Knowledge Graph Completion via Complex Tensor Factorization

Théo Trouillon   THEO.TROUILLON@IMAG.FR
Univ. Grenoble Alpes, 700 avenue Centrale, 38401 Saint Martin d'Hères, France

Christopher R. Dance   CHRIS.DANCE@NAVERLABS.COM
NAVER LABS Europe, 6 chemin de Maupertuis, 38240 Meylan, France

Éric Gaussier   ERIC.GAUSSIER@IMAG.FR
Univ. Grenoble Alpes, 700 avenue Centrale, 38401 Saint Martin d'Hères, France

Johannes Welbl   J.WELBL@CS.UCL.AC.UK
Sebastian Riedel   S.RIEDEL@CS.UCL.AC.UK
University College London, Gower St, London WC1E 6BT, United Kingdom

Guillaume Bouchard   G.BOUCHARD@CS.UCL.AC.UK
Bloomsbury AI, 115 Hampstead Road, London NW1 3EE, United Kingdom
University College London, Gower St, London WC1E 6BT, United Kingdom

Editor: Luc De Raedt

Abstract

In statistical relational learning, knowledge graph completion deals with automatically understanding the structure of large knowledge graphs—labeled directed graphs—and predicting missing relationships—labeled edges. State-of-the-art embedding models propose different trade-offs between modeling expressiveness, and time and space complexity. We reconcile both expressiveness and complexity through the use of complex-valued embeddings and explore the link between such complex-valued embeddings and unitary diagonalization. We corroborate our approach theoretically and show that *all* real square matrices—thus all possible relation/adjacency matrices—are the real part of some unitarily diagonalizable matrix. This results opens the door to a lot of other applications of square matrices factorization. Our approach based on complex embeddings is arguably simple, as it only involves a Hermitian dot product, the complex counterpart of the standard dot product between real vectors, whereas other methods resort to more and more complicated composition functions to increase their expressiveness. The proposed complex embeddings are scalable to large data sets as it remains linear in both space and time, while consistently outperforming alternative approaches on standard link prediction benchmarks.[1]

Keywords: complex embeddings, tensor factorization, knowledge graph, matrix completion, statistical relational learning

1. Introduction

Web-scale knowledge graph provide a structured representation of world knowledge, with projects such as the Google Knowledge Vault (Dong et al., 2014). They enable a wide range of applications including recommender systems, question answering and automated

---

1. Code is available at: https://github.com/ttrouill/complex

personal agents. The incompleteness of these knowledge graphs—also called knowledge bases—has stimulated research into predicting missing entries, a task known as *link prediction* or *knowledge graph completion*. The need for high quality predictions required by link prediction applications made it progressively become the main problem in statistical relational learning (Getoor and Taskar, 2007), a research field interested in relational data representation and modeling.

Knowledge graphs were born with the advent of the Semantic Web, pushed by the World Wide Web Consortium (W3C) recommendations. Namely, the Resource Description Framework (RDF) standard, that underlies knowledge graphs' data representation, provides for the first time a common framework across all connected information systems to share their data under the same paradigm. Being more expressive than classical relational databases, all existing relational data can be translated into RDF knowledge graphs (Sahoo et al., 2009).

Knowledge graphs express data as a directed graph with labeled edges (relations) between nodes (entities). Natural redundancies between the recorded relations often make it possible to fill in the missing entries of a knowledge graph. As an example, the relation `CountryOfBirth` could not be recorded for all entities, but it can be inferred if the relation `CityOfBirth` is known. The goal of link prediction is the automatic discovery of such regularities. However, many relations are non-deterministic: the combination of the two facts `IsBornIn(John,Athens)` and `IsLocatedIn(Athens,Greece)` does not always imply the fact `HasNationality(John,Greece)`. Hence, it is natural to handle inference probabilistically, and jointly with other facts involving these relations and entities. To this end, an increasingly popular method is to state the knowledge graph completion task as a 3D binary tensor completion problem, where each tensor slice is the adjacency matrix of one relation in the knowledge graph, and compute a decomposition of this partially-observed tensor from which its missing entries can be completed.

Factorization models with low-rank embeddings were popularized by the Netflix challenge (Koren et al., 2009). A partially-observed matrix or tensor is decomposed into a product of embedding matrices with much smaller dimensions, resulting in fixed-dimensional vector representations for each entity and relation in the graph, that allow completion of the missing entries. For a given fact $r(s,o)$ in which the subject entity s is linked to the object entity o through the relation r, a score for the fact can be recovered as a multilinear product between the embedding vectors of s, r and o, or through more sophisticated composition functions (Nickel et al., 2016a).

Binary relations in knowledge graphs exhibit various types of patterns: hierarchies and compositions like `FatherOf`, `OlderThan` or `IsPartOf`, with strict/non-strict orders or preorders, and equivalence relations like `IsSimilarTo`. These characteristics maps to different combinations of the following properties: reflexivity/irreflexivity, symmetry/antisymmetry and transitivity. As described in Bordes et al. (2013a), a relational model should (i) be able to learn all combinations of such properties, and (ii) be linear in both time and memory in order to scale to the size of present-day knowledge graphs, and keep up with their growth.

A natural way to handle any possible set of relations is to use the classic canonical polyadic (CP) decomposition (Hitchcock, 1927), which yields two different embeddings for each entity and thus low prediction performances as shown in Section 5. With unique entity embeddings, multilinear products scale well and can naturally handle both symmetry

and (ir)-reflexivity of relations, and when combined with an appropriate loss function, dot products can even handle transitivity (Bouchard et al., 2015). However, dealing with antisymmetric—and more generally asymmetric—relations has so far almost always implied superlinear time and space complexity (Nickel et al., 2011; Socher et al., 2013) (see Section 2), making models prone to overfitting and not scalable. Finding the best trade-off between expressiveness, generalization and complexity is the keystone of embedding models.

In this work, we argue that the standard dot product between embeddings can be a very effective composition function, provided that one uses the right *representation*: instead of using embeddings containing real numbers, we discuss and demonstrate the capabilities of complex embeddings. When using complex vectors, that is vectors with entries in $\mathbb{C}$, the dot product is often called the *Hermitian* (or sesquilinear) dot product, as it involves the conjugate-transpose of one of the two vectors. As a consequence, the dot product is not symmetric any more, and facts about one relation can receive different scores depending on the ordering of the entities involved in the fact. In summary, complex embeddings naturally represent arbitrary relations while retaining the efficiency of a dot product, that is linearity in both space and time complexity.

This paper extends a previously published article (Trouillon et al., 2016). This extended version adds proofs of existence of the proposed model in both single and multi-relational settings, as well as proofs of the non-uniqueness of the complex embeddings for a given relation. Bounds on the rank of the proposed decomposition are also demonstrated and discussed. The learning algorithm is provided in more details, and more experiments are provided, especially regarding the training time of the models.

The remainder of the paper is organized as follows. We first provide justification and intuition for using complex embeddings in the square matrix case (Section 2), where there is only a single type of relation between entities, and show the existence of the proposed decomposition for all possible relations. The formulation is then extended to a stacked set of square matrices in a third-order tensor to represent multiple relations (Section 3). The stochastic gradient descent algorithm used to learn the model is detailed in Section 4, where we present an equivalent reformulation of the proposed model that involves only real embeddings. This should help practitioners when implementing our method, without requiring the use of complex numbers in their software implementation. We then describe experiments on large-scale public benchmark knowledge graphs in which we empirically show that this representation leads not only to simpler and faster algorithms, but also gives a systematic accuracy improvement over current state-of-the-art alternatives (Section 5). Related work is discussed in Section 6.

2. Relations as the Real Parts of Low-Rank Normal Matrices

We consider in this section a simplified link prediction task with a single relation, and introduce complex embeddings for low-rank matrix factorization.

We will first discuss the desired properties of embedding models, show how this problem relates to the spectral theorems, and discuss the classes of matrices these theorems encompass in the real and in the complex case. We then propose a new matrix decomposition—to the best of our knowledge—and a proof of its existence for all real square matrices. Finally we discuss the rank of the proposed decomposition.

2.1 Modeling Relations

Let $\mathcal{E}$ be a set of entities, with $|\mathcal{E}| = n$. The truth of the single relation holding between two entities is represented by a sign value $y_{so} \in \{-1, 1\}$, where 1 represents true facts and -1 false facts, $s \in \mathcal{E}$ is the subject entity and $o \in \mathcal{E}$ is the object entity. The probability for the relation holding true is given by

$$P(y_{so} = 1) = \sigma(x_{so}) \tag{1}$$

where $X \in \mathbb{R}^{n \times n}$ is a latent matrix of scores indexed by the subject (rows) and object entities (columns), Y is a partially-observed sign matrix indexed in identical fashion, and σ is a suitable sigmoid function. Throughout this paper we used the logistic inverse link function $\sigma(x) = \frac{1}{1+e^{-x}}$.

2.1.1 HANDLING BOTH ASYMMETRY AND UNIQUE ENTITY EMBEDDINGS

In this work we pursue three objectives: finding a generic structure for X that leads to (i) a computationally efficient model, (ii) an expressive enough approximation of common relations in real world knowledge graphs, and (iii) good generalization performances in practice. Standard matrix factorization approximates X by a matrix product $UV^\top$, where U and V are two functionally-independent $n \times K$ matrices, K being the rank of the matrix. Within this formulation it is assumed that entities appearing as subjects are different from entities appearing as objects. In the Netflix challenge (Koren et al., 2009) for example, each row u_i corresponds to the user i and each column v_j corresponds to the movie j. This extensively studied type of model is closely related to the singular value decomposition (SVD) and fits well with the case where the matrix X is rectangular.

However, in many knowledge graph completion problems, the same entity i can appear as both subject *or* object and will have two different embedding vectors, u_i and v_i, depending on whether it appears as subject or object of a relation. It seems natural to learn unique embeddings of entities, as initially proposed by Nickel et al. (2011) and Bordes et al. (2011) and since then used systematically in other prominent approaches (Bordes et al., 2013b; Yang et al., 2015; Socher et al., 2013). In the factorization setting, using the same embeddings for left- and right-side factors boils down to a specific case of eigenvalue decomposition: *orthogonal diagonalization*.

Definition 1 *A real square matrix $X \in \mathbb{R}^{n \times n}$ is orthogonally diagonalizable if it can be written as $X = EWE^\top$, where $E, W \in \mathbb{R}^{n \times n}$, W is diagonal, and E orthogonal so that $EE^\top = E^\top E = I$ where I is the identity matrix.*

The spectral theorem for symmetric matrices tells us that a matrix is orthogonally diagonalizable if and only if it is symmetric (Cauchy, 1829). It is therefore often used to approximate covariance matrices, kernel functions and distance or similarity matrices.

However as previously stated, this paper is explicitly interested in problems where matrices—and thus the relation patterns they represent—can also be antisymmetric, or even not have any particular symmetry pattern at all (asymmetry). In order to both use a unique embedding for entities and extend the expressiveness to asymmetric relations, researchers have generalised the notion of dot products to *scoring functions*, also known as

Model	Scoring Function ϕ	Relation Parameters	$\mathcal{O}_{time}$	$\mathcal{O}_{space}$
CP (Hitchcock, 1927)	$\langle w_r, u_s, v_o \rangle$	$w_r \in \mathbb{R}^K$	$\mathcal{O}(K)$	$\mathcal{O}(K)$
RESCAL (Nickel et al., 2011)	$e_s^T W_r e_o$	$W_r \in \mathbb{R}^{K^2}$	$\mathcal{O}(K^2)$	$\mathcal{O}(K^2)$
TransE (Bordes et al., 2013b)	$-\|(e_s + w_r) - e_o\|_p$	$w_r \in \mathbb{R}^K$	$\mathcal{O}(K)$	$\mathcal{O}(K)$
NTN (Socher et al., 2013)	$u_r^\top f(e_s W_r^{[1..D]} e_o + V_r \begin{bmatrix} e_s \\ e_o \end{bmatrix} + b_r)$	$W_r \in \mathbb{R}^{K^2 D}, b_r \in \mathbb{R}^K$ $V_r \in \mathbb{R}^{2KD}, u_r \in \mathbb{R}^K$	$\mathcal{O}(K^2 D)$	$\mathcal{O}(K^2 D)$
DistMult (Yang et al., 2015)	$\langle w_r, e_s, e_o \rangle$	$w_r \in \mathbb{R}^K$	$\mathcal{O}(K)$	$\mathcal{O}(K)$
HolE (Nickel et al., 2016b)	$w_r^T(\mathcal{F}^{-1}[\overline{\mathcal{F}[e_s]} \odot \mathcal{F}[e_o]]))$	$w_r \in \mathbb{R}^K$	$\mathcal{O}(K \log K)$	$\mathcal{O}(K)$
ComplEx (this paper)	$\text{Re}(\langle w_r, e_s, \bar{e}_o \rangle)$	$w_r \in \mathbb{C}^K$	$\mathcal{O}(K)$	$\mathcal{O}(K)$

Table 1: Scoring functions of state-of-the-art latent factor models for a given fact $r(s, o)$, along with the representation of their relation parameters, and time and space (memory) complexity. K is the dimensionality of the embeddings. The entity embeddings e_s and e_o of subject s and object o are in $\mathbb{R}^K$ for each model, except for ComplEx, where $e_s, e_o \in \mathbb{C}^K$. $\bar{x}$ is the complex conjugate, and D is an additional latent dimension of the NTN model. $\mathcal{F}$ and $\mathcal{F}^{-1}$ denote respectively the Fourier transform and its inverse, $\odot$ is the element-wise product between two vectors, $\text{Re}(.)$ denotes the real part of a complex vector, and $\langle \cdot, \cdot, \cdot \rangle$ denotes the trilinear product.

composition functions, that allow more general combinations of embeddings. We briefly recall several examples of scoring functions in Table 1, as well as the extension proposed in this paper.

These models propose different trade-offs between the three essential points:

- Expressiveness, which is the ability to represent symmetric, antisymmetric and more generally asymmetric relations.

- Scalability, which means keeping linear time and space complexity scoring function.

- Generalization, for which having unique entity embeddings is critical.

RESCAL (Nickel et al., 2011) and NTN (Socher et al., 2013) are very expressive, but their scoring functions have quadratic complexity in the rank of the factorization. More recently the HolE model (Nickel et al., 2016b) proposes a solution that has quasi-linear complexity in time and linear space complexity. DistMult (Yang et al., 2015) can be seen as a joint orthogonal diagonalization with real embeddings, hence handling only symmetric relations. Conversely, TransE (Bordes et al., 2013b) handles symmetric relations to the price of strong constraints on its embeddings. The canonical-polyadic decomposition (CP) (Hitchcock, 1927) generalizes poorly with its different embeddings for entities as subject and as object.

We reconcile expressiveness, scalability and generalization by going back to the realm of well-studied matrix factorizations, and making use of complex linear algebra, a scarcely used tool in the machine learning community.

5

2.1.2 Decomposition in the Complex Domain

We introduce a new decomposition of real square matrices using unitary diagonalization, the generalization of orthogonal diagonalization to complex matrices. This allows decomposition of *arbitrary* real square matrices with unique representations of rows and columns.

Let us first recall some notions of complex linear algebra as well as specific cases of diagonalization of real square matrices, before building our proposition upon these results.

A complex-valued vector $x \in \mathbb{C}^K$, with $x = \text{Re}(x) + i\text{Im}(x)$ is composed of a real part $\text{Re}(x) \in \mathbb{R}^K$ and an imaginary part $\text{Im}(x) \in \mathbb{R}^K$, where i denotes the square root of -1. The conjugate $\overline{x}$ of a complex vector inverts the sign of its imaginary part: $\overline{x} = \text{Re}(x) - i\text{Im}(x)$.

Conjugation appears in the usual dot product for complex numbers, called the *Hermitian* product, or *sesquilinear* form, which is defined as:

$$\begin{aligned} \langle u, v \rangle \quad &:= \quad \bar{u}^\top v \\ &= \quad \text{Re}(u)^\top \text{Re}(v) + \text{Im}(u)^\top \text{Im}(v) \\ &\quad + i(\text{Re}(u)^\top \text{Im}(v) - \text{Im}(u)^\top \text{Re}(v)). \end{aligned}$$

A simple way to justify the Hermitian product for composing complex vectors is that it provides a valid topological norm in the induced vector space. For example, $\bar{x}^\top x = 0$ implies $x = 0$ while this is not the case for the bilinear form $x^\top x$ as there are many complex vectors x for which $x^\top x = 0$.

This yields an interesting property of the Hermitian product concerning the order of the involved vectors: $\langle u, v \rangle = \overline{\langle v, u \rangle}$, meaning that the real part of the product is symmetric, while the imaginary part is antisymmetric.

For matrices, we shall write $X^* \in \mathbb{C}^{n \times m}$ for the conjugate-transpose $X^* = (\overline{X})^\top = \overline{X^\top}$. The conjugate transpose is also often written $X^\dagger$ or X^H.

Definition 2 *A complex square matrix $X \in \mathbb{C}^{n \times n}$ is unitarily diagonalizable if it can be written as $X = EWE^*$, where $E, W \in \mathbb{C}^{n \times n}$, W is diagonal, and E is unitary such that $EE^* = E^*E = I$.*

Definition 3 *A complex square matrix X is normal if it commutes with its conjugate-transpose so that $XX^* = X^*X$.*

We can now state the spectral theorem for normal matrices.

Theorem 1 (Spectral theorem for normal matrices, von Neumann (1929)) *Let X be a complex square matrix. Then X is unitarily diagonalizable if and only if X is normal.*

It is easy to check that all real symmetric matrices are normal, and have pure real eigenvectors and eigenvalues. But the set of purely real normal matrices also includes all real antisymmetric matrices (useful to model hierarchical relations such as `IsOlder`), as well as all real orthogonal matrices (including permutation matrices), and many other matrices that are useful to represent binary relations, such as assignment matrices which represent bipartite graphs. However, far from all matrices expressed as $X = EWE^*$ are purely real, and Equation (1) requires the scores X to be purely real.

As we only focus on *real* square matrices in this work, let us summarize all the cases where X is real square and $X = EWE^*$ if X is unitarily diagonalizable, where $E, W \in \mathbb{C}^{n \times n}$, W is diagonal and E is unitary:

- X is symmetric if and only if X is orthogonally diagonalizable and E and W are purely real.

- X is normal and non-symmetric if and only if X is unitarily diagonalizable and E and W are *not* both purely real.

- X is not normal if and only if X is not unitarily diagonalizable.

We generalize all three cases by showing that, for any $X \in \mathbb{R}^{n \times n}$, there exists a unitary diagonalization in the complex domain, of which the real part equals X:

$$X = \mathrm{Re}(EWE^*) \,. \tag{2}$$

In other words, the unitary diagonalization is projected onto the real subspace.

Theorem 2 *Suppose $X \in \mathbb{R}^{n \times n}$ is a real square matrix. Then there exists a normal matrix $Z \in \mathbb{C}^{n \times n}$ such that $\mathrm{Re}(Z) = X$.*

Proof Let $Z := X + iX^\top$. Then

$$Z^* = X^\top - iX = -i(iX^\top + X) = -iZ \,,$$

so that

$$ZZ^* = Z(-iZ) = (-iZ)Z = Z^*Z \,.$$

Therefore Z is normal. ∎

Note that there also exists a normal matrix $Z = X^\top + iX$ such that $\mathrm{Im}(Z) = X$.

Following Theorem 1 and Theorem 2, any real square matrix can be written as the real part of a complex diagonal matrix through a unitary change of basis.

Corollary 1 *Suppose $X \in \mathbb{R}^{n \times n}$ is a real square matrix. Then there exist $E, W \in \mathbb{C}^{n \times n}$, where E is unitary, and W is diagonal, such that $X = \mathrm{Re}(EWE^*)$.*

Proof From Theorem 2, we can write $X = \mathrm{Re}(Z)$, where Z is a normal matrix, and from Theorem 1, Z is unitarily diagonalizable. ∎

Applied to the knowledge graph completion setting, the rows of E here are vectorial representations of the entities corresponding to rows and columns of the relation score matrix X. The score for the relation holding true between entities s and o is hence

$$x_{so} = \mathrm{Re}(e_s^\top W \bar{e}_o) \tag{3}$$

where $e_s, e_o \in \mathbb{C}^n$ and $W \in \mathbb{C}^{n \times n}$ is diagonal. For a given entity, its subject embedding vector is the complex conjugate of its object embedding vector.

To illustrate this difference of expressiveness with respect to real-valued embeddings, let us consider two complex embeddings $e_s, e_o \in \mathbb{C}$ of dimension 1, with arbitrary values: $e_s = 1 - 2i$, and $e_o = -3 + i$; as well as their real-valued, twice-bigger counterparts: $e'_s = \left(\begin{smallmatrix} 1 \\ -2 \end{smallmatrix} \right) \in \mathbb{R}^2$ and $e'_o = \left(\begin{smallmatrix} -3 \\ 1 \end{smallmatrix} \right) \in \mathbb{R}^2$. In the real-valued case, that corresponds to the DISTMULT model (Yang et al., 2015), the score is $x_{so} = e'^{\top}_s W' e'_o$. Figure 1 represents the heatmaps of the scores x_{so} and x_{os}, as a function of $W \in \mathbb{C}$ in the complex-valued case, and as a function of $W' \in \mathbb{R}^2$ diagonal in the real-valued case. In the real-valued case, that is symmetric in the subject and object entities, the scores x_{so} and x_{os} are equal for any value of $W' \in \mathbb{R}^2$ diagonal. Whereas in the complex-valued case, the variation of $W \in \mathbb{C}$ allows to score x_{so} and x_{os} with any desired pair of values.

This decomposition however is non-unique, a simple example of this non-uniqueness is obtained by adding a purely imaginary constant to the eigenvalues. Let $X \in \mathbb{R}^{n \times n}$, and $X = \mathrm{Re}(EWE^*)$ where E is unitary, W is diagonal. Then for any real constant $c \in \mathbb{R}$ we have:

$$
\begin{aligned}
X &= \mathrm{Re}(E(W + icI)E^*) \\
&= \mathrm{Re}(EWE^* + icEIE^*) \\
&= \mathrm{Re}(EWE^* + icI) \\
&= \mathrm{Re}(EWE^*) \,.
\end{aligned}
$$

In general, there are many other possible couples of matrices E and W that preserve the real part of the decomposition. In practice however this is no synonym of low generalization abilities, as many effective matrix and tensor decomposition methods used in machine learning lead to non-unique solutions (Paatero and Tapper, 1994; Nickel et al., 2011). In this case also, the learned representations prove useful as shown in the experimental section.

2.2 Low-Rank Decomposition

Addressing knowledge graph completion with data-driven approaches assumes that there is a sufficient regularity in the observed data to generalize to unobserved facts. When formulated as a matrix completion problem, as it is the case in this section, one way of implementing this hypothesis is to make the assumption that the matrix has a low rank or approximately low rank. We first discuss the rank of the proposed decomposition, and then introduce the sign-rank and extend the bound developed on the rank to the sign-rank.

2.2.1 RANK UPPER BOUND

First, we recall one definition of the rank of a matrix (Horn and Johnson, 2012).

Definition 4 *The rank of an m-by-n complex matrix* $\mathrm{rank}(X) = \mathrm{rank}(X^{\top}) = k$, *if X has exactly k linearly independent columns.*

Also note that if X is diagonalizable so that $X = EWE^{-1}$ with $\mathrm{rank}(X) = k$, then W has k non-zero diagonal entries for some diagonal W and some invertible matrix E. From this it is easy to derive a known additive property of the rank:

$$
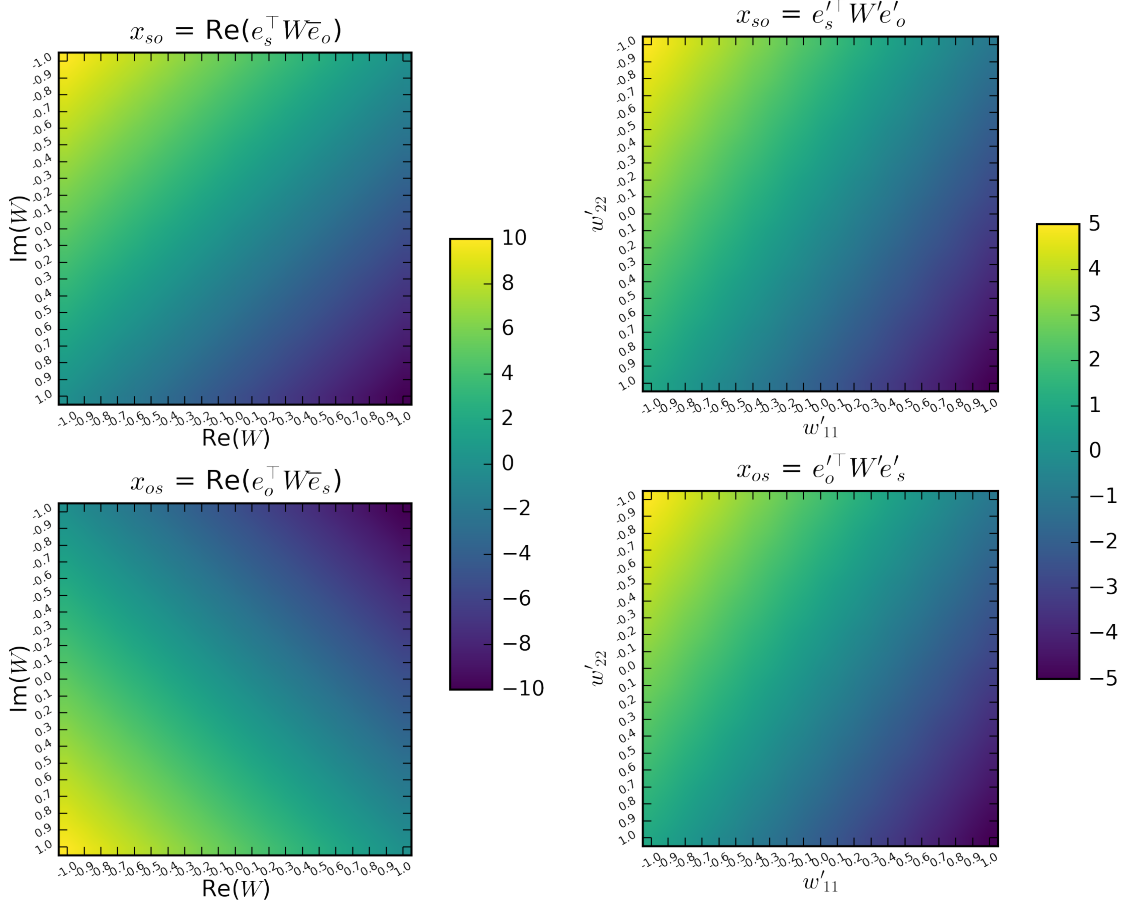\mathrm{rank}(B + C) \le \mathrm{rank}(B) + \mathrm{rank}(C) \tag{4}
$$

Figure 1: Left: Scores $x_{so} = \mathrm{Re}(e_s^\top W \bar{e}_o)$ (top) and $x_{os} = \mathrm{Re}(e_o^\top W \bar{e}_s)$ (bottom) for the proposed complex-valued decomposition, plotted as a function of $W \in \mathbb{C}$, for fixed entity embeddings $e_s = 1 - 2i$, and $e_o = -3 + i$. Right: Scores $x_{so} = e_s'^\top W' e_o'$ (top) and $x_{os} = e_o'^\top W' e_s'$ (bottom) for the corresponding real-valued decomposition with the same number of free real-valued parameters (*i.e.* in twice the dimension), plotted as a function of $W' \in \mathbb{R}^2$ diagonal, for fixed entity embeddings $e_s' = \left(\begin{smallmatrix} 1 \\ -2 \end{smallmatrix}\right)$ and $e_o' = \left(\begin{smallmatrix} -3 \\ 1 \end{smallmatrix}\right)$. By varying $W \in \mathbb{C}$, the proposed complex-valued decomposition can attribute any pair of scores to x_{so} and x_{os}, whereas $x_{so} = x_{os}$ for all $W' \in \mathbb{R}^2$ with the real-valued decomposition.

where $B, C \in \mathbb{C}^{m \times n}$.

We now show that any rank k real square matrix can be reconstructed from a $2k$-dimensional unitary diagonalization.

Corollary 2 *Suppose $X \in \mathbb{R}^{n \times n}$ and $\mathrm{rank}(X) = k$. Then there exist $E \in \mathbb{C}^{n \times 2k}$ such that the columns of E form an orthonormal basis of $\mathbb{C}^{2k}$, $W \in \mathbb{C}^{2k \times 2k}$ is diagonal, and $X = \mathrm{Re}(EWE^*)$.*

Proof Consider the complex square matrix $Z := X + iX^\top$. We have $\text{rank}(iX^\top) = \text{rank}(X^\top) = \text{rank}(X) = k$.

From Equation (4), $\text{rank}(Z) \leq \text{rank}(X) + \text{rank}(iX^\top) = 2k$.

The proof of Theorem 2 shows that Z is normal. Thus $Z = EWE^*$ with $E \in \mathbb{C}^{n \times 2k}$, $W \in \mathbb{C}^{2k \times 2k}$ where the columns of E form an orthonormal basis of $\mathbb{C}^{2k}$, and W is diagonal. ∎

Since E is not necessarily square, we replace the unitary requirement of Corollary 1 by the requirement that its columns form an orthonormal basis of its smallest dimension, $2k$.

Also, given that such decomposition always exists in dimension n (Theorem 2), this upper bound is not relevant when $\text{rank}(X) \geq \frac{n}{2}$.

2.2.2 SIGN-RANK UPPER BOUND

Since we encode the truth values of each fact with ± 1, we deal with square *sign matrices*: $Y \in \{-1, 1\}^{n \times n}$. Sign matrices have an alternative rank definition, the *sign-rank*.

Definition 5 *The sign-rank* $\text{rank}_\pm(Y)$ *of an m-by-n sign matrix Y, is the rank of the m-by-n real matrix of least rank that has the same sign-pattern as Y, so that*

$$\text{rank}_\pm(Y) := \min_{X \in \mathbb{R}^{m \times n}} \{\text{rank}(X) \,|\, \text{sign}(X) = Y\},$$

where $\text{sign}(X)_{ij} = \text{sign}(x_{ij})$.

We define the *sign function* of $c \in \mathbb{R}$ as

$$\text{sign}(c) = \begin{cases} 1 & \text{if } c \geq 0 \\ -1 & \text{otherwise} \end{cases}$$

where the value $c = 0$ is here arbitrarily assigned to 1 to allow zero entries in X, conversely to the stricter usual definition of the sign-rank.

To make generalization possible, we hypothesize that the true matrix Y has a low sign-rank, and thus can be reconstructed by the sign of a low-rank score matrix X. The low sign-rank assumption is theoretically justified by the fact that the sign-rank is a natural complexity measure of sign matrices (Linial et al., 2007) and is linked to learnability (Alon et al., 2016) and empirically confirmed by the wide success of factorization models (Nickel et al., 2016a).

Using Corollary 2, we can now show that any square sign matrix of sign-rank k can be reconstructed from a rank $2k$ unitary diagonalization.

Corollary 3 *Suppose $Y \in \{-1, 1\}^{n \times n}$, $\text{rank}_\pm(Y) = k$. Then there exists $E \in \mathbb{C}^{n \times 2k}$, $W \in \mathbb{C}^{2k \times 2k}$ where the columns of E form an orthonormal basis of $\mathbb{C}^{2k}$, and W is diagonal, such that $Y = \text{sign}(\text{Re}(EWE^*))$.*

Proof By definition, if $\text{rank}_\pm(Y) = k$, there exists a real square matrix X such that $\text{rank}(X) = k$ and $\text{sign}(X) = Y$. From Corollary 2, $X = \text{Re}(EWE^*)$ where $E \in \mathbb{C}^{n \times 2k}$, $W \in \mathbb{C}^{2k \times 2k}$ where the columns of E form an orthonormal basis of $\mathbb{C}^{2k}$, and W is diagonal. ∎

Previous attempts to approximate the sign-rank in relational learning did not use complex numbers. They showed the existence of compact factorizations under conditions on the sign matrix (Nickel et al., 2014), or only in specific cases (Bouchard et al., 2015). In contrast, our results show that if a square sign matrix has sign-rank k, then it can be exactly decomposed through a $2k$-dimensional unitary diagonalization.

Although we can only show the existence of a complex decomposition of rank $2k$ for a matrix with sign-rank k, the sign rank of Y is often *much* lower than the rank of Y, as we do not know any matrix $Y \in \{-1, 1\}^{n \times n}$ for which $\text{rank}_\pm(Y) > \sqrt{n}$ (Alon et al., 2016). For example, the $n \times n$ identity matrix has rank n, but its sign-rank is only 3! By swapping the columns $2j$ and $2j - 1$ for j in $1, \ldots, \frac{n}{2}$, the identity matrix corresponds to the relation `marriedTo`, a relation known to be hard to factorize over the reals (Nickel et al., 2014), since the rank is invariant by row/column permutations. Yet our model can express it at most in rank 6, for any n.

Hence, by enforcing a low-rank $K \ll n$ on EWE^*, individual relation scores $x_{so} = \text{Re}(e_s^\top W \bar{e}_o)$ between entities s and o can be efficiently predicted, as $e_s, e_o \in \mathbb{C}^K$ and $W \in \mathbb{C}^{K \times K}$ is diagonal.

Finding the K that matches the sign-rank of Y corresponds to finding the smallest K that brings the 0–1 loss on X to 0, as link prediction can be seen as binary classification of the facts. In practice, and as classically done in machine learning to avoid this NP-hard problem, we use a continuous surrogate of the 0–1 loss, in this case the logistic loss as described in Section 4, and validate models on different values of K, as described in Section 5.

2.2.3 RANK BOUND DISCUSSION

Corollaries 2 and 3 use the aforementioned subadditive property of the rank to derive the $2k$ upper bound. Let us give an example for which this bound is strictly greater than k.

Consider the following 2-by-2 sign matrix:

$$Y = \begin{bmatrix} -1 & -1 \\ 1 & 1 \end{bmatrix}.$$

Not only is this matrix not normal, but one can also easily check that there is no *real* normal 2-by-2 matrix that has the same sign-pattern as Y. Clearly, Y is a rank 1 matrix since its columns are linearly dependent, hence its sign-rank is also 1. From Corollary 3, we know that there is a normal matrix whose real part has the same sign-pattern as Y, and whose rank is at most 2.

However, there is no rank 1 unitary diagonalization of which the real part equals Y. Otherwise we could find a 2-by-2 complex matrix Z such that $\text{Re}(z_{11}) < 0$ and $\text{Re}(z_{22}) > 0$, where $z_{11} = e_1 w \bar{e}_1 = w|e_1|^2$, $z_{22} = e_2 w \bar{e}_2 = w|e_2|^2$, $e \in \mathbb{C}^2, w \in \mathbb{C}$. This is obviously unsatisfiable. This example generalizes to any n-by-n square sign matrix that only has -1 on its first row and is hence rank 1, the same argument holds considering $\text{Re}(z_{11}) < 0$ and $\text{Re}(z_{nn}) > 0$.

This example shows that the upper bound on the rank of the unitary diagonalization showed in Corollaries 2 and 3 can be strictly greater than k, the rank or sign-rank, of the decomposed matrix. However, there might be other examples for which the addition of

an imaginary part could—additionally to making the matrix normal—create some linear dependence between the rows/columns and thus decrease the rank of the matrix, up to a factor of 2.

We summarize this section in three points:

1. The proposed factorization encompasses all possible score matrices X for a single binary relation.

2. By construction, the factorization is well suited to represent both symmetric and antisymmetric relations.

3. Relation patterns can be efficiently approximated with a low-rank factorization using complex-valued embeddings.

3. Extension to Multi-Relational Data

Let us now extend the previous discussion to models with multiple relations. Let $\mathcal{R}$ be the set of relations, with $|\mathcal{R}| = m$. We shall now write $\mathbf{X} \in \mathbb{R}^{m \times n \times n}$ for the score tensor, $X_r \in \mathbb{R}^{n \times n}$ for the score matrix of the relation $r \in \mathcal{R}$, and $\mathbf{Y} \in \{-1, 1\}^{m \times n \times n}$ for the partially-observed sign tensor.

Given one relation $r \in \mathcal{R}$ and two entities $s, o \in \mathcal{E}$, the probability that the fact $r(s,o)$ is true given by:

$$P(y_{rso} = 1) = \sigma(x_{rso}) = \sigma(\phi(r, s, o; \Theta)) \tag{5}$$

where ϕ is the scoring function of the model considered and Θ denotes the model parameters. We denote the set of all possible facts (or triples) for a knowledge graph by $\mathcal{T} = \mathcal{R} \times \mathcal{E} \times \mathcal{E}$. While the tensor $\mathbf{X}$ as a whole is unknown, we assume that we observe a set of true and false triples $\Omega = \{((r, s, o), y_{rso}) \,|\, (r, s, o) \in \mathcal{T}_\Omega\}$ where $y_{rso} \in \{-1, 1\}$ and $\mathcal{T}_\Omega \subseteq \mathcal{T}$ is the set of observed triples. The goal is to find the probabilities of entries $y_{r's'o'}$ for a set of targeted unobserved triples $\{(r', s', o') \in \mathcal{T} \setminus \mathcal{T}_\Omega\}$.

Depending on the scoring function $\phi(r, s, o; \Theta)$ used to model the score tensor $\mathbf{X}$, we obtain different models. Examples of scoring functions are given in Table 1.

3.1 Complex Factorization Extension to Tensors

The single-relation model is extended by jointly factorizing all the square matrices of scores into a 3$^{\text{rd}}$-order tensor $\mathbf{X} \in \mathbb{R}^{m \times n \times n}$, with a different diagonal matrix $W_r \in \mathbb{C}^{K \times K}$ for each relation r, and by sharing the entity embeddings $E \in \mathbb{C}^{n \times K}$ across all relations:

$$\begin{aligned}
\phi(r, s, o; \Theta) &= \text{Re}(e_s^\top W_r \bar{e}_o) \\
&= \text{Re}(\sum_{k=1}^{K} w_{rk} e_{sk} \bar{e}_{ok}) \\
&= \text{Re}(\langle w_r, e_s, \bar{e}_o \rangle) \tag{6}
\end{aligned}$$

where K is the rank hyperparameter, $e_s, e_o \in \mathbb{C}^K$ are the rows in E corresponding to the entities s and o, $w_r = \text{diag}(W_r) \in \mathbb{C}^K$ is a complex vector, and $\langle a, b, c \rangle := \sum_k a_k b_k c_k$ is the

component-wise multilinear dot product[2]. For this scoring function, the set of parameters Θ is $\{e_i, w_r \in \mathbb{C}^K, i \in \mathcal{E}, r \in \mathcal{R}\}$. This resembles the real part of a complex matrix decomposition as in the single-relation case discussed above. However, we now have a different vector of eigenvalues for every relation. Expanding the real part of this product gives:

$$
\begin{aligned}
\mathrm{Re}(\langle w_r, e_s, \bar{e}_o \rangle) \;=\; & \langle \mathrm{Re}(w_r), \mathrm{Re}(e_s), \mathrm{Re}(e_o) \rangle \\
& + \langle \mathrm{Re}(w_r), \mathrm{Im}(e_s), \mathrm{Im}(e_o) \rangle \\
& + \langle \mathrm{Im}(w_r), \mathrm{Re}(e_s), \mathrm{Im}(e_o) \rangle \\
& - \langle \mathrm{Im}(w_r), \mathrm{Im}(e_s), \mathrm{Re}(e_o) \rangle .
\end{aligned}
\tag{7}
$$

These equations provide two interesting views of the model:

- *Changing the representation*: Equation (6) would correspond to DISTMULT with real embeddings (see Table 1), but handles asymmetry thanks to the complex conjugate of the object-entity embedding.

- *Changing the scoring function*: Equation (7) only involves real vectors corresponding to the real and imaginary parts of the embeddings and relations.

By separating the real and imaginary parts of the relation embedding w_r as shown in Equation (7), it is apparent that these parts naturally act as weights on each latent dimension: $\mathrm{Re}(w_r)$ over the real part of $\langle e_o, e_s \rangle$ which is symmetric, and $\mathrm{Im}(w)$ over the imaginary part of $\langle e_o, e_s \rangle$ which is antisymmetric.

Indeed, the decomposition of each score matrix X_r for each $r \in \mathcal{R}$ can be written as the sum of a symmetric matrix and an antisymmetric matrix. To see this, let us rewrite the decomposition of each score matrix X_r in matrix notation. We write the real part of matrices with primes $E' = \mathrm{Re}(E)$ and imaginary parts with double primes $E'' = \mathrm{Im}(E)$:

$$
\begin{aligned}
X_r \;=\; & \mathrm{Re}(E W_r E^*) \\
=\; & \mathrm{Re}((E' + iE'')(W_r' + iW_r'')(E' - iE'')^\top) \\
=\; & (E' W_r' E'^\top + E'' W_r' E''^\top) + (E' W_r'' E''^\top - E'' W_r'' E'^\top) .
\end{aligned}
\tag{8}
$$

It is trivial to check that the matrix $E' W_r' E'^\top + E'' W_r' E''^\top$ is symmetric and that the matrix $E' W_r'' E''^\top - E'' W_r'' E'^\top$ is antisymmetric. Hence this model is well suited to model jointly symmetric and antisymmetric relations between pairs of entities, while still using the same entity representations for subjects and objects. When learning, it simply needs to collapse $W_r'' = \mathrm{Im}(W_r)$ to zero for symmetric relations $r \in \mathcal{R}$, and $W_r' = \mathrm{Re}(W_r)$ to zero for antisymmetric relations $r \in \mathcal{R}$, as X_r is indeed symmetric when W_r is purely real, and antisymmetric when W_r is purely imaginary.

From a geometrical point of view, each relation embedding w_r is an anisotropic scaling of the basis defined by the entity embeddings E, followed by a projection onto the real subspace.

---

2. This is not the Hermitian extension of the multilinear dot product as there appears to be no standard definition of the Hermitian multilinear product in the linear algebra literature.

3.2 Existence of the Tensor Factorization

Let us first discuss the existence of the multi-relational model where the rank of the decomposition $K \leq n$, which relates to simultaneous unitary decomposition.

Definition 6 *A family of matrices $X_1, \ldots, X_m \in \mathbb{C}^{n \times n}$ is simultaneously unitarily diagonalizable, if there is a single unitary matrix $E \in \mathbb{C}^{n \times n}$, such that $X_i = EW_iE^*$ for all i in $1, \ldots, m$, where $W_i \in \mathbb{C}^{n \times n}$ are diagonal.*

Definition 7 *A family of normal matrices $X_1, \ldots, X_m \in \mathbb{C}^{n \times n}$ is a commuting family of normal matrices, if $X_iX_j^* = X_i^*X_j$, for all i, j in $1, \ldots, m$.*

Theorem 3 (see Horn and Johnson (2012)) *Suppose $\mathcal{F}$ is the family of matrices X_1, $\ldots$, $X_m \in \mathbb{C}^{n \times n}$. Then $\mathcal{F}$ is a commuting family of normal matrices if and only if $\mathcal{F}$ is simultaneously unitarily diagonalizable.*

To apply Theorem 3 to the proposed factorization, we would have to make the hypothesis that the relation score matrices X_r are a commuting family, which is too strong a hypothesis. Actually, the model is slightly different since we take only the real part of the tensor factorization. In the single-relation case, taking only the real part of the decomposition rids us of the normality requirement of Theorem 1 for the decomposition to exist, as shown in Theorem 2.

In the multiple-relation case, it is an open question whether taking the real part of the simultaneous unitary diagonalization will enable us to decompose families of arbitrary real square matrices—that is with a single unitary matrix E that has *at most n* columns. Though it seems unlikely, we could not find a counter-example yet.

However, by letting the rank of the tensor factorization K to be greater than n, we can show that the proposed tensor decomposition exists for families of arbitrary real square matrices, by simply concatenating the decomposition of Theorem 2 of each real square matrix X_i.

Theorem 4 *Suppose $X_1, \ldots, X_m \in \mathbb{R}^{n \times n}$. Then there exists $E \in \mathbb{C}^{n \times nm}$ and $W_i \in \mathbb{C}^{nm \times nm}$ are diagonal, such that $X_i = \mathrm{Re}(EW_iE^*)$ for all i in $1, \ldots, m$.*

Proof From Theorem 2 we have $X_i = \mathrm{Re}(E_iW_iE_i^*)$, where $W_i \in \mathbb{C}^{n \times n}$ is diagonal, and each $E_i \in \mathbb{C}^{n \times n}$ is unitary for all i in $1, \ldots, m$.

Let $E = [E_1 \ldots E_m]$, and

$$
\Lambda_i = \begin{bmatrix} \mathbf{0}^{((i-1)n) \times ((i-1)n)} & & \\ & W_i & \\ & & \mathbf{0}^{((m-i)n) \times ((m-i)n)} \end{bmatrix}
$$

where $\mathbf{0}^{l \times l}$ the zero $l \times l$ matrix. Therefore $X_i = \mathrm{Re}(E\Lambda_iE^*)$ for all i in $1, \ldots, m$. ∎

By construction, the rank of the decomposition is at most nm. When $m \leq n$, this bound actually matches the general upper bound on the rank of the canonical polyadic (CP) decomposition (Hitchcock, 1927; Kruskal, 1989). Since m corresponds to the number

of relations and n to the number of entities, m is always smaller than n in real world knowledge graphs, hence the bound holds in practice.

Though when it comes to relational learning, we might expect the actual rank to be much lower than nm for two reasons. The first one, as discussed above, is that we are dealing with sign tensors, hence the rank of the matrices X_r need only match the sign-rank of the partially-observed matrices Y_r. The second one is that the matrices are related to each other, as they all represent the same entities in different relations, and thus benefit from sharing latent dimensions. As opposed to the construction exposed in the proof of Theorem 4, where other relations dimensions are canceled out. In practice, the rank needed to generalize well is indeed much lower than nm as we show experimentally in Figure 5.

Also, note that with the construction of the proof of Theorem 4, the matrix $E = [E_1 \ldots E_m]$ is not unitary any more. However the unitary constraints in the matrix case serve only the proof of existence, which is just one solution among the infinite ones of same rank. In practice, imposing orthonormality is essentially a numerical commodity for the decomposition of dense matrices, through iterative methods for example (Saad, 1992). When it comes to matrix and tensor completion, and thus generalisation, imposing such constraints is more of a numerical hassle than anything else, especially for gradient methods. As there is no apparent link between orthonormality and generalisation properties, we did not impose these constraints when learning this model in the following experiments.

4. Algorithm

Algorithm 1 describes stochastic gradient descent (SGD) to learn the proposed multi-relational model with the AdaGrad learning-rate updates (Duchi et al., 2011). We refer to the proposed model as COMPLEX, for Complex Embeddings. We expose a version of the algorithm that uses only real-valued vectors, in order to facilitate its implementation. To do so, we use separate real-valued representations of the real and imaginary parts of the embeddings.

These real and imaginary part vectors are initialized with vectors having a zero-mean normal distribution with unit variance. If the training set Ω contains only positive triples, negatives are generated for each batch using the *local closed-world assumption* as in Bordes et al. (2013b). That is, for each triple, we randomly change either the subject or the object, to form a negative example. In this case the parameter $\eta > 0$ sets the number of negative triples to generate for each positive triple. Collision with positive triples in Ω is not checked, as it occurs rarely in real world knowledge graphs as they are largely sparse, and may also be computationally expensive.

Squared gradients are accumulated to compute AdaGrad learning rates, then gradients are updated. Every s iterations, the parameters Θ are evaluated over the evaluation set Ω_v (*evaluate_AP_or_MRR*$(\Omega_v; \Theta)$ function in Algorithm 1). If the data set contains both positive and negative examples, average precision (AP) is used to evaluate the model. If the data set contains only positives, then mean reciprocal rank (MRR) is used as average precision cannot be computed without true negatives. The optimization process is stopped when the measure considered decreases compared to the last evaluation (early stopping).

Bern(p) is the Bernoulli distribution, the *one_random_sample*($\mathcal{E}$) function sample uniformly one entity in the set of all entities $\mathcal{E}$, and the *sample_batch_of_size_b*(Ω, b) function sample b true and false triples uniformly at random from the training set Ω.

For a given embedding size K, let us rewrite Equation (7), by denoting the real part of embeddings with primes and the imaginary part with double primes: $e_i' = \text{Re}(e_i)$, $e_i'' = \text{Im}(e_i)$, $w_r' = \text{Re}(w_r)$, $w_r'' = \text{Im}(w_r)$. The set of parameters is $\Theta = \{e_i', e_i'', w_r', w_r'' \in \mathbb{R}^K, i \in \mathcal{E}, r \in \mathcal{R}\}$, and the scoring function involves only real vectors:

$$\phi(r, s, o; \Theta) = \quad \langle w_r', e_s', e_o' \rangle + \langle w_r', e_s'', e_o'' \rangle$$
$$+ \langle w_r'', e_s', e_o'' \rangle - \langle w_r'', e_s'', e_o' \rangle \qquad (9)$$

where each entity and each relation has two real embeddings.

Gradients are now easy to write:

$$\nabla_{e_s'}\phi(r, s, o; \Theta) = (w_r' \odot e_o') + (w_r'' \odot e_o''),$$
$$\nabla_{e_s''}\phi(r, s, o; \Theta) = (w_r' \odot e_o'') - (w_r'' \odot e_o'),$$
$$\nabla_{e_o'}\phi(r, s, o; \Theta) = (w_r' \odot e_s') - (w_r'' \odot e_s''),$$
$$\nabla_{e_o''}\phi(r, s, o; \Theta) = (w_r' \odot e_s'') + (w_r'' \odot e_s'),$$
$$\nabla_{w_r'}\phi(r, s, o; \Theta) = (e_s' \odot e_o') + (e_s'' \odot e_o''),$$
$$\nabla_{w_r''}\phi(r, s, o; \Theta) = (e_s' \odot e_o'') - (e_s'' \odot e_o'),$$

where $\odot$ is the element-wise (Hadamard) product.

We optimized the negative log-likelihood of the logistic model described in Equation (5) with L^2 regularization on the parameters Θ:

$$\gamma(\Omega; \Theta) \quad = \sum_{((r,s,o),y)\in\Omega} \log(1 + \exp(-y\phi(r, s, o; \Theta))) + \lambda||\Theta||_2^2 \qquad (10)$$

where $\lambda \in \mathbb{R}_+$ is the regularization parameter.

To handle regularization, note that using separate representations for the real and imaginary parts does not change anything as the squared L^2-norm of a complex vector $v = v' + iv''$ is the sum of the squared modulus of each entry:

$$||v||_2^2 \quad = \quad \sum_j \sqrt{v_j'^2 + v_j''^2}^2$$
$$= \quad \sum_j v_j'^2 + \sum_j v_j''^2$$
$$= \quad ||v'||_2^2 + ||v''||_2^2,$$

which is actually the sum of the L^2-norms of the vectors of the real and imaginary parts.

We can finally write the gradient of γ with respect to a *real* embedding v for one triple (r, s, o) and its truth value y:

$$\nabla_v\gamma(\{((r, s, o), y)\}; \Theta) \quad = \quad -y\sigma(-y\phi(r, s, o; \Theta))\nabla_v\phi(r, s, o; \Theta) + 2\lambda v. \qquad (11)$$

16

Algorithm 1 Stochastic gradient descent with AdaGrad for the COMPLEX model

---

Input Training set Ω, validation set Ω_v, learning rate $\alpha \in \mathbb{R}_{++}$, rank $K \in \mathbb{Z}_{++}$, L^2
    regularization factor $\lambda \in \mathbb{R}_+$, negative ratio $\eta \in \mathbb{Z}_{++}$, batch size $b \in \mathbb{Z}_{++}$, maximum
    iteration $m \in \mathbb{Z}_{++}$, validate every $s \in \mathbb{Z}_{++}$ iterations, AdaGrad regularizer $\epsilon = 10^{-8}$.
Output Embeddings e', e'', w', w''.
    $e'_i \sim \mathcal{N}(\mathbf{0}^k, I^{k \times k})$, $e''_i \sim \mathcal{N}(\mathbf{0}^k, I^{k \times k})$ for each $i \in \mathcal{E}$
    $w'_i \sim \mathcal{N}(\mathbf{0}^k, I^{k \times k})$, $w''_i \sim \mathcal{N}(\mathbf{0}^k, I^{k \times k})$ for each $i \in \mathcal{R}$
    $g_{e'_i} \leftarrow \mathbf{0}^k$, $g_{e''_i} \leftarrow \mathbf{0}^k$ for each $i \in \mathcal{E}$
    $g_{w'_i} \leftarrow \mathbf{0}^k$, $g_{w''_i} \leftarrow \mathbf{0}^k$ for each $i \in \mathcal{R}$
    $previous\_score \leftarrow 0$
    for $i = 1, \ldots, m$ **do**
        for $j = 1, \ldots, |\Omega|/b$ **do**
            $\Omega_b \leftarrow sample\_batch\_of\_size\_b(\Omega, b)$
            // Negative sampling:
            $\Omega_n \leftarrow \{\emptyset\}$
            for $((r, s, o), y)$ in Ω_b **do**
                for $l = 1, \ldots, \eta$ **do**
                    $e \leftarrow one\_random\_sample(\mathcal{E})$
                    if $\text{Bern}(0.5) > 0.5$ **then**
                        $\Omega_n \leftarrow \Omega_n \cup \{((r, e, o), -1)\}$
                    else
                        $\Omega_n \leftarrow \Omega_n \cup \{((r, s, e), -1)\}$
                    end if
                end for
            end for
            $\Omega_b \leftarrow \Omega_b \cup \Omega_n$
            for $((r, s, o), y)$ in Ω_b **do**
                for v in Θ **do**
                    // AdaGrad updates:
                    $g_v \leftarrow g_v + (\nabla_v \gamma(\{((r, s, o), y)\}; \Theta))^2$
                    // Gradient updates:
                    $v \leftarrow v - \frac{\alpha}{g_v + \epsilon} \nabla_v \gamma(\{((r, s, o), y)\}; \Theta)$
                end for
            end for
        end for
        // Early stopping
        if $i \mod s = 0$ **then**
            $current\_score \leftarrow evaluate\_AP\_or\_MRR(\Omega_v; \Theta)$
            if $current\_score \leq previous\_score$ **then**
                break
            end if
            $previous\_score \leftarrow current\_score$
        end if
    end for
    return Θ

---

5. Experiments

We evaluated the method proposed in this paper on both synthetic and real data sets. The synthetic data set contains both symmetric and antisymmetric relations, whereas the real data sets are standard link prediction benchmarks based on real knowledge graphs.

We compared COMPLEX to state-of-the-art models, namely TRANSE (Bordes et al., 2013b), DISTMULT (Yang et al., 2015), RESCAL (Nickel et al., 2011) and also to the canonical polyadic decomposition (CP) (Hitchcock, 1927), to emphasize empirically the importance of learning unique embeddings for entities. For experimental fairness, we reimplemented these models within the same framework as the COMPLEX model, using a Theano-based SGD implementation[3] (Bergstra et al., 2010).

For the TRANSE model, results were obtained with its original max-margin loss, as it turned out to yield better results for this model only. To use this max-margin loss on data sets with observed negatives (Sections 5.1 and 5.2), positive triples were replicated when necessary to match the number of negative triples, as described in Garcia-Duran et al. (2016). All other models are trained with the negative log-likelihood of the logistic model (Equation (10)). In all the following experiments we used a maximum number of iterations $m = 1000$, a batch size $b = \frac{|\Omega|}{100}$, and validated the models for early stopping every $s = 50$ iterations.

5.1 Synthetic Task

To assess our claim that COMPLEX can accurately model jointly symmetry and antisymmetry, we randomly generated a knowledge graph of two relations and 30 entities. One relation is entirely symmetric, while the other is completely antisymmetric. This data set corresponds to a $2 \times 30 \times 30$ tensor. Figure 2 shows a part of this randomly generated tensor, with a symmetric slice and an antisymmetric slice, decomposed into training, validation and test sets. To ensure that all test values are predictable, the upper triangular parts of the matrices are always kept in the training set, and the diagonals are unobserved. We conducted a 5-fold cross-validation on the lower-triangular matrices, using the upper-triangular parts plus 3 folds for training, one fold for validation and one fold for testing. Each training set contains 1392 observed triples, whereas validation and test sets contain 174 triples each.

Figure 3 shows the best cross-validated average precision (area under the precision-recall curve) for different factorization models of ranks ranging up to 50. The regularization parameter λ is validated in {0.1, 0.03, 0.01, 0.003,0.001, 0.0003, 0.00001, 0.0} and the learning rate α was initialized to 0.5.

As expected, DISTMULT (Yang et al., 2015) is not able to model antisymmetry and only predicts the symmetric relations correctly. Although TRANSE (Bordes et al., 2013b) is not a symmetric model, it performs poorly in practice, particularly on the antisymmetric relation. RESCAL (Nickel et al., 2011), with its large number of parameters, quickly overfits as the rank grows. Canonical Polyadic (CP) decomposition (Hitchcock, 1927) fails on both relations as it has to push symmetric and antisymmetric patterns through the entity embeddings. Surprisingly, only COMPLEX succeeds even on such simple data.

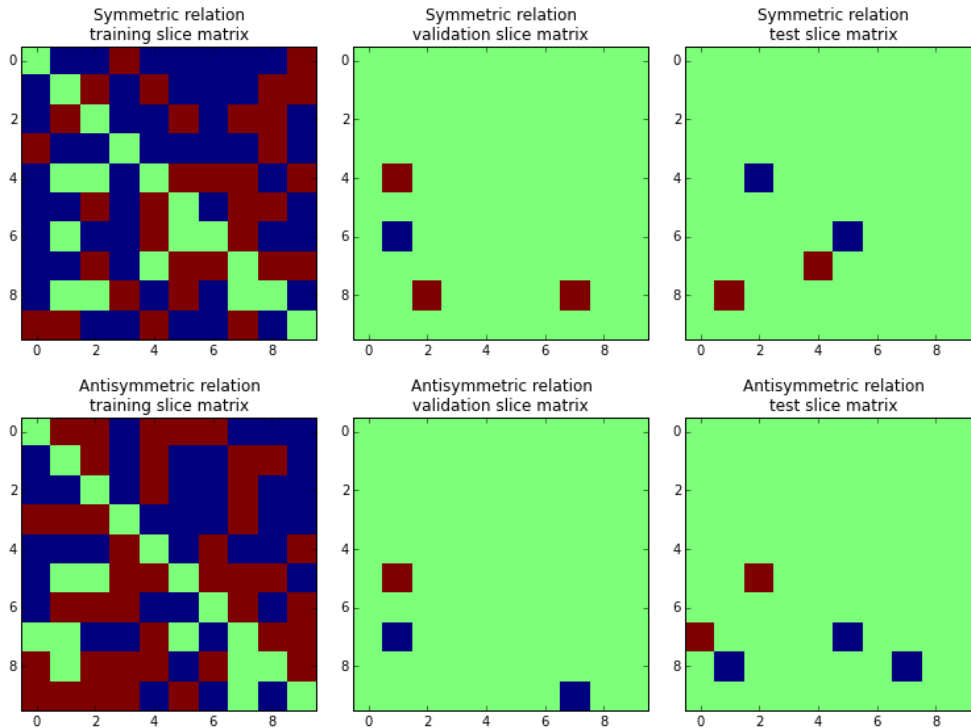
3. https://github.com/lmjohns3/downhill

Figure 2: Parts of the training, validation and test sets of the generated experiment with one symmetric and one antisymmetric relation. Red pixels are positive triples, blue are negatives, and green missing ones. Top: Plots of the symmetric slice (relation) for the 10 first entities. Bottom: Plots of the antisymmetric slice for the 10 first entities.

5.2 Real Fully-Observed Data Sets: Kinships and UMLS

We then compare all models on two fully observed data sets, that contain both positive and negative triples, also called the *closed-world assumption*. The Kinships data set (Denham, 1973) describes the 26 different kinship relations of the Alyawarra tribe in Australia, among 104 individuals. The unified medical language system (UMLS) data set (McCray, 2003) represents 135 medical concepts and diseases, linked by 49 relations describing their interactions. Metadata for the two data sets is summarized in Table 2.

Data set	$|\mathcal{E}|$	$|\mathcal{R}|$	Total number of triples
Kinships	104	26	281,216
UMLS	135	49	893,025

Table 2: Number of entities $|\mathcal{E}|$, relations $|\mathcal{R}|$, and observed triples (all are observed) for the Kinships and UMLS data sets.
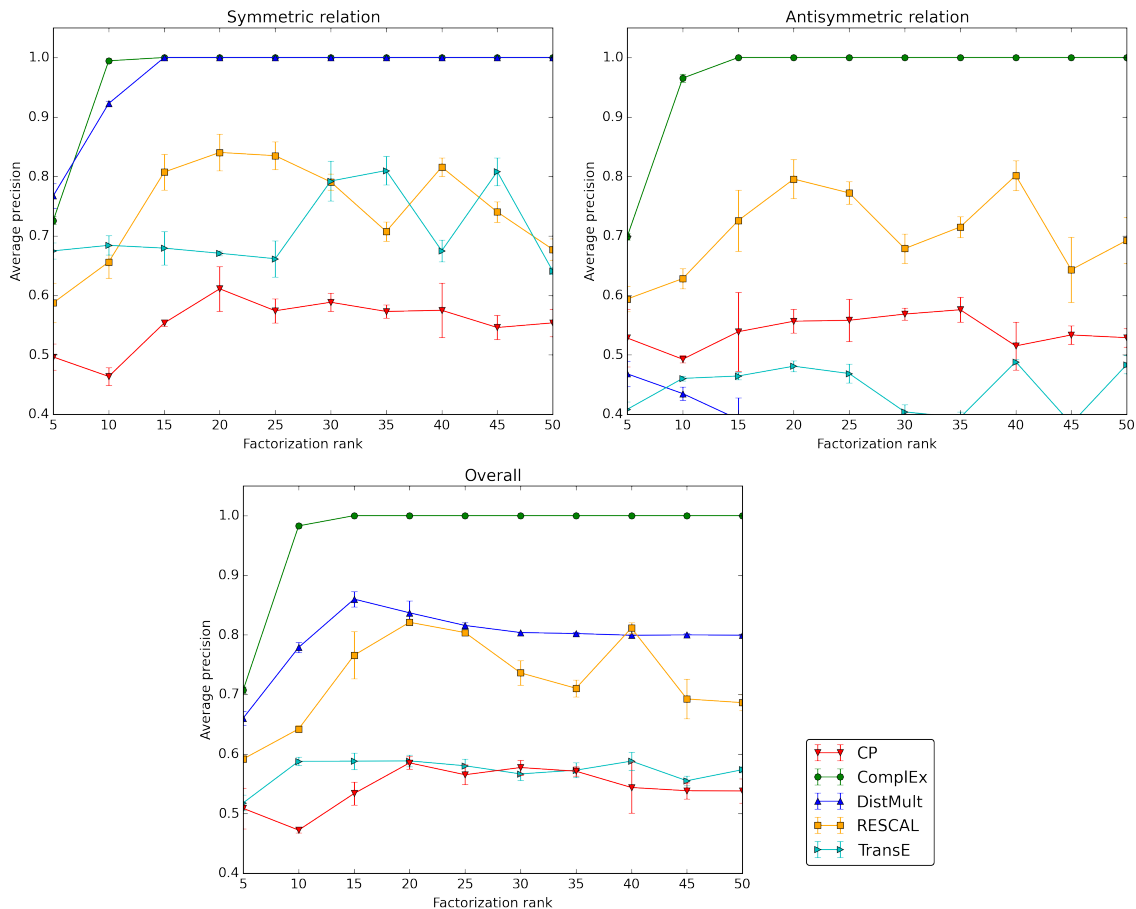
Figure 3: Average precision (AP) for each factorization rank ranging from 5 to 50 for different state-of-the-art models on the synthetic task. Learning is performed jointly on the symmetric relation and on the antisymmetric relation. Top-left: AP over the symmetric relation only. Top-right: AP over the antisymmetric relation only. Bottom: Overall AP.

We performed a 10-fold cross-validation, keeping 8 for training, one for validation and one for testing. Figure 4 shows the best cross-validated average precision for ranks ranging up to 50, and error bars show the standard deviation over the 10 runs. The regularization parameter λ is validated in {0.1, 0.03, 0.01, 0.003, 0.001, 0.0003, 0.00001, 0.0} and the learning rate α was initialized to 0.5.

On both data sets CompLEx, RESCAL and CP are very close, with a slight advantage for CompLEx on Kinships, and for RESCAL on UMLS. DistMult performs poorly here as many relations are antisymmetric both in UMLS (causal links, anatomical hierarchies) and Kinships (being father, uncle or grand-father).

The fact that CP, RESCAL and CompLEx work so well on these data sets illustrates the importance of having an expressive enough model, as DistMult fails because
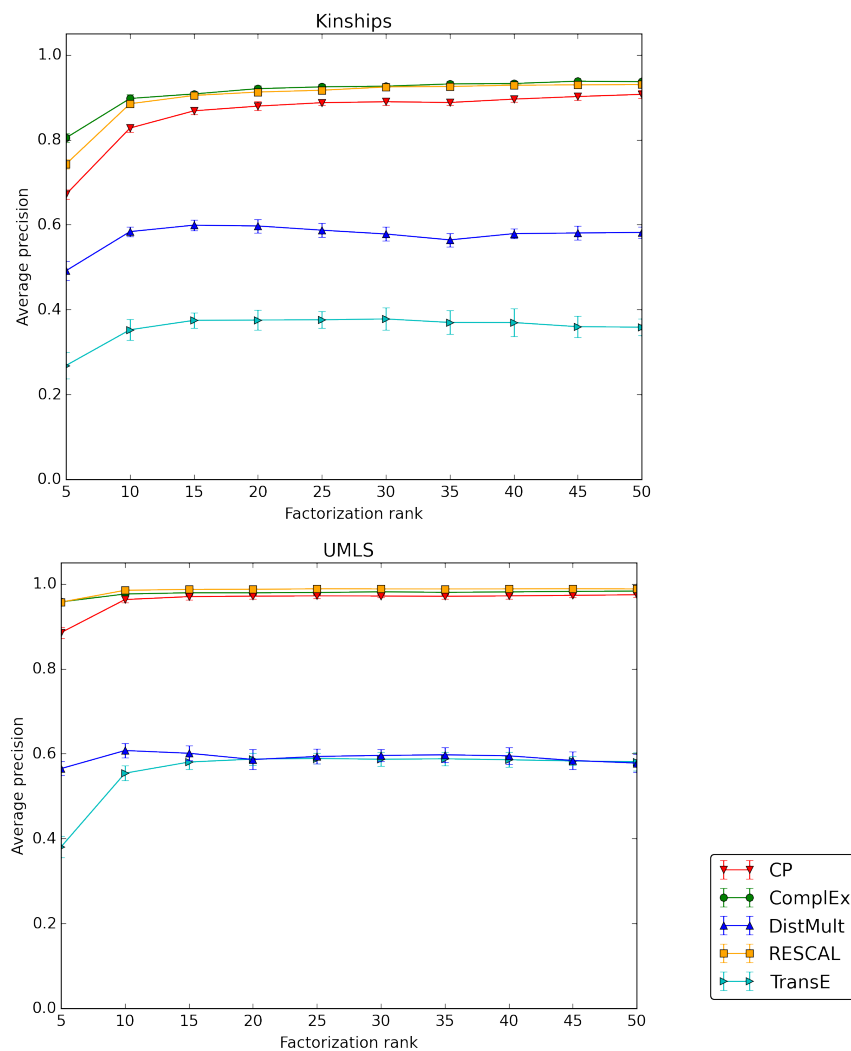
Figure 4: Average precision (AP) for each factorization rank ranging from 5 to 50 for different state-of-the-art models on the Kinships data set (top) and on the UMLS data set (bottom).

of antisymmetry; the power of the multilinear product—that is the tensor factorization approach—as TRANSE can be seen as a sum of bilinear products (Garcia-Duran et al., 2016); but not yet the importance of having unique entity embeddings, as CP works well. We believe having separate subject and object-entity embeddings works well under the closed-world assumption, because of the amount of training data compared to the number of embeddings to learn. Though when only a fractions of the positive training examples are observed (as it is most often the case), we will see in the next experiments that enforcing unique entity embeddings is key to good generalization.

Data set	$|\mathcal{E}|$	$|\mathcal{R}|$	Number of triples in sets:		
			Training	Validation	Test
WN18	40,943	18	141,442	5,000	5,000
FB15K	14,951	1,345	483,142	50,000	59,071

Table 3: Number of entities $|\mathcal{E}|$, relations $|\mathcal{R}|$, and observed triples in each split for the FB15K and WN18 data sets.

5.3 Real Sparse Data Sets: FB15K and WN18

Finally, we evaluated COMPLEX on the FB15K and WN18 data sets, as they are well established benchmarks for the link prediction task. FB15K is a subset of Freebase (Bollacker et al., 2008), a curated knowledge graph of general facts, whereas WN18 is a subset of WordNet (Fellbaum, 1998), a database featuring lexical relations between words. We used the same training, validation and test set splits as in Bordes et al. (2013b). Table 3 summarizes the metadata of the two data sets.

5.3.1 EXPERIMENTAL SETUP

As both data sets contain only positive triples, we generated negative samples using the local closed-world assumption, as described in Section 4. For evaluation, we measure the quality of the ranking of each test triple among all possible subject and object substitutions : $r(s', o)$ and $r(s, o')$, for each s', o' in $\mathcal{E}$, as used in previous studies (Bordes et al., 2013b; Nickel et al., 2016b). Mean Reciprocal Rank (MRR) and Hits at N are standard evaluation measures for these data sets and come in two flavours: raw and filtered. The filtered metrics are computed *after* removing all the other positive observed triples that appear in either training, validation or test set from the ranking, whereas the raw metrics do not remove these.

Since ranking measures are used, previous studies generally preferred a max-margin ranking loss for the task (Bordes et al., 2013b; Nickel et al., 2016b). We chose to use the negative log-likelihood of the logistic model—as described in the previous section—as it is a continuous surrogate of the sign-rank, and has been shown to learn compact representations for several important relations, especially for transitive relations (Bouchard et al., 2015). As previously stated, we tried both losses in preliminary work, and indeed training the models with the log-likelihood yielded better results than with the max-margin ranking loss, especially on FB15K—except with TRANSE.

We report both filtered and raw MRR, and filtered Hits at 1, 3 and 10 in Table 4 for the evaluated models. The HOLE model has recently been shown to be equivalent to COMPLEX (Hayashi and Shimbo, 2017), we record the original results for HOLE as reported in Nickel et al. (2016b) and briefly discuss the discrepancy of results obtained with COMPLEX.

Reported results are given for the best set of hyper-parameters evaluated on the validation set for each model, after a distributed grid-search on the following values: $K \in \{10, 20, 50, 100, 150, 200\}$, $\lambda \in \{0.1, 0.03, 0.01, 0.003, 0.001, 0.0003, 0.0\}$, $\alpha \in \{1.0, 0.5, 0.2, 0.1, 0.05, 0.02, 0.01\}$, $\eta \in \{1, 2, 5, 10\}$ with λ the L^2 regularization parameter, α the initial learning rate, and η the number of negatives generated per positive training triple. We also

| Model | WN18 | | | | | FB15K | | | | |
| | MRR | | Hits at | | | MRR | | Hits at | | |
	Filtered	Raw	1	3	10	Filtered	Raw	1	3	10
CP	0.075	0.058	0.049	0.080	0.125	0.326	0.152	0.219	0.376	0.532
TransE	0.454	0.335	0.089	0.823	0.934	0.380	0.221	0.231	0.472	0.641
RESCAL	0.894	0.583	0.867	0.918	0.935	0.461	0.226	0.324	0.536	0.720
DistMult	0.822	0.532	0.728	0.914	0.936	0.654	**0.242**	0.546	0.733	0.824
HolE*	0.938	**0.616**	0.930	**0.945**	**0.949**	0.524	0.232	0.402	0.613	0.739
ComplEx	**0.941**	0.587	**0.936**	**0.945**	0.947	**0.692**	**0.242**	**0.599**	**0.759**	**0.840**

Table 4: Filtered and raw mean reciprocal rank (MRR) for the models tested on the FB15K and WN18 data sets. Hits@N metrics are filtered. *Results reported from Nickel et al. (2016b) for the HolE model, that has been shown to be equivalent to ComplEx (Hayashi and Shimbo, 2017), score divergence on FB15K is only due to the loss function used (Trouillon and Nickel, 2017).

tried varying the batch size but this had no impact and we settled with 100 batches per epoch. With the best hyper-parameters, training the ComplEx model on a single GPU (NVIDIA Tesla P40) takes 45 minutes on WN18 ($K = 150, \eta = 1$), and three hours on FB15K ($K = 200, \eta = 10$).

5.3.2 Results

WN18 describes lexical and semantic hierarchies between concepts and contains many antisymmetric relations such as hypernymy, hyponymy, and being part of. Indeed, the DistMult and TransE models are outperformed here by ComplEx and HolE, which are on a par with respective filtered MRR scores of 0.941 and 0.938, which is expected as both models are equivalent.

Table 5 shows the filtered MRR for the reimplemented models and each relation of WN18, confirming the advantage of ComplEx on antisymmetric relations while losing nothing on the others. 2D projections of the relation embeddings (Figures 8 & 9) visually corroborate the results.

On FB15K, the gap is much more pronounced and the ComplEx model largely outperforms HolE, with a filtered MRR of 0.692 and 59.9% of Hits at 1, compared to 0.524 and 40.2% for HolE. This difference of scores between the two models, though they have been proved to be equivalent (Hayashi and Shimbo, 2017), is due to the use of the aforementioned max-margin loss in the original HolE publication (Nickel et al., 2016b) that performs worse than the log-likelihood on this dataset, and to the generation of more than one negative sample per positive in these experiments. This has been confirmed and discussed in details by Trouillon and Nickel (2017). The fact that DistMult yields fairly high scores (0.654 filtered MRR) is also due to the task itself and the evaluation measures used. As the dataset only involves true facts, the test set never includes the opposite facts $r(o, s)$ of each test fact $r(s, o)$ for *antisymmetric* relations—as the opposite fact is always false. Thus highly scoring the opposite fact barely impacts the rankings for antisymmetric relations. This is not the case in the fully observed experiments (Section 5.2), as the opposite fact is known

Relation name	CompLEx	RESCAL	DistMult	TransE	CP
hypernym	**0.953**	0.935	0.791	0.446	0.109
hyponym	**0.946**	0.932	0.710	0.361	0.009
member_meronym	**0.921**	0.851	0.704	0.418	0.019
member_holonym	**0.946**	0.861	0.740	0.465	0.134
instance_hypernym	**0.965**	0.833	0.943	0.961	0.233
instance_hyponym	**0.945**	0.849	0.940	0.745	0.040
has_part	**0.933**	0.879	0.753	0.426	0.035
part_of	**0.940**	0.888	0.867	0.455	0.094
member_of_domain_topic	**0.924**	0.865	0.914	0.861	0.007
synset_domain_topic_of	**0.930**	0.855	0.919	0.917	0.153
member_of_domain_usage	**0.917**	0.629	**0.917**	0.875	0.001
synset_domain_usage_of	**1.000**	0.541	**1.000**	**1.000**	0.134
member_of_domain_region	**0.865**	0.632	0.635	**0.865**	0.001
synset_domain_region_of	0.919	0.655	0.888	**0.986**	0.149
derivationally_related_form	**0.946**	0.928	0.940	0.384	0.100
similar_to	**1.000**	0.001	**1.000**	0.244	0.000
verb_group	**0.936**	0.857	0.897	0.323	0.035
also_see	0.603	0.302	**0.607**	0.279	0.020

Table 5: Filtered Mean Reciprocal Rank (MRR) for the models tested on each relation of the WordNet data set (WN18).

to be false—for antisymmetric relations—and largely impacts the average precision of the DistMult model (Figure 4).

RESCAL, that represents each relation with a $K \times K$ matrix, performs well on WN18 as there are few relations and hence not so many parameters. On FB15K though, it probably overfits due to the large number of relations and thus the large number of parameters to learn, and performs worse than a less expressive model like DistMult. On both data sets, TransE and CP are largely left behind. This illustrates again the power of the multilinear product in the first case, and the importance of learning unique entity embeddings in the second. CP performs especially poorly on WN18 due to the small number of relations, which magnifies this subject/object difference.

Figure 5 shows that the filtered MRR of the CompLEx model quickly converges on both data sets, showing that the low-rank hypothesis is reasonable in practice. The little gain of performances for ranks comprised between 50 and 200 also shows that CompLEx does not perform better because it has twice as many parameters for the same rank—the real and imaginary parts—compared to other linear space complexity models but indeed thanks to its better expressiveness.

Best ranks were generally 150 or 200, in both cases scores were always very close for all models, suggesting there was no need to grid-search on higher ranks. The number of negative samples per positive sample also had a large influence on the filtered MRR on FB15K (up to +0.08 improvement from 1 to 10 negatives), but not much on WN18. On both data sets regularization was important (up to +0.05 on filtered MRR between $\lambda = 0$ and optimal one). We found the initial learning rate to be very important on FB15K, while not so
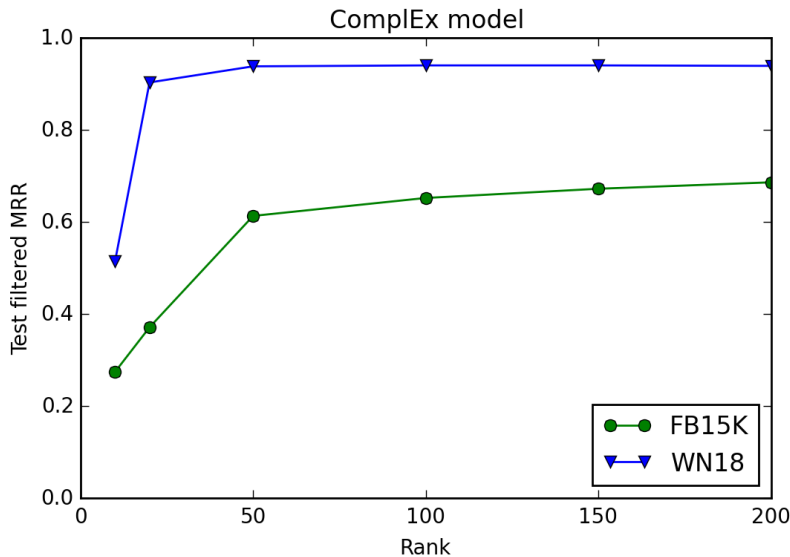
Figure 5: Best filtered MRR for ComplEx on the FB15K and WN18 data sets for different ranks. Increasing the rank gives little performance gain for ranks of $50 - 200$.

much on WN18. We think this may also explain the large gap of improvement ComplEx provides on this data set compared to previously published results—as DistMult results are also better than those previously reported (Yang et al., 2015)—along with the use of the log-likelihood objective. It seems that in general AdaGrad is relatively insensitive to the initial learning rate, perhaps causing some overconfidence in its ability to tune the step size online and consequently leading to less efforts when selecting the initial step size.

5.4 Training time

As defended in Section 2, having a linear time and space complexity becomes critical when the dataset grows. To illustrate this, we report in Figure 6 the evolution of the filtered MRR on the validation set as a function of time, for the best set of validated hyper-parameters for each model. The convergence criterion used is the decrease of the validation filtered MRR—computed every 50 iterations—with a maximum number of iterations of 1000 (see Algorithm 1). All models have a linear complexity except for RESCAL that has a quadratic one in the rank of the decomposition, as it learns one matrix embedding for each relation $r \in \mathcal{R}$. Timings are measured on a single NVIDIA Tesla P40 GPU.

On WN18, all models reach convergence in a reasonable time, between 15 minutes and 1 hour and 20 minutes. The difference between RESCAL and the other models is not sharp there, first because its optimal embedding size ($K = 50$) is lower compared to the other models. Secondly, there are only $|\mathcal{R}| = 18$ relations in WN18, hence the memory footprint of RESCAL is pretty similar to the other models—because it represents only relations with matrices and not entities. On FB15K, the difference is much more pronounced, as RESCAL optimal rank is similar to the other models; and with $|\mathcal{R}| = 1345$ relations, RESCAL has
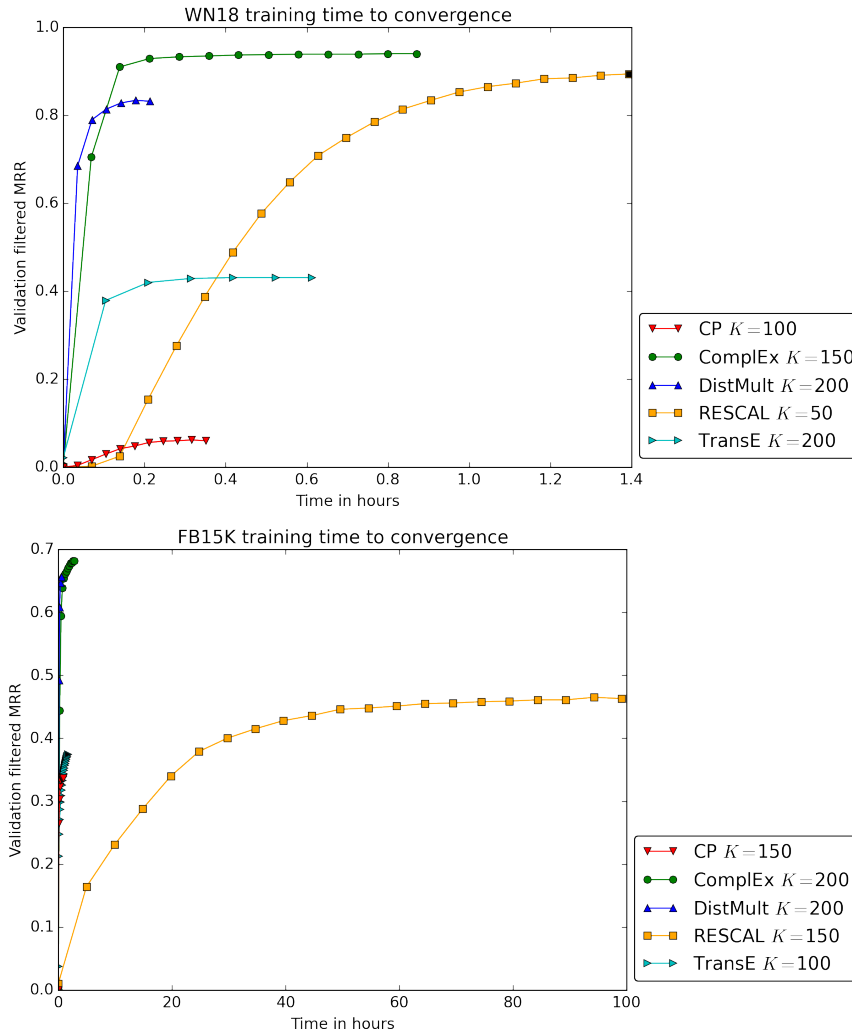
Figure 6: Evolution of the filtered MRR on the validation set as a function of time, on WN18 (top) and FB15K (bottom) for each model for its best set of hyper-parameters. The best rank K is reported in legend. Final black marker indicates that the maximum number of iterations (1000) has been reached (RESCAL on WN18, TRANSE on FB15K).

a much higher memory footprint, which implies more processor cache misses due to the uniformly-random nature of the SGD sampling.

RESCAL took more than four days to train on FB15K, whereas other models took between 40 minutes and 3 hours. While a few days might seem manageable, this could not be the case on larger data sets, as FB15K is but a small subset of Freebase that contains $|\mathcal{R}| = 35000$ relations (Bollacker et al., 2008). This experimentally supports our claim that linear complexity is required for scalability.
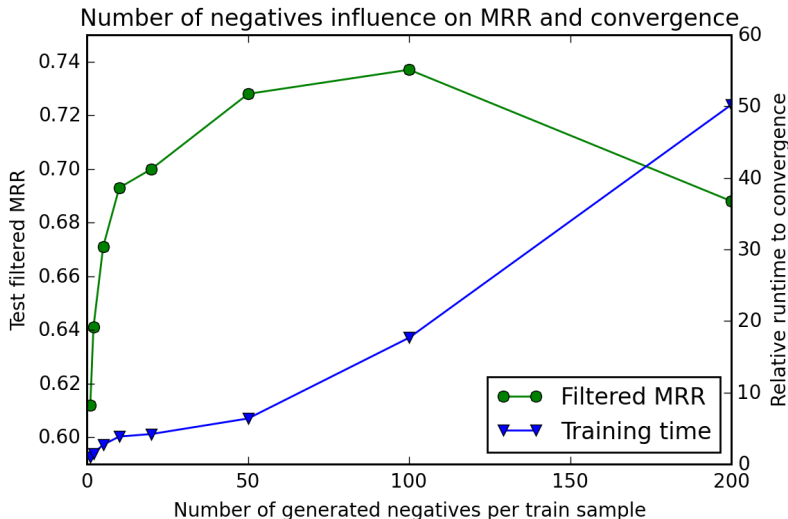
Figure 7: Influence of the number of negative triples generated per positive training example on the filtered test MRR and on training time to convergence on FB15K for the ComplEx model with $K = 200$, $\lambda = 0.01$ and $\alpha = 0.5$. Times are given relative to the training time with one negative triple generated per positive training sample ($= 1$ on time scale).

5.4.1 Influence of Negative Samples

We further investigated the influence of the number of negatives generated per positive training sample. In the previous experiment, due to computational limitations, the number of negatives per training sample, η, was validated over the set $\{1, 2, 5, 10\}$. On WN18 it proved to be of no help to have more than one generated negative per positive. Here we explore in which proportions increasing the number of generated negatives leads to better results on FB15K. To do so, we fixed the best validated λ, K, α obtained from the previous experiment. We then let η vary in $\{1, 2, 5, 10, 20, 50, 100, 200\}$.

Figure 7 shows the influence of the number of generated negatives per positive training triple on the performance of ComplEx on FB15K. Generating more negatives clearly improves the results up to 100 negative triples, with a filtered MRR of 0.737 and 64.8% of Hits@1, before decreasing again with 200 negatives, probably due to the too large class imbalance. The model also converges with fewer epochs, which compensates partially for the additional training time per epoch, up to 50 negatives. It then grows linearly as the number of negatives increases.

5.4.2 WN18 Embeddings Visualization

We used principal component analysis (PCA) to visualize embeddings of the relations of the WordNet data set (WN18). We plotted the four first components of the best DistMult and ComplEx model's embeddings in Figures 8 & 9. For the ComplEx model, we simply concatenated the real and imaginary parts of each embedding.
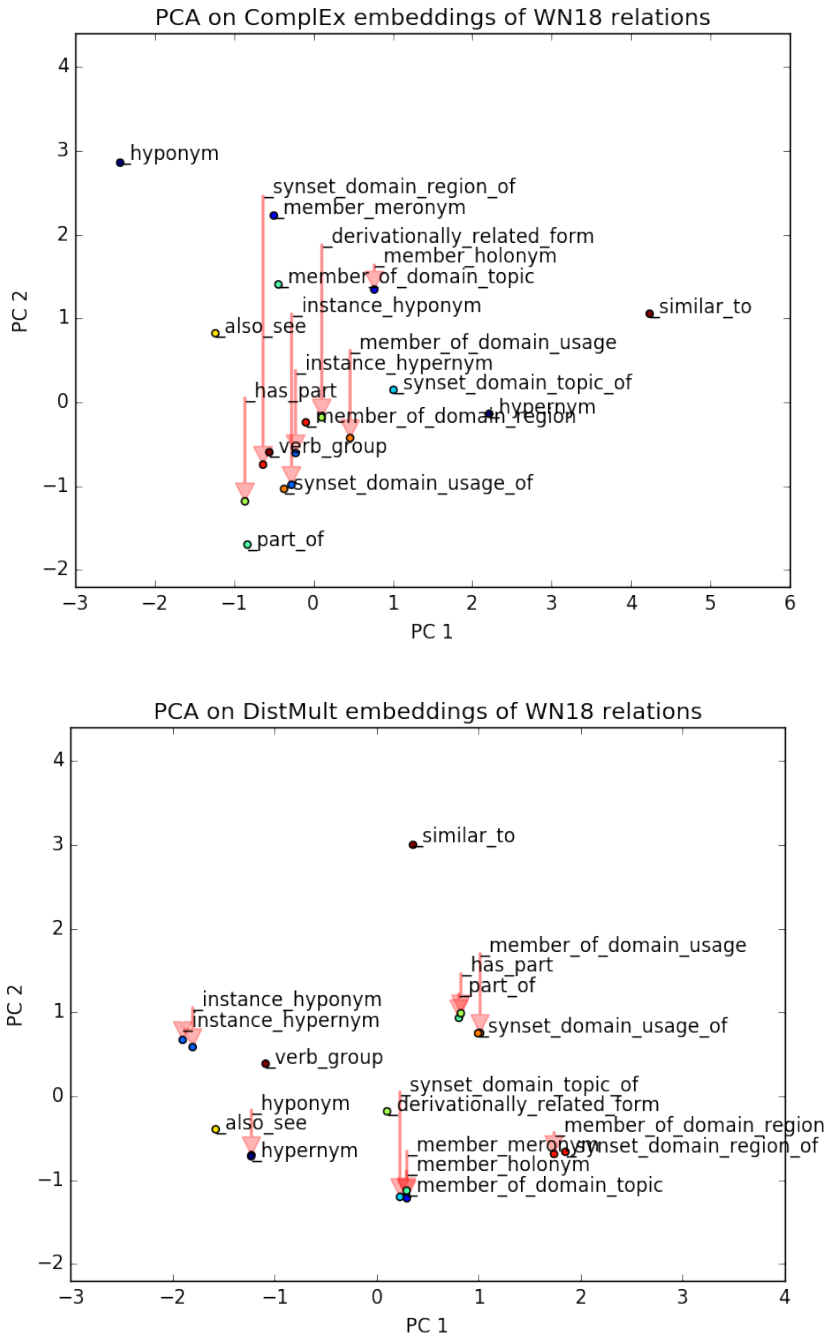
Figure 8: Plots of the first and second components of the WN18 relations embeddings using principal component analysis. Red arrows link the labels to their point. Top: COMPLEX embeddings. Bottom: DISTMULT embeddings. Opposite relations are clustered together by DISTMULT while correctly separated by COMPLEX.
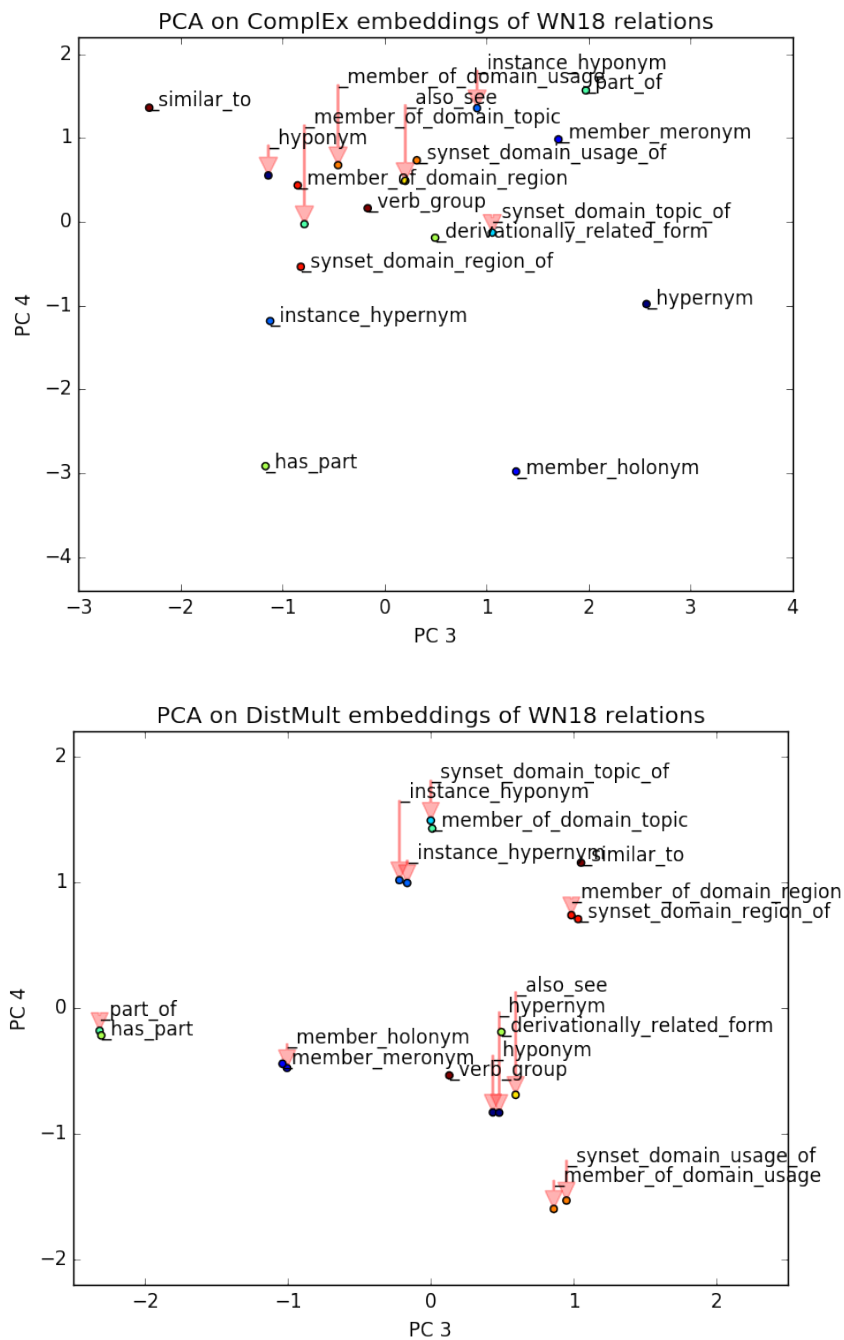
Figure 9: Plots of the third and fourth components of the WN18 relations embeddings using principal component analysis. Red arrows link the labels to their point. Top: ComplEx embeddings. Bottom: DistMult embeddings. Opposite relations are clustered together by DistMult while correctly separated by ComplEx.

Most of WN18 relations describe hierarchies, and are thus antisymmetric. Each of these hierarchic relations has its inverse relation in the data set. For example: `hypernym` / `hyponym`, `part_of` / `has_part`, `synset_domain_topic_of` / `member_of_domain_topic`. Since DistMult is unable to model antisymmetry, it will correctly represent the nature of each pair of opposite relations, but not the direction of the relations. Loosely speaking, in the `hypernym` / `hyponym` pair the nature is sharing semantics, and the direction is that one entity generalizes the semantics of the other. This makes DistMult representing the opposite relations with very close embeddings. It is especially striking for the third and fourth principal component (Figure 9). Conversely, CompleEx manages to oppose spatially the opposite relations.

6. Related Work

We first discuss related work about complex-valued matrix and tensor decompositions, and then review other approaches for knowledge graph completion.

6.1 Complex Numbers

When factorization methods are applied, the representation of the decomposition is generally chosen in accordance with the data, despite the fact that most real square matrices only have eigenvalues in the complex domain. Indeed in the machine learning community, the data is usually real-valued, and thus eigendecomposition is used for symmetric matrices, or other decompositions such as (real-valued) singular value decomposition (Beltrami, 1873), non-negative matrix factorization (Paatero and Tapper, 1994), or canonical polyadic decomposition when it comes to tensors (Hitchcock, 1927).

Conversely, in signal processing, data is often complex-valued (Stoica and Moses, 2005) and the complex-valued counterparts of these decompositions are then used. Joint diagonalization is also a much more common tool than in machine learning for decomposing sets of (complex) dense square matrices (Belouchrani et al., 1997; De Lathauwer et al., 2001).

Some works on recommender systems use complex numbers as an encoding facility, to merge two real-valued relations, similarity and liking, into one single complex-valued matrix which is then decomposed with complex embeddings (Kunegis et al., 2012; Xie et al., 2015). Still, unlike our work, it is not real data that is decomposed in the complex domain.

In deep learning, Danihelka et al. (2016) proposed an LSTM extended with an associative memory based on complex-valued vectors for memorization tasks, and Hu et al. (2016) a complex-valued neural network for speech synthesis. In both cases again, the data is first encoded in complex vectors that are then fed into the network.

Conversely to these contributions, this work suggests that processing real-valued data with complex-valued representation, through a projection onto the real-valued subspace, can be a very simple way of increasing the expressiveness of the model considered.

6.2 Knowledge Graph Completion

Many knowledge graphs have recently arisen, pushed by the W3C recommendation to use the resource description framework (RDF) (Cyganiak et al., 2014) for data representation. Examples of such knowledge graphs include DBPedia (Auer et al., 2007), Freebase (Bollacker

et al., 2008) and the Google Knowledge Vault (Dong et al., 2014). Motivating applications of knowledge graph completion include question answering (Bordes et al., 2014b) and more generally probabilistic querying of knowledge bases (Huang and Liu, 2009; Krompaß et al., 2014).

First approaches to relational learning relied upon probabilistic graphical models (Getoor and Taskar, 2007), such as bayesian networks (Friedman et al., 1999) and markov logic networks (Richardson and Domingos, 2006; Raedt et al., 2016).

With the first embedding models, asymmetry of relations was quickly seen as a problem and asymmetric extensions of tensors were studied, mostly by either considering independent embeddings (Franz et al., 2009) or considering relations as matrices instead of vectors in the RESCAL model (Nickel et al., 2011), or both (Sutskever, 2009). Direct extensions were based on uni-,bi- and trigram latent factors for triple data (Garcia-Duran et al., 2016), as well as a low-rank relation matrix (Jenatton et al., 2012). Bordes et al. (2014a) propose a two-layer model where subject and object embeddings are first separately combined with the relation embedding, then each intermediate representation is combined into the final score.

Pairwise interaction models were also considered to improve prediction performances. For example, the Universal Schema approach (Riedel et al., 2013) factorizes a 2D unfolding of the tensor (a matrix of entity pairs vs. relations) while Welbl et al. (2016) extend this also to other pairs. Riedel et al. (2013) also consider augmenting the knowledge graph facts by exctracting them from textual data, as does Toutanova et al. (2015). Injecting prior knowledge in the form of Horn clauses in the objective loss of the Universal Schema model has also been considered (Rocktaschel et al., 2015). Chang et al. (2014) enhance the RESCAL model to take into account information about the entity types. For recommender systems (thus with different subject/object sets of entities), Baruch (2014) proposed a non-commutative extension of the CP decomposition model. More recently, Gaifman models that learn neighborhood embeddings of local structures in the knowledge graph showed competitive performances (Niepert, 2016).

In the Neural Tensor Network (NTN) model, Socher et al. (2013) combine linear transformations and multiple bilinear forms of subject and object embeddings to jointly feed them into a nonlinear neural layer. Its non-linearity and multiple ways of including interactions between embeddings gives it an advantage in expressiveness over models with simpler scoring function like DistMult or RESCAL. As a downside, its very large number of parameters can make the NTN model harder to train and overfit more easily.

The original multilinear DistMult model is symmetric in subject and object for every relation (Yang et al., 2015) and achieves good performance on FB15K and WN18 data sets. However it is likely due to the absence of true negatives in these data sets, as discussed in Section 5.3.2.

The TransE model from Bordes et al. (2013b) also embeds entities and relations in the same space and imposes a geometrical structural bias into the model: the subject entity vector should be close to the object entity vector once translated by the relation vector.

A recent novel way to handle antisymmetry is via the Holographic Embeddings (HolE) model by Nickel et al. (2016b). In HolE the circular correlation is used for combining entity embeddings, measuring the covariance between embeddings at different dimension

shifts. This model has been shown to be equivalent to the CompLEx model (Hayashi and Shimbo, 2017; Trouillon and Nickel, 2017).

7. Discussion and Future Work

Though the decomposition proposed in this paper is clearly not unique, it is able to learn meaningful representations. Still, characterizing all possible unitary diagonalizations that preserve the real part is an interesting open question. Especially in an approximation setting with a constrained rank, in order to characterize the decompositions that minimize a given reconstruction error. That might allow the creation of an iterative algorithm similar to eigendecomposition iterative methods (Saad, 1992) for computing such a decomposition for any given real square matrix.

The proposed decomposition could also find applications in many other asymmetric square matrices decompositions applications, such as spectral graph theory for directed graphs (Cvetković et al., 1997), but also factorization of asymmetric measures matrices such as asymmetric distance matrices (Mao and Saul, 2004) and asymmetric similarity matrices (Pirasteh et al., 2015).

From an optimization point of view, the objective function (Equation (10)) is clearly non-convex, and we could indeed not be reaching a globally optimal decomposition using stochastic gradient descent. Recent results show that there are no spurious local minima in the completion problem of positive semi-definite matrix (Ge et al., 2016; Bhojanapalli et al., 2016). Studying the extensibility of these results to our decomposition is another possible line of future work. The first step would be generalizing these results to symmetric real-valued matrix completion, then generalization to normal matrices should be straight-forward. The two last steps would be extending to matrices that are expressed as real part of normal matrices, and finally to the joint decomposition of such matrices as a tensor. We indeed noticed a remarkable stability of the scores across different random initialization of CompLEx for the same hyper-parameters, which suggests the possibility of such theoretical property.

Practically, an obvious extension is to merge our approach with known extensions to tensor factorization models in order to further improve predictive performance. For example, the use of pairwise embeddings (Riedel et al., 2013; Welbl et al., 2016) together with complex numbers might lead to improved results in many situations that involve non-compositionality. Adding bigram embeddings to the objective could also improve the results as shown on other models (Garcia-Duran et al., 2016). Another direction would be to develop a more intelligent negative sampling procedure, to generate more informative negatives with respect to the positive sample from which they have been sampled. This would reduce the number of negatives required to reach good performance, thus accelerating training time. Extension to relations between more than two entities, n-tuples, is not straightforward, as CompLEx's expressiveness comes from the complex conjugation of the object-entity, that breaks the symmetry between the subject and object embeddings in the scoring function. This stems from the Hermitian product, which seems to have no standard multilinear extension in the linear algebra literature, this question hence remains largely open.

8. Conclusion

We described a new matrix and tensor decomposition with complex-valued latent factors called CompLEx. The decomposition exists for all real square matrices, expressed as the real part of normal matrices. The result extends to sets of real square matrices—tensors— and answers to the requirements of the knowledge graph completion task : handling a large variety of different relations including antisymmetric and asymmetric ones, while being scalable. Experiments confirm its theoretical versatility, as it substantially improves over the state-of-the-art on real knowledge graphs. It shows that real world relations can be efficiently approximated as the real part of low-rank normal matrices. The generality of the theoretical results and the effectiveness of the experimental ones motivate for the application to other real square matrices factorization problems. More generally, we hope that this paper will stimulate the use of complex linear algebra in the machine learning community, even and especially for processing real-valued data.

Acknowledgments

References

Noga Alon, Shay Moran, and Amir Yehudayoff. Sign rank versus vc dimension. In *Conference on Learning Theory*, pages 47–80, 2016.

Sren Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, and Zachary Ives. DBpedia: A nucleus for a web of open data. In *International Semantic Web Conference, Busan, Korea*, pages 11–15. Springer, 2007.

Guy Baruch. A ternary non-commutative latent factor model for scalable three-way real tensor completion. *arXiv preprint arXiv:1410.7383*, 2014.

Adel Belouchrani, Karim Abed-Meraim, J-F Cardoso, and Eric Moulines. A blind source separation technique using second-order statistics. *IEEE Transactions on Signal Processing*, 45(2):434–444, 1997.

Eugenio Beltrami. Sulle funzioni bilineari. *Giornale di Matematiche ad Uso degli Studenti Delle Universita*, 11(2):98–106, 1873.

James Bergstra, Olivier Breuleux, Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, Guillaume Desjardins, Joseph Turian, David Warde-Farley, and Yoshua Bengio. Theano: a CPU and GPU math expression compiler. In *Python for Scientific Computing Conference (SciPy)*, June 2010.

Srinadh Bhojanapalli, Behnam Neyshabur, and Nathan Srebro. Global optimality of local search for low rank matrix recovery. *arXiv preprint arXiv:1605.07221*, 2016.

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

Antoine Bordes, Jason Weston, Ronan Collobert, and Yoshua Bengio. Learning structured embeddings of knowledge bases. In *AAAI Conference on Artificial Intelligence*, 2011.

Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. Irreflexive and hierarchical relations as translations. *Computing Research Repository*, abs/1304.7158, 2013a.

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

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

Antoine Bordes, Jason Weston, and Nicolas Usunier. Open question answering with weakly supervised embedding models. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 165–180. Springer, 2014b.

Guillaume Bouchard, Sameer Singh, and Théo Trouillon. On approximate reasoning capabilities of low-rank vector spaces. *AAAI Spring Symposium on Knowledge Representation and Reasoning: Integrating Symbolic and Neural Approaches*, 2015.

Augustin-Louis Cauchy. Sur l'équation à l'aide de laquelle on détermine les inégalités séculaires des mouvements des planètes. *Œuvres complètes, IIe série*, 9:174–195, 1829.

K. W. Chang, W. T. Yih, B. Yang, and C. Meek. Typed tensor decomposition of knowledge bases for relation extraction. In *Conference on Empirical Methods on Natural Language Processing*, 2014.

Dragoš M. Cvetković, Peter Rowlinson, and Slobodan Simic. *Eigenspaces of graphs*. Number 66. Cambridge University Press, 1997.

Richard Cyganiak, David Wood, and Markus Lanthaler. Rdf 1.1 concepts and abstract syntax. *W3C Recommendation*, 2014.

Ivo Danihelka, Greg Wayne, Benigno Uria, Nal Kalchbrenner, and Alex Graves. Associative long short-term memory. *arXiv preprint arXiv:1602.03032*, 2016.

Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. Independent component analysis and (simultaneous) third-order tensor diagonalization. *IEEE Transactions on Signal Processing*, 49(10):2262–2271, 2001.

Woodrow W Denham. *The detection of patterns in Alyawara nonverbal behavior.* PhD thesis, University of Washington, Seattle., 1973.

Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 601–610, 2014.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159, 2011.

Christiane Fellbaum. *WordNet.* Wiley Online Library, 1998.

Thomas Franz, Antje Schultz, Sergej Sizov, and Steffen Staab. Triplerank: Ranking semantic web data by tensor decomposition. In *International Semantic Web Conference*, pages 213–228, 2009.

Nir Friedman, Lise Getoor, Daphne Koller, and Avi Pfeffer. Learning Probabilistic Relational Models. In *International Joint Conference on Artificial Intelligence*, number August, pages 1300–1309, 1999. ISBN 3540422897. doi: 10.1.1.101.3165.

Alberto Garcia-Duran, Antoine Bordes, Nicolas Usunier, and Yves Grandvalet. Combining two and three-way embedding models for link prediction in knowledge bases. *Journal of Artificial Intelligence Research*, 55:715–742, 2016.

Rong Ge, Jason D Lee, and Tengyu Ma. Matrix completion has no spurious local minimum. *arXiv preprint arXiv:1605.07272*, 2016.

Lise Getoor and Ben Taskar. *Introduction to Statistical Relational Learning.* The MIT Press, 2007. ISBN 0262072882.

Katsuhiko Hayashi and Masashi Shimbo. On the equivalence of holographic and complex embeddings for link prediction. *arXiv preprint arXiv:1702.05563*, 2017.

F. L. Hitchcock. The expression of a tensor or a polyadic as a sum of products. *J. Math. Phys*, 6(1):164–189, 1927.

Roger A Horn and Charles R Johnson. *Matrix analysis.* Cambridge University Press, 2012.

Qiong Hu, Junichi Yamagishi, Korin Richmond, Kartick Subramanian, and Yannis Stylianou. Initial investigation of speech synthesis based on complex-valued neural networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5630–5634, 2016.

Hai Huang and Chengfei Liu. Query evaluation on probabilistic rdf databases. In *International Conference on Web Information Systems Engineering*, pages 307–320. Springer, 2009.

Rodolphe Jenatton, Antoine Bordes, Nicolas Le Roux, and Guillaume Obozinski. A Latent Factor Model for Highly Multi-relational Data. In *Advances in Neural Information Processing Systems 25*, pages 3167–3175, 2012.

Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

Denis Krompaß, Maximilian Nickel, and Volker Tresp. Querying factorized probabilistic triple databases. In *International Semantic Web Conference*, pages 114–129, 2014.

Joseph B Kruskal. Rank, decomposition, and uniqueness for 3-way and n-way arrays. In *Multiway data analysis*, pages 7–18. North-Holland Publishing Co., 1989.

Jérôme Kunegis, Gerd Gröner, and Thomas Gottron. Online dating recommender systems: The split-complex number approach. In *ACM RecSys Workshop on Recommender Systems and the Social Web*, pages 37–44. ACM, 2012.

Nati Linial, Shahar Mendelson, Gideon Schechtman, and Adi Shraibman. Complexity measures of sign matrices. *Combinatorica*, 27(4):439–463, 2007.

Yun Mao and Lawrence K Saul. Modeling distances in large-scale networks by matrix factorization. In *ACM SIGCOMM conference on Internet Measurement*, pages 278–287, 2004.

Alexa T McCray. An upper-level ontology for the biomedical domain. *Comparative and Functional Genomics*, 4(1):80–84, 2003.

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

Maximilian Nickel, Xueyan Jiang, and Volker Tresp. Reducing the rank in relational factorization models by including observable patterns. In *Advances in Neural Information Processing Systems*, pages 1179–1187, 2014.

Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104(1):11–33, 2016a.

Maximilian Nickel, Lorenzo Rosasco, and Tomaso A. Poggio. Holographic embeddings of knowledge graphs. In *AAAI Conference on Artificial Intelligence*, pages 1955–1961, 2016b.

Mathias Niepert. Discriminative gaifman models. In *Advances in Neural Information Processing Systems*, pages 3405–3413, 2016.

Pentti Paatero and Unto Tapper. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5(2):111–126, 1994.

Parivash Pirasteh, Dosam Hwang, and Jason J Jung. Exploiting matrix factorization to asymmetric user similarities in recommendation systems. *Knowledge-Based Systems*, 83: 51–57, 2015.

Luc De Raedt, Kristian Kersting, Sriraam Natarajan, and David Poole. Statistical relational artificial intelligence: Logic, probability, and computation. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 10(2):1–189, 2016.

Matthew Richardson and Pedro Domingos. Markov logic networks. *Machine Learning*, 62 (1-2):107–136, 2006.

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

T Rocktaschel, S Singh, and S Riedel. Injecting Logical Background Knowledge into Embeddings for Relation Extraction. In *Conference of the North American Chapter of the Association for Computational Linguistics*, pages 1119–1129, 2015.

Youcef Saad. *Numerical methods for large eigenvalue problems*, volume 158. SIAM, 1992.

Satya S Sahoo, Wolfgang Halb, Sebastian Hellmann, Kingsley Idehen, Ted Thibodeau Jr, Sören Auer, Juan Sequeda, and Ahmed Ezzat. A survey of current approaches for mapping of relational databases to rdf. *W3C RDB2RDF Incubator Group Report*, pages 113–130, 2009.

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

Petre Stoica and Randolph L Moses. *Spectral analysis of signals*, volume 452. Pearson Prentice Hall Upper Saddle River, NJ, 2005.

Ilya Sutskever. Modelling relational data using bayesian clustered tensor factorization. In *Advances in Neural Information Processing Systems*, pages 1–8, 2009.

Kristina Toutanova, Patrick Pantel, and Michael Gamon. Representing Text for Joint Embedding of Text and Knowledge Bases. In *Conference on Empirical Methods on Natural Language Processing*, 2015.

Théo Trouillon and Maximilian Nickel. Complex and holographic embeddings of knowledge graphs: a comparison. *International Workshop on Statistical Relational AI*, 2017.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. Complex embeddings for simple link prediction. In *International Conference on Machine Learning*, volume 48, pages 2071–2080, 2016.

John von Neumann. Zur algebra der funktionaloperationen und der theorie der normalen operatoren. *Mathematische Annalen*, 102:370–427, 1929.

Johannes Welbl, Guillaume Bouchard, and Sebastian Riedel. A factorization machine framework for testing bigram embeddings in knowledge base completion. In *Workshop on Automated Knowledge Base Construction AKBC@NAACL-HLT*, pages 103–107, 2016.

Feng Xie, Zhen Chen, Jiaxing Shang, Xiaoping Feng, and Jun Li. A link prediction approach for item recommendation with complex numbers. *Knowledge-Based Systems*, 81:148–158, 2015.

Bishan Yang, Wen-Tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. Embedding entities and relations for learning and inference in knowledge bases. In *International Conference on Learning Representations*, 2015.