	**36.19**	16.10	43.75	**237.38**	**50.80**	**38.35**	276.94	**25.00**	**37.14**
SME (lin)-LLE	38.22	**15.60**	**43.96**	241.70	63.70	36.54	**252.87**	**25.00**	**37.14**
SME (bilin)	47.66	20.90	37.85	314.49	124.00	33.83	848.39	28.00	35.71
SME (bilin)-Cat	40.75	16.20	42.71	298.09	**103.80**	35.86	560.76	**24.00**	**37.83**
SME (bilin)-LE	33.41	14.00	44.24	297.90	116.10	**38.95**	**448.31**	**24.00**	37.80
SME (bilin)-LLE	**32.84**	**13.60**	**46.25**	**286.63**	110.10	35.67	452.43	28.00	36.51
SE	108.15	69.90	14.72	426.70	242.60	24.72	904.84	44.00	27.81
SE-Cat	88.36	48.20	20.76	435.44	231.00	35.39	529.38	40.00	28.68
SE-LE	**36.43**	**16.00**	**42.92**	252.30	**90.50**	37.19	456.20	43.00	30.89
SE-LLE	38.47	17.50	42.08	**235.44**	105.40	**37.83**	**447.05**	**37.00**	**31.55**

Table 4: Link prediction results on the test sets of L , S , and N 186.



(a) TransE. (b) TransE-Cat. (c) TransE-LE. (d) TransE-LLE.

Figure 1: Embeddings of entities belonging to the 10 largest categories in N 186 (best viewed in color).

color is used for each category. It is easy to see that imposing the semantic smoothness assumptions helps in capturing the semantic correlation between entities in the embedding space. Entities within the same category lie closer to each other, while entities belonging to different categories are easily distinguished (see Figure 1(c) and Figure 1(d)). Incorporating the entity category information directly could also helps. But it fails on some "hard" entities (i.e., those belonging to different categories but mixed together in the center of Figure 1(b)). We have conducted the same experiments with the other methods and observed similar phenomena.

4.3 Triple Classification

This task is to verify whether a given triple $\langle e_i, r_k, e_j \rangle$ is correct or not. We test our SSE models in this task, with the same comparison settings as used in the link prediction task.

Evaluation protocol. We follow the same evaluation protocol used in (Socher et al., 2013; Wang et al., 2014b). To create labeled data for classification, for each triple in the test and validation sets, we construct a negative triple for it by randomly corrupting the entities. To corrupt a position (head or tail), only entities that have appeared in that position are allowed. During triple classification, a triple is predicted as positive if the energy is below a relation-specific threshold δ_r; otherwise as negative. We report two metrics on the test sets: micro-averaged accuracy and macro-averaged accuracy, denoted as Micro-ACC and Macro-ACC respectively. The former is a per-triple average, while the latter is a per-relation average.

Implementation details. We use the same hyperparameter settings as in the link prediction task. The relation-specific threshold δ_r is determined by maximizing Micro-ACC on the validation sets. Again, training is limited to at most 1000 iterations, and the best model is selected by early stopping on the validation sets (by monitoring Micro-ACC).

Results. Table 5 reports the results on the test sets of L , S , and N 186. The results indicate that: 1) SSE (regularized via either LE or LLE) performs consistently better than the base-

	L		S		N 186	
	Micro-ACC	Macro-ACC	Micro-ACC	Macro-ACC	Micro-ACC	Macro-ACC
TransE	86.11	81.66	72.52	73.78	84.21	77.86
TransE-Cat	82.50	77.81	75.09	74.23	87.34	81.27
TransE-LE	86.39	81.50	79.88	77.34	**90.32**	**84.61**
TransE-LLE	**87.01**	**83.03**	**80.29**	**77.71**	90.08	84.50
SME (lin)	75.90	71.82	72.61	71.24	88.54	84.17
SME (lin)-Cat	83.33	**80.90**	73.52	72.28	91.00	86.20
SME (lin)-LE	**84.65**	79.33	79.25	74.95	92.44	88.07
SME (lin)-LLE	84.58	79.60	**79.45**	**75.61**	**92.99**	**88.68**
SME (bilin)	73.06	67.26	71.33	67.78	88.78	84.79
SME (bilin)-Cat	79.38	74.35	75.12	72.41	91.67	86.48
SME (bilin)-LE	**83.75**	79.66	79.23	**76.18**	93.37	89.29
SME (bilin)-LLE	83.54	**80.36**	**79.33**	75.35	**93.64**	**89.39**
SE	65.14	60.01	68.61	63.71	90.18	83.93
SE-Cat	68.61	62.82	67.62	62.17	92.87	87.72
SE-LE	81.67	**77.52**	**81.46**	74.72	93.94	**88.62**
SE-LLE	**82.01**	77.45	80.25	**76.07**	**93.95**	88.54

Table 5: Triple classification results (%) on the test sets of L , S , and N 186.

line methods on all the data sets in both metrics. The improvements are usually quite substantial. The metric Micro-ACC rises by about 1% to 25%, and Macro-ACC by about 2% to 30%. 2) Incorporating the entity category information directly can also improve the baselines in the triple classification task, again demonstrating the effectiveness of leveraging additional information to deal with the data sparsity issue. 3) It is a better choice to incorporate the entity category information as manifold regularization terms as opposed to encoding it directly. The *-Cat models sometimes perform even worse than the baselines (e.g. TransE-Cat on L data and SE-Cat on S data), while the SSE models consistently achieve better results. The observations are similar to those observed during the link prediction task, and further demonstrate the superiority and generality of our approach.

5 Related Work

This section reviews two lines of related work: KG embedding and manifold learning.

KG embedding aims to embed a KG composed of entities and relations into a low-dimensional vector space, and model the plausibility of each fact in that space. Yang et al. (2014) categorized the literature into three major groups: 1) methods based on neural networks, 2) methods based on matrix/tensor factorization, and 3) methods based on Bayesian clustering. The first group performs the embedding task using neural network architectures (Bordes et al., 2013; Bordes et al., 2014; Socher et al., 2013). Several state-of-the-art neural network-based embedding models have been introduced in Section 2. For other work please refer to (Jenatton et al., 2012; Wang et al., 2014b; Lin et al., 2015). In the second group, KGs are represented as tensors, and embedding is performed via tensor factorization or collective matrix factorization techniques (Singh and Gordon, 2008; Nickel et al., 2011; Chang et al., 2014). The third group embeds factorized representations of entities and relations into a nonparametric Bayesian clustering framework, so as to obtain more interpretable embeddings (Kemp et al., 2006; Sutskever et al., 2009). Our work falls into the first group, but differs in that it further imposes constraints on the geometric structure of the embedding space, which exists intrinsically but is overlooked in previous work. Although this paper focuses on incorporating geometrically based regularization terms into neural network architectures, it can be easily extended to matrix/tensor factorization techniques.

Manifold learning is a geometrically motivated framework for machine learning, enforcing the learning model to be smooth w.r.t. the geometric structure of data (Belkin et al., 2006). Within this framework, various manifold learning algorithms have been proposed, such as ISOMAP (Tenenbaum et al., 2000), Laplacian Eigenmaps (Belkin and Niyogi, 2001), and Locally Linear Embedding (Roweis and Saul, 2000). All these algorithms are based on the so-called local invariance assumption, i.e., nearby points are likely to have similar embeddings or labels. Manifold learning has been widely applied in many different areas, from dimensionality reduction (Belkin and Niyogi, 2001; Cai et al., 2008) and semi-supervised learning (Zhou et al., 2004; Zhu and Niyogi, 2005) to recommender systems (Ma et al., 2011) and community question answering (Wang et al., 2014a). This paper employs manifold learning algorithms to model the semantic smoothness assumptions in KG embedding.

6 Conclusion and Future Work

In this paper, we have proposed a novel approach to KG embedding, referred to as *Semantically Smooth Embedding* (SSE). The key idea of SSE is to impose constraints on the geometric structure of the embedding space and enforce it to be semantically smooth. The semantic smoothness assumptions are constructed by using entities' category information, and then formulated as geometrically based regularization terms to constrain the embedding task. The embeddings learned in this way are capable of capturing the semantic correlation between entities. By leveraging additional information besides observed triples, SSE can also deal with the data sparsity issue that commonly exists in most KGs. We empirically evaluate SSE in two benchmark tasks of link prediction and triple classification. Experimental results show that by incorporating the semantic smoothness assumptions, SSE significantly and consistently outperforms state-of-the-art embedding methods, demonstrating the superiority of our approach. In addition, our approach is quite general. The smoothness assumptions can actually be imposed to a wide variety of embedding models, and it can also be constructed using other information besides entities' semantic categories.

As future work, we would like to: 1) Construct the manifold regularization terms using other data sources. The only information required to construct the manifold regularization terms is the similarity between entities (used to define the adjacency matrix in LE and to select nearest neighbors for each entity in LLE). We would try entity similarities derived in different ways, e.g., specified by users or calculated from entities' textual descriptions. 2) Enhance the efficiency and scalability of SSE. Processing the manifold regularization terms can be time- and space-consuming (especially the one induced by the LE algorithm). We would investigate how to address this problem, e.g., via the efficient iterative algorithms introduced in (Saul and Roweis, 2003) or via parallel/distributed computing. 3) Impose the semantic smoothness assumptions on other KG embedding methods (e.g. those based on matrix/tensor factorization or Bayesian clustering), and even on other embedding tasks (e.g. word embedding or sentence embedding).

Acknowledgments

We would like to thank the anonymous reviewers for their valuable comments and suggestions. This work is supported by the National Natural Science Foundation of China (grant No. 61402465), the Strategic Priority Research Program of the Chinese Academy of Sciences (grant No. XDA06030200), and the National Key Technology R&D Program (grant No. 2012BAH46B03).

References

Eneko Agirre, Oier Lopez de Lacalle, and Aitor Soroa. 2014. Random walks for knowledge-based word sense disambiguation. *Computational Linguistics*, 40(1):57–84.

Mikhail Belkin and Partha Niyogi. 2001. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Advances in Neural Information Processing Systems*, pages 585–591.

Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. 2006. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research*, 7:2399–2434.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: A collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, pages 1247–1250.

Antoine Bordes, Jason Weston, Ronan Collobert, and Yoshua Bengio. 2011. Learning structured embeddings of knowledge bases. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence*, pages 301–306.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems*, pages 2787–2795.

Antoine Bordes, Xavier Glorot, Jason Weston, and Yoshua Bengio. 2014. A semantic matching energy function for learning with multi-relational data. *Machine Learning*, 94(2):233–259.

Deng Cai, Xiaofei He, Xiaoyun Wu, and Jiawei Han. 2008. Non-negative matrix factorization on manifold. In *Proceedings of the 8th IEEE International Conference on Data Mining*, pages 63–72.

Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka Jr, and Tom M. Mitchell. 2010. Toward an architecture for never-ending language learning. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence*, pages 1306–1313.

Kai-Wei Chang, Wen-tau Yih, and Christopher Meek. 2013. Multi-relational latent semantic analysis. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1602–1612.

Kai-Wei Chang, Wen-tau Yih, Bishan Yang, and Christopher Meek. 2014. Typed tensor decomposition of knowledge bases for relation extraction. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1568–1579.

Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke Zettlemoyer, and Daniel S. Weld. 2011. Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 541–550.

Rodolphe Jenatton, Nicolas L. Roux, Antoine Bordes, and Guillaume R. Obozinski. 2012. A latent factor model for highly multi-relational data. In *Advances in Neural Information Processing Systems*, pages 3167–3175.

Charles Kemp, Joshua B. Tenenbaum, Thomas L. Griffiths, Takeshi Yamada, and Naonori Ueda. 2006. Learning systems of concepts with an infinite relational model. In *Proceedings of the 21st AAAI Conference on Artificial Intelligence*, pages 381–388.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, et al. 2014. Dbpedia: A large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web Journal*.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, pages 2181–2187.

Hao Ma, Dengyong Zhou, Chao Liu, Michael R. Lyu, and Irwin King. 2011. Recommender systems with social regularization. In *Proceedings of the 4th ACM International Conference on Web Search and Data Mining*, pages 287–296.

Bernardo Magnini, Matteo Negri, Roberto Prevete, and Hristo Tanev. 2002. A wordnet-based approach to named entities recognition. In *Proceedings of the 2002 Workshop on Building and Using Semantic Networks*, pages 1–7.

George A. Miller. 1995. Wordnet: A lexical database for english. *Communications of the ACM*, 38(11):39–41.

Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2011. A three-way model for collective learning on multi-relational data. In *Proceedings of the 28th International Conference on Machine Learning*, pages 809–816.

Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2012. Factorizing yago: Scalable machine learning for linked data. In *Proceedings of the 21st International Conference on World Wide Web*, pages 271–280.

Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M. Marlin. 2013. Relation extraction with matrix factorization and universal schemas. In *Proceedings of the 2013 Conference on North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 74–84.

Sam T. Roweis and Lawrence K. Saul. 2000. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326.

Lawrence K. Saul and Sam T. Roweis. 2003. Think globally, fit locally: Unsupervised learning of low dimensional manifolds. *Journal of Machine Learning Research*, 4:119–155.

Geoffrey J. Singh and Ajit P. Gordon. 2008. Relational learning via collective matrix factorization. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 650–658.

Richard Socher, Danqi Chen, Christopher D. Manning, and Andrew Y. Ng. 2013. Reasoning with neural tensor networks for knowledge base completion. In *Advances in Neural Information Processing Systems*, pages 926–934.

Ilya Sutskever, Joshua B. Tenenbaum, and Ruslan R. Salakhutdinov. 2009. Modelling relational data using bayesian clustered tensor factorization. In *Advances in Neural Information Processing Systems*, pages 1821–1828.

Joshua B. Tenenbaum, Vin De Silva, and John C. Langford. 2000. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(85):2579–2605.

Quan Wang, Jing Liu, Bin Wang, and Li Guo. 2014a. A regularized competition model for question difficulty estimation in community question answering services. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1115–1126.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014b. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence*, pages 1112–1119.

Quan Wang, Bin Wang, and Li Guo. 2015. Knowledge base completion using embeddings and rules. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*.

Jason Weston, Antoine Bordes, Oksana Yakhnenko, and Nicolas Usunier. 2013. Connecting language and knowledge bases with embedding models for relation extraction. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1366–1371.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2014. Learning multi-relational semantics using neural-embedding models. *arXiv preprint arXiv:1411.4072*.

Dengyong Zhou, Olivier Bousquet, Thomas Navin Lal, Jason Weston, and Bernhard Schölkopf. 2004. Learning with local and global consistency. In *Advances in Neural Information Processing Systems*, pages 321–328.

Xiaojin Zhu and Partha Niyogi. 2005. Harmonic mixtures: combining mixture models and graph-based methods for inductive and scalable semi-supervised learning. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 1052–1059.