NAIST-IS-MT1651100

Master's Thesis

Data-dependent Learning of Symmetric/Asymmetric Relations for Knowledge Base Completion

Hitoshi Manabe

January 30, 2018

Department of Information Processing
Graduate School of Information Science
Nara Institute of Science and Technology

A Master's Thesis
submitted to Graduate School of Information Science,
Nara Institute of Science and Technology
in partial fulfillment of the requirements for the degree of
Master of ENGINEERING

Hitoshi Manabe

Thesis Committee:

Professor Yuji Matsumoto	(Supervisor)
Professor Satoshi Nakamura	(Co-supervisor)
Associate Professor Masashi Shimbo	(Co-supervisor)
Assistant Professor Hiroyuki Shindo	(Co-supervisor)
Assistant Professor Hiroshi Noji	(Co-supervisor)

Data-dependent Learning of Symmetric/Asymmetric Relations for Knowledge Base Completion*

Hitoshi Manabe

Abstract

Embedding-based methods for knowledge base completion (KBC) learn representations of entities and relations in a vector space, along with the scoring function to estimate the likelihood of relations between entities. The learnable class of scoring functions is designed to be expressive enough to cover a variety of real-world relations, but this expressive comes at the cost of an increased number of parameters. In particular, parameters in these methods are superfluous for relations that are either symmetric or asymmetric. To mitigate this problem, we propose a new L1 regularizer for Complex Embeddings, which is one of the state-of-the- art embedding-based methods for KBC. This regularizer promotes symmetry or skew-symmetry of the scoring function on a relation-by-relation basis, in accordance with the observed data. Our empirical evaluation shows that the proposed method outperforms the original Complex Embeddings and other baseline methods on the FB15k dataset.

Keywords:

knowledge base completion, sparse modeling, matrix factorization, graph embedding, statistical relational learning

*Master's Thesis, Department of Information Processing, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-MT1651100, January 30, 2018.

知識ベース補完における関係の対称/非対称性の学習*

真鍋 陽俊

内容梗概

知識ベース補完タスクにおいて、埋め込みベースの手法はエンティティ間にある関係が成立するかどうかを評価するスコアリング関数に従い、各エンティティと関係の表現学習を行う手法である。これらの手法では、実世界の様々な関係に対応するために表現力の高いスコアリング関数を定義する必要がある。その一方でモデルの表現力を高めようとすると学習すべきパラメータの数が増加してしまう問題がある。特に、関係が対称もしくは非対称な性質のどちらか一方を持つ場合、このような手法では冗長なパラメータが発生してしまい精度に悪影響を及ぼすと考えられる。この問題を緩和するために、本研究では知識ベース補完タスクにおいて高精度を達成している複素埋め込みに対して新たな L1 正則化を提案する。この正則化により関係毎にスコアリング関数が対称もしくは反対称になることを促し、冗長なパラメータを削減することが可能となる。評価実験において、実際に我々の手法が通常の複素埋め込みや他のベースライン手法に比べ、高精度な予測を行えることを示す。

キーワード

知識ベース補完, スパースモデリング, 行列分解, グラフ埋め込み, 統計的関係学習

*奈良先端科学技術大学院大学 情報科学研究科 情報処理学専攻 修士論文, NAIST-IS-MT1651100, 2018 年 1 月 30 日.

Acknowledgments

本修士論文を執筆するにあたり、様々な方の助けを頂きましたことをこの場をお借りして深く感謝申し上げます。

主指導教員の松本裕治教授には、研究会で様々な助言・指導をしていただきました。また本学に入学する以前にも関わらず私の相談に親身に対応くださいました。誠にありがとうございました。知能コミュニケーション学研究室の中村哲教授には、ゼミナール等で有益なアドバイスをいただきました。誠にありがとうございました。

副指導教員である新保仁准教授には本研究の元となる方向性を示していただき、論文執筆に至るまで様々なご助言を頂きました。また、研究内容に関するご助言だけでなく、研究者としてあるべき姿勢など様々なことを学ばせていただきました。誠にありがとうございました。

副指導教員である進藤裕之助教には、研究会・勉強会の際に有益なコメントを頂き、入学当初から右も左もわからない私を指導してくださりました。同じく副指導教員である能地宏助教にも、研究会・勉強会の際に貴重なご助言を頂きました。また、研究面だけでなく、日頃から公私に渡りお声を掛けていただきました。誠にありがとうございました。

大阪大学の林克彦助教には、本研究分野に取り組むきっかけを与えていただき、様々なご助言を頂きました。また、それより以前にも共同研究を通して貴重なご助言・指導をして頂き、それらの経験を通して様々な事を学ばせていただきました。誠にありがとうございました。

本研究室の同期や先輩、後輩の皆様には、公私共に学生生活をより充実したものにしていただきました。研究面でも数多くの相談・議論をさせていただき、楽しい時間を過ごすことができました。誠にありがとうございました。最後に家族や友人に深く感謝申し上げます。これまでご支援していただき誠にありがとうございました。

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Large-scale knowledge bases, such as YAGO [16], Freebase [1], and WordNet [5] are utilized in knowledge-oriented applications such as question answering and dialog systems. Facts are stored in these knowledge bases as triplets of form $(\textit{subject entity}, \textit{relation}, \textit{object entity})$.

Although a knowledge base may contain more than a million facts, many facts are still missing [12]. *Knowledge base completion* (KBC) aims to find such missing facts automatically. In recent years, vector embedding of knowledge bases has been actively pursued as a promising approach to KBC. In this approach, entities and relations are embedded into a vector space as their representations, in most cases as vectors and sometimes as matrices. A variety of methods have been proposed, each of which computes the likeliness score of given triplets using different vector/matrix operations over the representations of entities and relations involved.

In most of the previous methods, the scoring function is designed to cover general non-symmetric relations, i.e., relations r such that for some entities e_1 and e_2, triplet (e_1, r, e_2) holds but not (e_2, r, e_1). This reflects the fact that the subject and object in a relation are not interchangeable in general (e.g., *parent_of*).

However, the degree of symmetry differs from relation to relation. In particular, a non-negligible number of symmetric relations exist in knowledge bases (e.g., *sibling_of*). Moreover, many non-symmetric relations in knowledge base are actually *asymmetric*, in the sense that for *every* distinct pair e_1, e_2 of entities, if (e_1, r, e_2) holds, then (e_2, r, e_1) never holds.[1] For relations that show certain regularities such as above,

[1] Note that if a relation is asymmetric in the above sense, its truth-value matrix may not be skew-symmetric, which means even if (e_1, r, e_2) never holds, then (e_2, r, e_1) doesn't always hold. If a relation is skew-symmetric, its truth-value matrix must be skew-symmetric.

the expressiveness of models to capture general relations might be superfluous, and a model with less parameters might be preferable.

It thus seems desirable to encourage the scoring function to produce sparser models, if the observed data suggests a relation being symmetric or asymmetric. As we do not assume any background knowledge about individual relations, the choice between these contrasting properties must be made solely from the observed data. Further, we do not want the model to sacrifice the expressiveness to cope with relations that are neither purely symmetric or asymmetric.

Complex Embeddings (ComplEx) [21] are one of the state-of-the-art methods for KBC. ComplEx represents entities and relations as complex vectors, and it can model general non-symmetric relations thanks to the scoring function defined by the Hermitian inner product of these vectors. However, for symmetric relations, the imaginary parts in relation vectors are redundant parameters since they only contribute to non-symmetry of the scoring function. ComplEx is thus not exempt from the issue we mentioned above: the lack of symmetry/asymmetry consideration for individual relations. Our experimental results show that this issue indeed impairs the performance of ComplEx.

In this work, we propose a technique for training ComplEx relation vectors adaptively to the degree of symmetry/asymmetry observed in the data. Our method is based on L1 regularization, but not in the standard way; the goal here is not to make a sparse, succinct model but to adjust the degree of symmetry/asymmetry on the relation-by-relation basis, in a data-driven fashion. In our model, L1 regularization is imposed on the products of *coupled* parameters, with each parameter contributing to either the symmetry or skew-symmetry of the learned scoring function.

Experiments with synthetic data show that our method works as expected: Compared with the standard L1 regularization, the learned functions is more symmetric for symmetric relations and more skew-symmetric for asymmetric relations. Moreover, in KBC tasks on real datasets, our method outperforms the original ComplEx with standard L1 and L2 regularization, as well as other baseline methods.

Chapter 2

Background

Let $\mathbb{R}$ be the set of reals, and $\mathbb{C}$ be the set of complex numbers. $[\mathbf{v}]_j$ denotes the jth component of vector $\mathbf{v}$, and $[\mathbf{M}]_{jk}$ denotes the (j,k)-element of matrix $\mathbf{M}$. A superscript T (e.g., $\mathbf{v}^{\mathrm{T}}$) represents vector/matrix transpose. For a complex scalar, vector, or matrix $\mathbf{Z}$, $\overline{\mathbf{Z}}$ represent its complex conjugate, with $\mathrm{Re}(\mathbf{Z})$ and $\mathrm{Im}(\mathbf{Z})$ denoting its real and imaginary parts, respectively.

2.1 Knowledge Base Completion

Let $\mathscr{E}$ and $\mathscr{R}$ respectively be the sets of entities and the (names of) binary relations over entities in an incomplete knowledge base. Suppose a relational triplet (s,r,o) is not in the knowledge base for some $s,o \in \mathscr{E}$ and $r \in \mathscr{R}$. The task of KBC is to determine the truth value of such an unknown triplet; i.e., whether relation r holds between subject entity s and object entity o.

A typical approach to KBC is to learn a *scoring function* $\phi(s,r,o)$ to estimate the likeliness of an unknown triplet (s,r,o), using as training data the existing triplets in the knowledge base and their truth values. A higher score indicates that the triplet is more likely to hold.

The scoring function ϕ is usually parameterized, and the task of learning ϕ is recast as that of tuning the model parameters. To indicate this explicitly, model parameters Θ are sometimes included in the arguments of the scoring function, as in $\phi(s,r,o;\Theta)$.

2.2 Complex Embeddings (ComplEx)

The embedding-based approach to KBC defines the scoring function in terms of the vector representation (or, *embeddings*) of entities and relations. In this approach, model parameters Θ consist of these representation vectors.

ComplEx [21] is one of the latest embedding-based methods for KBC. It represents entities and relations as complex vectors. Let $\mathbf{e}_j, \mathbf{w}_r \in \mathbb{C}^d$ respectively denote the d-dimensional complex vector representations of entity $j \in \mathscr{E}$ and relation $r \in \mathscr{R}$. The scoring function of ComplEx is defined by

$$\begin{aligned}
\phi(s,r,o;\Theta) &= \mathrm{Re}\left(\mathbf{e}_s^{\mathrm{T}} \,\mathrm{diag}(\mathbf{w}_r)\overline{\mathbf{e}_o}\right) && (2.1)\\
&= \mathrm{Re}\left(\langle \mathbf{w}_r, \mathbf{e}_s, \overline{\mathbf{e}_o}\rangle\right) \\
&= \langle \mathrm{Re}(\mathbf{w}_r), \mathrm{Re}(\mathbf{e}_s), \mathrm{Re}(\mathbf{e}_o)\rangle \\
&\quad + \langle \mathrm{Re}(\mathbf{w}_r), \mathrm{Im}(\mathbf{e}_s), \mathrm{Im}(\mathbf{e}_o)\rangle \\
&\quad + \langle \mathrm{Im}(\mathbf{w}_r), \mathrm{Re}(\mathbf{e}_s), \mathrm{Im}(\mathbf{e}_o)\rangle \\
&\quad - \langle \mathrm{Im}(\mathbf{w}_r), \mathrm{Im}(\mathbf{e}_s), \mathrm{Re}(\mathbf{e}_o)\rangle, && (2.2)
\end{aligned}$$

where $\mathrm{diag}(\mathbf{v})$ denotes a diagonal matrix with the diagonal given by vector $\mathbf{v}$, and $\langle \mathbf{u}, \mathbf{v}, \mathbf{w}\rangle = \left(\sum_{k=1}^{d} [\mathbf{u}]_k [\mathbf{v}]_k [\mathbf{w}]_k\right)$, with $\Theta = \{\mathbf{e}_j \in \mathbb{C}^d \mid j \in \mathscr{E}\} \cup \{\mathbf{w}_r \in \mathbb{C}^d \mid r \in \mathscr{R}\}$. The use of complex vectors and Hermitian inner product makes ComplEx both expressive and computationally efficient.

Chapter 3

Learning Symmetric/Asymmetric Relations with L1 Regularization

3.1 Roles of Real/Imaginary Parts in Relation Vectors

Many relations in knowledge bases are either symmetric or asymmetric. For example, all 18 relations in WordNet are either symmetric (4 relations) or asymmetric (14 relations). Also, relations that take different "types" of entities as the subject and object are necessarily asymmetric; take relation *born_in* for example, which is defined for a person and a location. Clearly, if $(\mathit{Barack\_Obama}, \mathit{born\_in}, \mathit{Hawaii})$ holds, then $(\mathit{Hawaii}, \mathit{born\_in}, \mathit{Barack\_Obama})$ does not.

Now, let us look closely at the scoring function of ComplEx given by Eq. (2.1). We observe the following: If the relation vector $\mathbf{w}_r$ is a real vector, then $\phi(s,r,o) = \phi(o,r,s)$ for any $s, o \in \mathscr{E}$; i.e., the scoring function $\phi(s,r,o)$ is symmetric with respect to s and o. This can be seen by substituting $\mathrm{Im}(\mathbf{w}_r) = \mathbf{0}$ in Eq. (2.2), in which case the last two terms vanish. If, to the contrary, $\mathbf{w}_r$ is purely imaginary, $\phi(s,r,o)$ is skew-symmetric in s and o, in the sense that $\phi(s,r,o) = -\phi(o,r,s)$. Again, this can be verified with Eq. (2.2), but this time by substituting $\mathrm{Re}(\mathbf{w}_r) = \mathbf{0}$.

As we see from these two cases, the real parts in the components of $\mathbf{w}_r$ are responsible for making the scoring function ϕ symmetric, whereas the imaginary parts in $\mathbf{w}_r$ are responsible for making it skew-symmetric.

Each relation has a different degree of symmetry/asymmetry, but the original ComplEx, which is usually trained with L2 regularization, does not take this difference into account. Specifically, L2 regularization is equivalent to making a prior assumption that all parameters, including the real and imaginary parts of relation vectors, are in-

dependent from the perspective of MAP estimation. As we have discussed above, this independence assumption is unsuited for symmetric and asymmetric relations. For instance, we expect the vector for symmetric relations to have a large number of nonzero real parts and zero imaginary parts.

3.2 Multiplicative L1 Regularization for Coupled Parameters

On the basis of the observation above, we introduce a new regularization term for training ComplEx vectors. This term encourages individual relation vectors to be more symmetric or skew-symmetric in accordance with the observed data. The resulting objective function is

$$\min_{\Theta} \sum_{(s,r,o)\in\Omega} \log(1+\exp(-y_{rso}\phi(s,r,o;\Theta))) + \lambda(\alpha R_1(\Theta) + (1-\alpha)R_2(\Theta)) \tag{3.1}$$

where Ω is the training samples of triplets; $y_{rso} \in \{+1,-1\}$ gives the truth value of the triplet (s,r,o); hyperparameter $\lambda \geq 0$ determines the overall weight on the regularization terms; and $\alpha \in [0,1]$ (also a hyperparameter) controls the balance between two regularization terms R_1 and R_2. These terms are defined as follows:

$$R_1(\Theta) = \sum_{r\in\mathscr{R}} \sum_{k=1}^{d} \left|\mathrm{Re}([\mathbf{w}_r]_k) \cdot \mathrm{Im}([\mathbf{w}_r]_k)\right|, \tag{3.2}$$

$$R_2(\Theta) = \|\Theta\|_2^2. \tag{3.3}$$

In Eq. (3.3), Θ is treated as a vector, with all the parameters it contains as the vector components.

Eq. (3.1) differs from the original ComplEx objective in that it introduces the proposed regularizer R_1, which is a form of L1-norm penalty (see Equation (3.2)). In general, L1-norm penalty terms promote producing sparse solutions for the model parameters and are used for feature selection or the improve the interpretability of the model. Note, however, that our L1 penalty encourages sparsity of *pairwise* products. This means that only one of the coupled parameters needs to be propelled towards zero to minimize the summarized in Eq. (3.2). To distinguish from the standard L1 regularization, we call the regularization term in R_1 *multiplicative L1 regularizer*, since it is

based on the L1 norm of the vector of the product of the real and imaginary parts of each component.

As we explain in the next subsection, standard L1 regularization implies the independence of parameters as a prior. By contrast, our regularization term R_1 dictates the interaction between the real and imaginary parts of a component in a relation vector; if, as a result of L1 regularization, one of these parts falls to zero, the other can freely move to minimize the objective (3.1). This encourages selecting either of the coupled parameters to be zero, but not necessarily both.

Unlike the standard L1 regularization, the proposed regularization term is non-convex, and makes the optimization harder[1]. However, in our experiments reported below, multiplicative L1 regularization outperforms the standard one in KBC, and is robust against random initialization.

Since the real and imaginary parts of a relation vector govern the symmetry/skew-symmetry of the scoring function for the relation, this L1 penalty term is expected to help guide learning a vector for relation r in accordance with whether r is symmetric, asymmetric, or neither of them, as observed in the training data. For example, if the data suggests r is likely to be symmetric, our L1 regularizer should encourage the imaginary parts to be zero while allowing the real parts to take on arbitrary values. Because parameters are coupled componentwise, the proposed model can also cope with non-symmetric, non-skew-symmetric relations with different degree of symmetry/skew-symmetry.

3.3 MAP Interpretation

MAP estimation finds the best model parameters $\hat{\Theta}$ by maximizing a posterior distribution:

$$\begin{aligned}\hat{\Theta} &= \operatorname*{argmax}_{\Theta} \log p(\Theta|\mathscr{D}) \\ &= \operatorname*{argmax}_{\Theta} \log p(\mathscr{D}|\Theta) + \log p(\Theta),\end{aligned} \tag{3.4}$$

where $\mathscr{D}$ is the observed data. The first term represents the likelihood function, and the second term represents the prior distribution of parameters.

[1] Notice that the objective function in ComplEx is already non-convex without a regularization term.

Our objective function Eq. (3.1) can also be viewed as MAP estimation in the form of Eq. (3.4); the first term in our objective corresponds to the likelihood function, and the regularizer terms define the prior.

Let us discuss the prior distribution implicitly assumed by using the proposed multiplicative L1 regularizer R_1.[2] Let $C, C', C'', \ldots$ denote constants. Our multiplicative L1 regularization is equivalent to assuming the prior

$$p(\Theta) = \prod_{r \in \mathscr{R}} p(\mathbf{w}_r)$$

with

$$p(\mathbf{w}_r) = \prod_{k=1}^{d} C \exp\left(- \frac{|\mathrm{Re}([\mathbf{w}_r]_k) \cdot \mathrm{Im}([\mathbf{w}_r]_k)|}{C'} \right).$$

In other words, a 0-mean Laplacian prior is assumed on the distribution of $\mathrm{Re}([\mathbf{w}_r]_k) \cdot \mathrm{Im}([\mathbf{w}_r]_k)$. The equivalence can be seen by

$$\begin{aligned}
\log p(\Theta) &= \log \prod_{r \in \mathscr{R}} p(\mathbf{w}_r) \\
&= \log \prod_{r \in \mathscr{R}} \prod_{k=1}^{d} C \exp\left(- \frac{|\mathrm{Re}([\mathbf{w}_r]_k) \cdot \mathrm{Im}([\mathbf{w}_r]_k)|}{C'} \right) \\
&= -C'' \sum_{r \in \mathscr{R}} \sum_{k=1}^{d} |\mathrm{Re}([\mathbf{w}_r]_k) \cdot \mathrm{Im}([\mathbf{w}_r]_k)| + C'''.
\end{aligned}$$

Neglecting the scaling factor C'' and the constant term C''', we see that this is equal to the regularizer R_1 in Eq. (3.2).

Now, suppose the standard L1 regularizer

$$R_{\text{std L1}}(\Theta) = \sum_{r \in \mathscr{R}} \sum_{k=1}^{d} (|\mathrm{Re}([\mathbf{w}_r]_k)| + |\mathrm{Im}([\mathbf{w}_r]_k)|) \tag{3.5}$$

is instead of the proposed $R_1(\Theta)$. In terms of MAP estimation, in this case, its use is equivalent to assuming a prior distribution

$$p(\mathbf{w}_r) = \prod_{k=1}^{d} p(\mathrm{Re}([\mathbf{w}_r]_k))\, p(\mathrm{Im}([\mathbf{w}_r]_k)) \tag{3.6}$$

[2] For brevity, we neglect the regularizer R_2 and focus on R_1 in this discussion.

where both $p(\mathrm{Re}([\mathbf{w}_r]_k))$ and $p(\mathrm{Im}([\mathbf{w}_r]_k))$ obey a 0-mean Laplacian distribution. Notice that the distribution (3.6) assumes the independence of the real and imaginary parts of components. This assumption can be harmful if parameters are (positively or negatively) correlated with each other, which is indeed the case with symmetric or asymmetric relations.

3.4 Training Procedures

Optimizing our objective function (Eq. (3.1)) is difficult with standard online optimization methods, such as stochastic gradient descent. In this work, we extend the Regularized Dual Averaging (RDA) algorithm [22], which can produce sparse solutions effectively and is used in learning sparse word representations [4, 17]. Let us indicate the parameter value at time t by superscript (t). RDA keeps track of the online average subgradients at time t: $\widetilde{\mathbf{g}}^{(t)} = (1/t)\sum_{\tau=1}^{t} \mathbf{g}^{(\tau)}$, where $\mathbf{g}^{(t)}$ is the subgradient at time t. In this work, we calculate the subgradients $\mathbf{g}^{(t)}$ in terms of only the loss function and L2 norm penalty term; i.e., they are the derivatives of the objective without the L1 penalty term.

The update formulas for multiplicative L1 regularizer are given as follows:

$$\mathrm{Re}([\mathbf{w}_r]_k^{(t+1)}) = \begin{cases} 0, & \text{if } \left|[\widetilde{\mathbf{g}_r}]_k^{(t)}\right| \leq \beta \left|\mathrm{Im}([\mathbf{w}_r]_k^{(t)})\right|, \\ \gamma, & \text{otherwise,} \end{cases}$$

$$\mathrm{Im}([\mathbf{w}_r]_k^{(t+1)}) = \begin{cases} 0, & \text{if } \left|[\widetilde{\mathbf{g}'_r}]_k^{(t)}\right| \leq \beta \left|\mathrm{Re}([\mathbf{w}_r]_k^{(t)})\right|, \\ \gamma' & \text{otherwise,} \end{cases}$$

where $\beta = \lambda\alpha$ is a constant, $\mathbf{g}_r, \mathbf{g}'_r \in \mathbb{R}^d$ are the real and imaginary parts of the subgradients with respect to relation r, and

$$\gamma = -\eta t \left([\widetilde{\mathbf{g}_r}]_k^{(t)} - \beta \left|\mathrm{Im}([\mathbf{w}_r]_k^{(t)})\right| \mathrm{sign}([\widetilde{\mathbf{g}_r}]_k^{(t)})\right),$$

$$\gamma' = -\eta t \left([\widetilde{\mathbf{g}'_r}]_k^{(t)} - \beta \left|\mathrm{Re}([\mathbf{w}_r]_k^{(t)})\right| \mathrm{sign}([\widetilde{\mathbf{g}'_r}]_k^{(t)})\right).$$

From these formulas, we notice the interaction of the real and imaginary parts of a component; the imaginary part appears in the update formula for the real part, and vice versa. The term $\beta|\mathrm{Im}([\mathbf{w}_r^{(t)}]_k)|$ can be regarded as the strength of L1 regularizer specialized for $\mathrm{Re}([\mathbf{w}_r]_k)$ at time t; if $\mathrm{Im}([\mathbf{w}_r]_k) = 0$, then $\mathrm{Re}([\mathbf{w}_r]_k)$ keeps a nonzero

value γ. Likewise, if $\mathrm{Re}([\mathbf{w}_r]_k) = 0$, $\mathrm{Im}([\mathbf{w}_r]_k)$ is free to take on a nonzero value. Notice that the above update formulas are applied only to relation vectors. Entity vectors are learned with a standard optimization method.

Chapter 4

Related Work

4.1 Knowledge Base Embedding

RESCAL [14] is an embedding-based KBC method whose scoring function is formulated as $\mathbf{e}_s^{\mathrm{T}}\mathbf{W}_r\mathbf{e}_o$, where $\mathbf{e}_s, \mathbf{e}_o \in \mathbb{R}^d$ are the vector representations of entities s and o, respectively, and (possibly non-symmetric) matrix $\mathbf{W}_r \in \mathbb{R}^{d\times d}$ represents a relation r. Although RESCAL is able to output non-symmetric scoring functions, each relation vector $\mathbf{W}_r$ holds d^2 parameters. This can be problematic both in terms of overfitting and computational cost. To avoid this problem, several methods have been proposed recently.

DistMult [23] restricts the relation matrix to be diagonal, $\mathbf{W}_r = \mathrm{diag}(\mathbf{w}_r)$, and it can compute the likelihood score in time $O(d)$ by way of $\phi(s,r,o) = \mathbf{e}_s^{\mathrm{T}}\,\mathrm{diag}(\mathbf{w}_r)\mathbf{e}_o$. However, this form of function is necessarily symmetric in s and o; i.e., $\phi(s,r,o) = \phi(o,r,s)$. To reconcile efficiency and expressiveness, [21] proposed ComplEx, using the complex-valued representations and Hermitian inner product to define the scoring function (Eq. (2.1)). ComplEx is founded on the unitary diagonalization of normal matrices [20]. Unlike DistMult, the scoring function can be non-symmetric in s and o. [6] found that ComplEx is equivalent to another state-of-the-art KBC method, Holographic Embeddings (HolE) [13].

ANALOGY [9] also assumes that $\mathbf{W}_r$ is real normal. They showed that any real normal matrix can be block-diagonalized, where each diagonal block is either a real scalar or a 2×2 real matrix of form $\begin{pmatrix} a & b \\ -b & a \end{pmatrix}$. Notice that this 2×2 matrix is exactly the real-valued encoding of a complex value $a+ib$. In this sense, ANALOGY can be regarded as a hybrid of ComplEx (corresponding to the 2×2 diagonal blocks) and

DistMult (real scalar diagonal elements). It can also be viewed as an attempt to reduce of the number of parameters in ComplEx, by constraining some of the imaginary parts of the vector components to be zero. Although the idea resembles our proposed method, in ANALOGY, the reduced parameters (the number of scalar diagonal components) is a hyperparameter. By contrast, our approach lets the data adjust the number of parameters for individual relations, by means of the multiplicative L1 regularizer.

4.2 Sparse Modeling

Sparse modeling is often used to improve model interpretability or to enhance performance when the size of training data is insufficient relative to the number of model parameters. Lasso [18] is a sparse modeling technique for linear regression. Lasso uses L1 norm penalty to effectively find which parameters can be dispensed with. Following Lasso, several variants for inducing structural sparsity have been proposed, e.g., [19, 7]. One of them, the exclusive group Lasso [7] uses an $l_{1,2}$-norm penalty, to enforce sparsity at an intra-group level. An interesting connection exists between our regularization terms in Eq. (3.1) and the exclusive group Lasso. Let $R_j(\mathbf{w}_r)$, $j = 1,2$ represents the terms in $R_j(\Theta)$ that are concerned with relation r. When $\alpha = 2/3$, we can show that the regularizer terms $\alpha R_1(\mathbf{w}_r) + (1-\alpha)R_2(\mathbf{w}_r) = (1/3)\sum_{k=1}^{d}(\mathrm{Re}([\mathbf{w}_r]_k) + \mathrm{Im}([\mathbf{w}_r]_k))^2$. The right-hand side can be seen as an instance of the exclusive group Lasso.

Word representation learning [10, 15] has proven useful for a variety of natural language processing tasks. Sparse modeling has been applied to improve the interpretability of the learned word vectors while maintaining the expressive power of the model [4, 17]. Unlike our proposed method, however, the improvement of the model performance was not the main focus.

Chapter 5

Experiments

5.1 Demonstrations on Synthetic Data

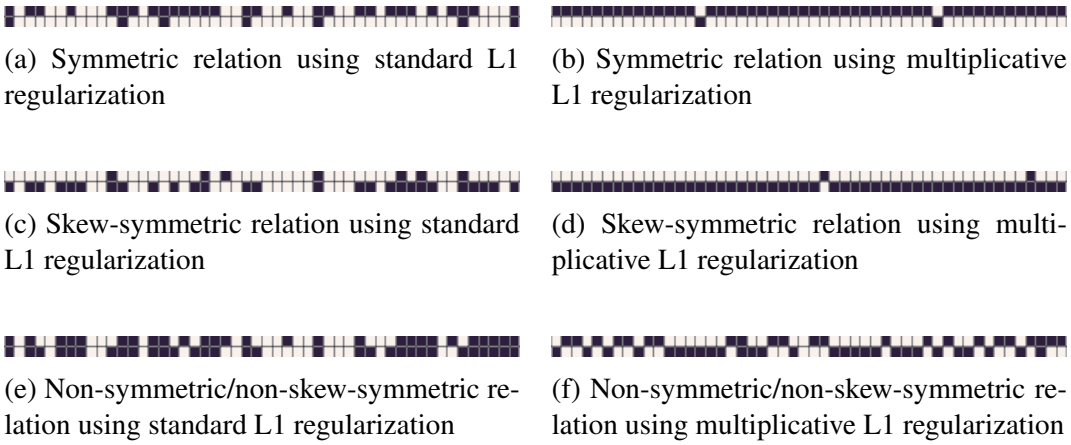
We conducted experiments with synthetic data to verify that our proposed method can learn symmetric and skew-symmetric relations, given such data.

We randomly created a knowledge base of 3 relations and 50 entities. We generated a total of 6,712 triplets, and sampled 5369 triplets as training set, then one-half of the remaining triplet as validation set and the other as test set. The truth values of triplets were determined such that the first relation was symmetric, the second was skew-symmetric, and the last was neither symmetric or skew-symmetric.

We compared the standard and multiplicative L1 regularizers on this dataset. For the standard L1 regularization, the regularizer R_{std} (Eq. (3.5)) was used in place of R_1 (Eq. (3.2)) in the objective function (3.1). The dimension of the embedding space was set to $d = 50$. For the multiplicative L1 regularizer, hyperparameters were set as follows: $\alpha = 1.0, \lambda = 0.05, \eta = 0.1$.

Figure 5.1 displays which real and imaginary parts of the learned relation vectors have non-zero values. The multiplicative L1 regularizer produced the expected results: most of the imaginary parts were zero in the symmetric relation and the real parts were zero in the skew-symmetric relation.

Table 5.1 shows the triplet classification accuracy on the test set. Triplet classification is the task of predicting the truth value of given triplets in the test set. For a given triplet, the prediction of the systems was determined by the sign of the output score ($+$ = true, $-$ = false). The multiplicative L1 regularizer ('ComplEx w/ mul L1') outperformed the standard L1 regularizer ('ComplEx w/ std L1') considerably for both symmetric and skew-symmetric relations.



(a) Symmetric relation using standard L1 regularization

(b) Symmetric relation using multiplicative L1 regularization

(c) Skew-symmetric relation using standard L1 regularization

(d) Skew-symmetric relation using multiplicative L1 regularization

(e) Non-symmetric/non-skew-symmetric relation using standard L1 regularization

(f) Non-symmetric/non-skew-symmetric relation using multiplicative L1 regularization

Figure 5.1: Visualization of the vector representations trained on synthetic data. Each column represents a complex-valued vector component, with the upper and lower cells representing the real and imaginary parts, respectively. The black cells represent non-zero values.

Table 5.1: Classification accuracy (%) on synthetic data.

Models	sym	skew	other	all
ComplEx w/ std L1	89.8	92.1	**65.7**	81.5
ComplEx w/ mul L1	**93.5**	**94.4**	65.3	**83.3**

5.2 Real Datasets: WN18 and FB15k

Following previous work, we used the WordNet (WN18) and Freebase (FB15k) datasets to verify the benefits of our proposed method. The dataset statistics are shown in Table 5.2. Because the datasets contain only positive triplets, (pseudo-)negative samples must be generated in this experiment. In this experiment, negative samples were generated by replacing the subject s and object o in a positive triplet (s,r,o) with a randomly sampled entity from $\mathscr{E}$ for training.

For evaluation, we performed the entity prediction task. In this task, an incomplete triplet is given, which is generated by hiding one of the entities, either s or o, from a positive triplet. The system must output the rankings of entities in $\mathscr{E}$ for the missing s or o in the triplet, with the goal of placing (unknown) true s or o higher in the rankings. Systems that learn a scoring function $\phi(s,r,o)$ can use the score for computing the

Table 5.2: Dataset statistics for FB15k and WN18.

Dataset	$\lvert\mathscr{E}\rvert$	$\lvert\mathscr{R}\rvert$	#train	#valid	#test
FB15k	14,951	1,345	483,142	50,000	59,071
WN18	40,943	18	141,442	5,000	5,000

rankings. The quality of the output rankings is measured by two standard evaluation measures for the KBC task: Mean Reciprocal Rank (MRR) and Hits@1, 3 and 10. We here report results in both the filtered and raw settings [2] for MRR, but only filtered values for Hits@n.

5.3 Experimental Setup

We selected the hyperparameters λ, α, and η via grid search such that they maximize the filtered MRR on the validation set. The ranges for the grid search were as follows: $\lambda \in \{0.01, 0.001, 0.0001, 0\}$, $\alpha \in \{0, 0.3, 0.5, 0.7, 1.0\}$, $\eta \in \{0.1, 0.05\}$. During the training, learning rate η was tuned with AdaGrad [3], both for entity and relation vectors. The maximum number of training epochs was set to 500 and the dimension of the vector space was $d = 200$. The number of negative triplets generated per positive training triplet was 10 for FB15k and 5 for WN18.

5.4 Results

We compared our proposed model ('ComplEx w/ mul L1') with ComplEx and ComplEx with standard L1 regularization ('ComplEx w/ std L1'). Note that our model reduces to ComplEx when $\alpha = 0$.

Table 5.3 shows the results. The results for TransE, DistMult, HolE, and ANALOGY are transcribed from the literature [21, 9]. All the compared models, except for DistMult, can produce a non-symmetric scoring function.

For most of the evaluation metrics, the multiplicative L1 regularizer ('ComplEx w/ mul L1') outperformed or was competitive to the best baseline. Of particular interest, its performance on FB15k was much better than that of WN18. Indeed, the accuracy of the original ComplEx is already quite high for WN18.

This difference between two datasets comes from the percentage of infrequent relations in two datasets. Most of the 1,345 relations in FB15k are infrequent, i.e., each of

Table 5.3: Results on the WN18 and FB15k datasets: (Filtered and raw) MRR and filtered Hits@$\{1,3,10\}$ (%). * and ** denote the results reported in [21] and [9], respectively.

	WN18					FB15k				
	MRR		Hits@			MRR		Hits@		
Models	Filter	Raw	1	3	10	Filter	Raw	1	3	10
TransE*	45.4	33.5	8.9	82.3	93.4	38.0	22.1	23.1	47.2	64.1
DistMult*	82.2	53.2	72.8	91.4	93.6	65.4	24.2	54.6	73.3	82.4
HolE*	93.8	61.6	93.0	94.5	94.9	52.4	23.2	40.2	61.3	73.9
ComplEx*	94.1	58.7	93.6	94.5	94.7	69.2	24.2	59.9	75.9	84.0
ANALOGY**	94.2	**65.7**	93.9	94.4	94.7	72.5	25.3	**64.6**	78.5	85.4
ComplEx ($\alpha = 0$)	**94.3**	58.2	**94.0**	**94.6**	94.8	69.5	24.2	59.8	76.9	85.0
ComplEx w/ std L1	**94.3**	57.9	**94.0**	94.5	94.8	71.1	25.5	61.8	78.3	85.6
ComplEx w/ mul L1	**94.3**	58.5	**94.0**	**94.6**	**94.9**	**73.3**	**25.8**	64.3	**80.3**	**86.8**

Table 5.4: Results on WN18 when the size of training data is reduced to a half

	MRR		Hits@		
Models	Filter	Raw	1	3	10
ComplEx ($\alpha = 0$)	48.3	32.9	47.4	48.7	49.8
ComplEx w/ std L1	48.2	33.4	47.2	48.7	50.2
ComplEx w/ mul L1	**49.0**	**34.6**	**47.7**	**49.7**	**51.2**

them has only several dozen triplets in training data. By contrast, of the 18 relations in WN18, most have more than one thousand training triplets.

Because sparse modeling is, in general, much effective when training data is scarce, this difference in the proportion of infrequent relations should have contributed to the different performance improvements on the two datasets. To support this hypothesis, we ran additional experiments with WN18, by reducing the number of training samples to one half. The test data remained the same as in the previous experiment. The results are shown in Table 5.4. The standard and multiplicative L1 regularizers work better than the original ComplEx and the improvement in each evaluation metric is greater than that observed with all training data. The proposed method also outperformed the

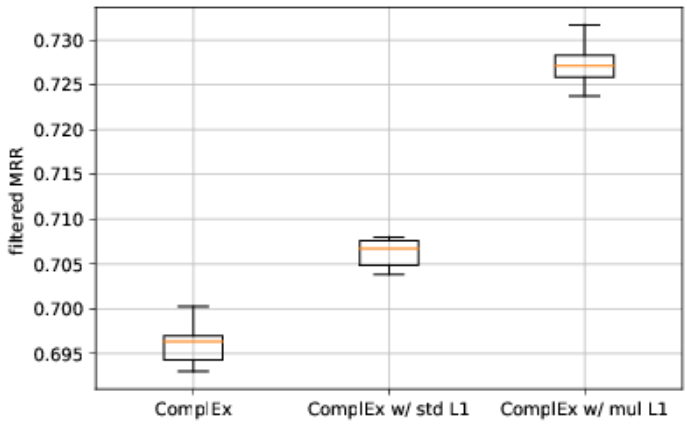


Figure 5.2: The box plots for the variance of filtered MRR on the FB15k dataset.

standard L1 regularization consistently.

The reliability of the result in Table 5.3 was verified by computing the variance of the filtered MRR scores over 8 trials on FB15k. For each of the trials, different random choices were used to generate the initial values of the representation vectors and the order of samples to process. The result, shown in Figure 5.2, confirms that random initial values have little effect on the result; there is no overlapping MRR range among the compared methods, and the proposed method ('ComplEx w/ mul L1') consistently outperformed the standard L1 ('ComplEx w/ std L1') and vanilla ComplEx with only L2 regularization.

5.5 Analysis

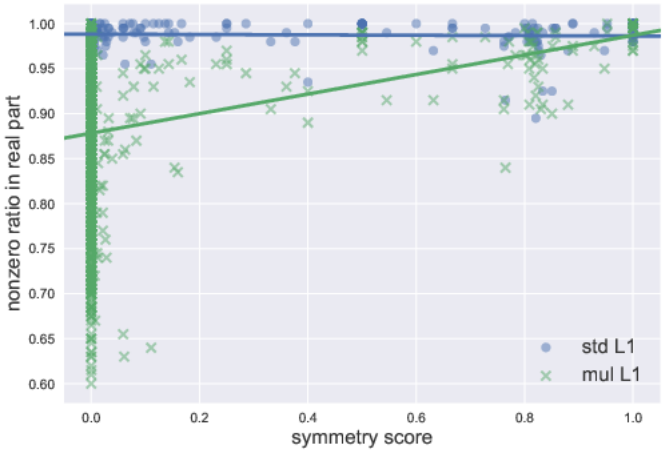
We analyzed how the two L1 regularizers, standard and multiplicative, work for symmetric and asymmetric relations on FB15k. Because of the large number of relations FB15k contains, manually extracting symmetric/asymmetric relations is difficult. To quantify the degree of symmetry of each relation r, we define its "symmetry score" by $\mathrm{sym}(r) = |\mathscr{T}_r^{\mathrm{sym}}|/|\mathscr{T}_r|$ where

$$\mathscr{T}_r = \{(s,r,o) \mid (s,r,o) \in \Omega, y_{rso} = +1\},$$
$$\mathscr{T}_r^{\mathrm{sym}} = \{(s,r,o) \mid (s,r,o) \in \Omega, y_{rso} = y_{ros} = +1\},$$

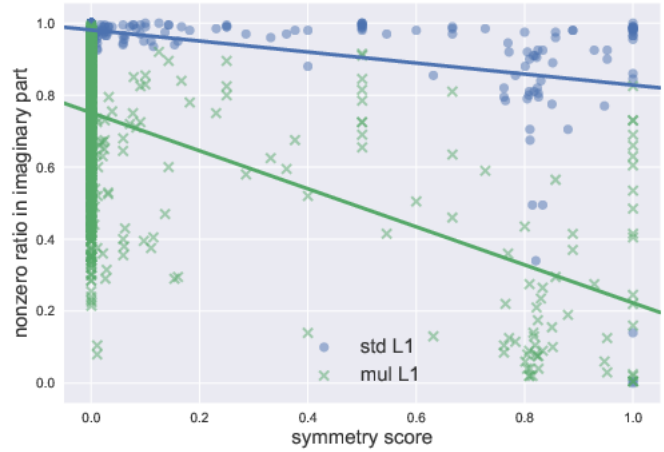
and compute this score for all relations in the FB15k training set. $\mathscr{T}_r^{\text{sym}}$ enumerates the positive triplets (s,r,o) where the interchanged triplet (o,r,s) is also positive. By definition, symmetric relation r should give $\text{sym}(r) = 1.0$ and asymmetric relation r should give $\text{sym}(r) = 0.0$. Thus, this score should indicate the degree of symmetry/asymmetry of relations.

In Figure 5.3, we plot the percentage of non-zero elements in the real and imaginary parts of its representation vector $\mathbf{w}_r$ against $\text{sym}(r)$ for each relation r. Panels (a) and (b) are for $\text{Re}(\mathbf{w}_r)$ and $\text{Im}(\mathbf{w}_r)$, respectively. It is expected that the higher (resp. lower) the symmetry is, the more imaginary (resp. real) parts become zero. With the multiplicative L1 regularizer (denoted by 'mul L1' and green crosses in the figure), the density of real parts correlates with symmetry, and the density of imaginary parts inversely correlates with symmetry. By contrast, correlation is weak or not observable for the standard L1 regularizer ('std L1'; blue circles). As well as demonstrated on synthetic data, we conclude that the gains in Table 5.3 come mainly from the more desirable representations for symmetric/asymmetric relations learned with the multiplicative L1 regularization.

However, compared to the results on synthetic data shown in Figure 5.1, many non-zero values exist in $\text{Re}(\mathbf{w}_r)$ and $\text{Im}(\mathbf{w}_r)$ even for completely symmetric/asymmetric relations. We suspect that this is related to the fact that an asymmetric relations does not imply an skew-symmetric truth-value matrix; even if a relation is asymmetric, there are many entity pairs (e_1,e_2) such that neither of (e_1,r,e_2) or (e_2,r,e_1) holds.



(a) The real part on the FB15k dataset



(b) The imaginary part on the FB15k dataset

Figure 5.3: The scatter plots showing the degree of sparsity against the symmetry score for each relation.

Chapter 6

Conclusions

In this work, we have presented a new regularizer for ComplEx to encourage vector representations of relations to be more symmetric or skew-symmetric in accordance with data. In the experiments, the proposed regularizer improved over the original ComplEx (with only L2 regularization) on FB15k, as well as WN18 with limited training data.

Recently, researchers have been making attempt to leverage background knowledge to improve KBC, such as by incorporating information of entity types, hierarchy, and relation attributes into the models [8, 11]. Our method does not assume background knowledge but adapt to symmetry/skew-symmetry of relations in a data-driven manner. However, learning more diverse knowledge about relations from data should be an interesting future research topic. We would also like to consider different type of regularization for entity vectors.

As another future research direction, we would like to develop a better strategy for sampling negative triplets that are suitable for our method.

Bibliography

[1] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor. Freebase: A collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data (SIGMOD '08)*, pp. 1247–1250, 2008.

[2] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems 26 (NIPS '13)*, pp. 2787–2795, 2013.

[3] J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159, July 2011.

[4] M. Faruqui, Y. Tsvetkov, D. Yogatama, C. Dyer, and N. A. Smith. Sparse overcomplete word vector representations. In *Proceedings of the 53rd Annual Meeting of the Association of Computational Linguistics (ACL '15)*, pp. 1491–1500, 2015.

[5] C. Fellbaum. *WordNet: An Electronic Lexical Database*. MIT Press, 1998.

[6] K. Hayashi and M. Shimbo. On the equivalence of holographic and complex embeddings for link prediction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL '17)*, pp. 554–559, 2017.

[7] D. Kong, R. Fujimaki, J. Liu, F. Nie, and C. Ding. Exclusive feature learning on arbitrary structures via $\ell_{1,2}$-norm. In *Advances in Neural Information Processing Systems 27*, pp. 1655–1663, 2014.

[8] D. Krompaß, S. Baier, and V. Tresp. Type-constrained representation learning in knowledge graphs. In *International Semantic Web Conference*, pp. 640–655. Springer, 2015.

[9] H. Liu, Y. Wu, and Y. Yang. Analogical inference for multi-relational embeddings. In *Proceedings of the 34th International Conference on Machine Learning (ICML '17)*, pp. 2168–2178, 2017.

[10] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv*, 1301.3781, 2013.

[11] P. Minervini, L. Costabello, E. Muñoz, V. Nováček, and P.-Y. Vandenbussche. Regularizing knowledge graph embeddings via equivalence and inversion axioms. In *Proceedings of the 2017 European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD '17)*, 2017.

[12] M. Nickel, K. Murphy, V. Tresp, and E. Gabrilovich. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104(1):11–33, 2016.

[13] M. Nickel, L. Rosasco, and T. Poggio. Holographic embeddings of knowledge graphs. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI '16)*, pp. 1955–1961, 2016.

[14] M. Nickel, V. Tresp, and H.-P. Kriegel. A three-way model for collective learning on multi-relational data. In *Proceedings of the 28th International Conference on Machine Learning (ICML '11)*, pp. 809–816, 2011.

[15] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP '14)*, pp. 1532–1543, 2014.

[16] F. M. Suchanek, G. Kasneci, and G. Weikum. Yago: A core of semantic knowledge. In *Proceedings of the 16th International Conference on World Wide Web (WWW '07)*, pp. 697–706, 2007.

[17] F. Sun, J. Guo, Y. Lan, J. Xu, and X. Cheng. Sparse word embeddings using ℓ_1 regularized online learning. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI '16)*, pp. 2915–2921, 2016.

[18] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58:267–288, 1994.

[19] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society Series B*, 67(1):91–108, 2005.

[20] T. Trouillon, C. R. Dance, É. Gaussier, and G. Bouchard. Decomposing real square matrices via unitary diagonalization. *arXiv*, 1605.07103, 2016.

[21] T. Trouillon, J. Welbl, S. Riedel, E. Gaussier, and G. Bouchard. Complex embeddings for simple link prediction. In *Proceedings of the 33rd International Conference on Machine Learning (ICML '16)*, pp. 2071–2080, 2016.

[22] L. Xiao. Dual averaging method for regularized stochastic learning and online optimization. In *Advances in Neural Information Processing Systems 22 (NIPS '09)*, 2009.

[23] B. Yang, W. Yih, X. He, J. Gao, and L. Deng. Embedding entities and relations for learning and inference in knowledge bases. *Proceedings of the 3rd International Conference on Learning Representations (ICLR '15)*, 2015.

List of Publications

International Conference

1. Hitoshi Manabe, Katsuhiko Hayashi, and Masashi Shimbo. Data-dependent Learning of Symmetry/Antisymmetry Relations for Knowledge Base Completion. The Association for the Advancement of Artificial Intelligence (AAAI), 2018.

Other Publications

1. 真鍋 陽俊, 林 克彦, 新保 仁. 知識ベース埋め込みのためのペアワイズ積 L1 正則化. 言語処理学会第 24 回年次大会, 2018.