Low-Dimensional Embeddings of Logic

Tim Rocktäschel[§] **Matko Bosnjak**[§] **Sameer Singh**[†] **Sebastian Riedel**[§]
[§]Department of Computer Science, University College London, UK
[†]Computer Science & Engineering, University of Washington, Seattle
{t.rocktaschel,m.bosnjak,s.riedel}@cs.ucl.ac.uk, sameer@cs.washington.edu

Abstract

Many machine reading approaches, from shallow information extraction to deep semantic parsing, map natural language to symbolic representations of meaning. Representations such as first-order logic capture the richness of natural language and support complex reasoning, but often fail in practice due to their reliance on logical background knowledge and the difficulty of scaling up inference. In contrast, low-dimensional embeddings (*i.e.* distributional representations) are efficient and enable generalization, but it is unclear how reasoning with embeddings could support the full power of symbolic representations such as first-order logic. In this proof-of-concept paper we address this by learning embeddings that simulate the behavior of first-order logic.

1 Introduction

Much of the work in machine reading follows an approach that is, at its heart, symbolic: language is transformed, possibly in a probabilistic way, into a symbolic world model such as a relational database or a knowledge base of first-order formulae. For example, a statistical relation extractor reads texts and populates relational tables (Mintz et al., 2009). Likewise, a semantic parser can turn sentences into complex first-order logic statements (Zettlemoyer and Collins, 2005).

Several properties make symbolic representations of knowledge attractive as a target of machine reading. They support a range of well understood symbolic reasoning processes, capture semantic concepts such as determiners, negations and tense, can be interpreted, edited and curated by humans to inject prior knowledge. However, on practical applications fully symbolic approaches have often shown low recall (*e.g.* Bos and Markert, 2005) as they are affected by the limited coverage of ontologies such as WordNet. Moreover, due to their deterministic nature they often cannot cope with noise and uncertainty inherent to real world data, and inference with such representations is difficult to scale up.

Embedding-based approaches address some of the concerns above. Here relational worlds are described using low-dimensional embeddings of entities and relations based on relational evidence in knowledge bases (Bordes et al., 2011) or surface-form relationships mentioned in text (Riedel et al., 2013). To overcome the generalization bottleneck, these approaches learn to embed similar entities and relations as vectors close in distance. Subsequently, unseen facts can be inferred by simple and efficient linear algebra operations (*e.g.* dot products).

The core argument against embeddings is their supposed inability to capture deeper semantics, and more complex patterns of reasoning such as those enabled by first-order logic (Lewis and Steedman, 2013). Here we argue that this does not need to be true. We present an approach that enables us to learn low-dimensional embeddings such that the model behaves *as if* it follows a complex first-order reasoning process—but still operates in terms of simple vector and matrix representations. In this view, machine reading becomes the process of taking (inherently symbolic) knowledge in language and injecting this knowledge into a sub-symbolic distributional world model. For example, one could envision a semantic parser that turns a sentence into a first-order logic statement,

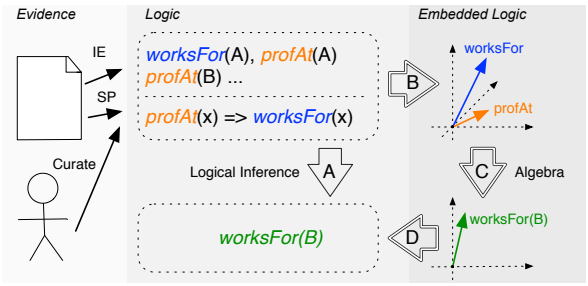

Figure 1: Information extraction (IE) and semantic parsing (SP) extract factual and more general logical statements from text, respectively. Humans can manually curate this knowledge. Instead of reasoning with this knowledge directly (A) we inject it into low dimensional representations of entities and relations (B). Linear algebra operations manipulate embeddings to derive truth vectors (C), which can be discretized or thresholded to retrieve truth values (D).

just to then inject this statement into the embeddings of relations and entities mentioned in the sentence.

2 Background

Figure 1 shows our problem setup. We assume a domain of a set of entities, such as SMITH and CAMBRIDGE, and relations among these (*e.g.* $profAt(\cdot,\cdot)$). We start from a knowledge base of observed logical statements, *e.g.*, $profAt(\text{SMITH}, \text{CAMBRIDGE})$ or $\forall x, y : profAt(x, y) \implies worksFor(x, y)$. These statements can be extracted from text through information extraction (for factual statements), be the output from a semantic parsing (for first-order statements) or come from human curators or external knowledge bases.

The task at hand is to predict the truth value of unseen statements, for example $worksFor(\text{SMITH}, \text{CAMBRIDGE})$. Assuming we have the corresponding formulae, logical inference can be used to arrive at this statement (arrow A in Figure 1). However, in practice the relevant background knowledge is usually missing. By contrast, a range of work (*e.g.* Bordes et al., 2011; Riedel et al., 2013) has successfully predicted unseen *factual* statements by learning entity and relation embeddings that recover the observed facts and generalize to unseen facts through dimensionality reduction (B). Inference in these approaches amounts to a series of algebraic operations on the learned embeddings that returns a numeric representation of the degree of truth (C), which can be thresholded to arrive back at a true or false statement (D) if needed.

Our goal in this view is to generalize (B) to allow richer logical statements to be recovered by low-dimensional embeddings. To this end we first describe how richer logical statements can be embedded at *full* dimension where the number of dimensions equals to the number of entities in the domain.

2.1 Tensor Calculus

Grefenstette (2013) presents an isomorphism between statements in predicate logic and expressions in tensor calculus. Let $[\cdot]$ denote this mapping from a logical expression $\mathcal{F}$ to an expression in tensor algebra. Here, logical statements evaluating to `true` or `false` are mapped to $[\texttt{true}] := \top = \begin{bmatrix}1 & 0\end{bmatrix}^T$ and $[\texttt{false}] := \bot = \begin{bmatrix}0 & 1\end{bmatrix}^T$ respectively.

Entities are represented by logical constants and mapped to one-hot vectors where each component represents a unique entity. For example, let $k = 3$ be the number of entities in a domain, then SMITH may be mapped to $[\text{SMITH}] = \begin{bmatrix}1 & 0 & 0\end{bmatrix}^T$. Unary predicates are represented as $2 \times k$ matrices, whose columns are composed of $\top$ and $\bot$ vectors. For example, for a $isProfessor$ predicate we may get

$$[isProfessor] = \begin{bmatrix}1 & 0 & 1\\ 0 & 1 & 0\end{bmatrix}.$$

In this paper we treat binary relations as unary predicates over constants $\langle \text{X}, \text{Y} \rangle$ that correspond to pairs of entities X and Y in the domain.[1]

The application of a unary predicate to a constant is realized through matrix-vector multiplication. For example, for $profAt$ and the entity pair $\langle \text{X}, \text{Y} \rangle$ we get

$$[profAt(\langle \text{X}, \text{Y} \rangle)] = [profAt]\,[\langle \text{X}, \text{Y} \rangle]\,.$$

In Grefenstette's calculus, binary boolean operators are mapped to mode 3 tensors. For example, for the implication operator holds:

$$[\implies] := \left[\begin{array}{cc|cc} 1 & 0 & 1 & 1\\ 0 & 1 & 0 & 0\end{array}\right].$$

Let A and B be two logical statements that, when evaluated in tensor algebra, yield a vector

[1] This simplifies our exposition and approach, and it can be shown that both representations are logically equivalent.

in $\{\top, \bot\}$. The application of a binary operator to statements A and B is realized via two consecutive tensor-vector products in their respective modes (see Kolda and Bader (2009) for details), *e.g.*,

$$[A \implies B] := [\implies] \times_1 [A] \times_2 [B] .$$

3 Method

Grefenstette's mapping to tensors exactly recovers the behavior of predicate logic. However, it also inherits the lack of generalization that comes with a purely symbolic representation. To overcome this problem we propose an alternate mapping. We retain the representation of truth values and boolean operators as the 2×1 and the $2 \times 2 \times 2$ sized tensors respectively. However, instead of mapping entities and predicates to one-hot representations, we estimate low-dimensional embeddings that recover the behavior of their one-hot counterparts when plugged into a set of tensor-logic statements.

In the following we first present a general learning objective that encourages low-dimensional embeddings to behave like one-hot representations. Then we show how this objective can be optimized for facts and implications.

3.1 Objective

Let $\mathfrak{R}$ be the set of all relation embeddings and $\mathfrak{P}$ be the set of all entity pair embeddings. Given a knowledge base (KB) of logical formulae K which we assume to hold, the objective is

$$\min_{[p]\in\mathfrak{P},\,[R]\in\mathfrak{R}} \sum_{\mathcal{F}\in K} \left\| [\mathcal{F}] - \top \right\|_2 . \qquad (1)$$

That is, we prefer embeddings for which the given formulae evaluate to the vector representation for truth. The same can be done for negative data by working with $\bot$, but we omit details for brevity.

To optimize this function we require the gradients of $\left\| [\mathcal{F}] - \top \right\|_2$ terms. Below we discuss these for two types of formulae: ground atoms and first-order formulae.

3.2 Ground Atoms

The KB may contain ground atoms (*i.e.* facts) of the form $\mathcal{F} = R(p)$ for a pair of entities p and a relation R. These atoms correspond to observed cells in an entity-pair-relation matrix, and injecting these facts into the embedding roughly corresponds to matrix factorization for link prediction or relation extraction (Riedel et al., 2013).

Let $\hat{\boldsymbol{\eta}}_{\mathcal{F}} := ([\mathcal{F}] - \top) \,/\, \left\| [\mathcal{F}] - \top \right\|_2$, then it is easy to show that the gradients with respect to relation embedding $[R]$ and entity pair embedding $[p]$ are

$$\partial / \partial [p] = [R] \, \hat{\boldsymbol{\eta}}_{\mathcal{F}} \quad \text{and} \quad \partial / \partial [R] = \hat{\boldsymbol{\eta}}_{\mathcal{F}} \otimes [p] .$$

3.3 First-order Formulae

Crucially, and in contrast to matrix factorization, we can inject more expressive logical formulae than just ground atoms. For example, the KB K may contain a universally quantified first-order rule such as $\forall x : R_1(x) \implies R_2(x)$. Assuming a finite domain, this statement can be unrolled into a conjunction of propositional statements of the form $\mathcal{F} = R_1(p) \implies R_2(p)$, one for each pair p. We can directly inject these propositional statements into the embeddings, and their gradients are straightfoward to derive. For example,

$$\partial / \partial [R_1] = (([\implies] \times_2 [R_2(p)]) \, \hat{\boldsymbol{\eta}}_{\mathcal{F}}) \otimes [p] .$$

3.4 Learning and Inference

We learn embeddings for entity pairs and relations by minimizing objective 1 using stochastic gradient descent (SGD). To infer the (two-dimensional) truth value (C in Figure 1) of a formula $\mathcal{F}$ in embedded logic we evaluate $[\mathcal{F}]$. An easier to intpret one-dimensional representation can be derived by

$$\left(\langle [\mathcal{F}] , \begin{bmatrix} 1 & -1 \end{bmatrix}^T \rangle + 1 \right) / 2,$$

followed by truncation to the interval $[0, 1]$. Other ways of projecting $[\mathcal{F}]$ to $\mathbb{R}$, such as using cosine similarity to $\top$, are possible as well.

4 Experiments

We perform experiments on synthetic data defined over 7 entity pairs and 6 relations. We fix the embedding size k to 4 and train the model for 100 epochs using SGD with ℓ_2-regularization on the values of the embeddings. The learning rate and the regularization parameter are set to 0.05.

The left part of Table 1 shows the observed (bold) and inferred truth values for a set of factual stamements of the form $R(p)$, mapped to $\mathbb{R}$ as discussed above. Due to the generalization obtained by low-dimensional embeddings, the model infers that, for example, SMITH is an employee at CAMBRIDGE and DAVIES lives in LONDON. However, we would like the model to also capture that every professor works for his or her university

	With Factual Constraints						With Factual and First-Order Constraints					
	profAt	*worksFor*	*employeeAt*	*registeredIn*	*livesIn*	*bornIn*	*profAt*	*worksFor*	*employeeAt*	*registeredIn*	*livesIn*	*bornIn*
⟨JONES, UWASH⟩	**1.00**	**1.00**	**1.00**	0.00	0.18	0.01	**0.98**	**0.98**	**0.95**	0.03	0.00	0.04
⟨TAYLOR, UCL⟩	**1.00**	**1.00**	**0.98**	0.00	0.20	0.00	**0.98**	**0.96**	**0.95**	0.05	0.00	0.06
⟨SMITH, CAMBRIDGE⟩	**0.98**	$^{\top}$ 0.00	$^{\top}$ 0.64	0.75	0.07	0.72	**0.92**	$^{\top}$ 0.97	$^{\top}$ 0.89	0.04	0.04	0.05
⟨WILLIAMS, OXFORD⟩	$^{\bot}$ 0.02	**1.00**	0.08	0.00	0.93	0.02	$^{\bot}$ 0.05	**0.91**	0.02	0.05	0.87	0.06
⟨BROWN, CAMBRIDGE⟩	$^{\bot}$ 0.00	**0.97**	0.02	$^{\bot}$ 0.01	**0.95**	0.06	$^{\bot}$ 0.01	**0.90**	0.00	$^{\bot}$ 0.07	**0.92**	0.07
⟨DAVIES, LONDON⟩	0.00	0.00	0.00	**0.99**	$^{\top}$ 0.50	**1.00**	0.01	0.00	0.00	**0.98**	$^{\top}$ 0.98	**0.97**
⟨EVANS, PARIS⟩	0.00	0.00	0.00	**1.00**	$^{\top}$ 0.48	**1.00**	0.00	0.00	0.00	**0.97**	$^{\top}$ 1.00	**0.96**

Table 1: Reconstructed matrix without (left) and with (right) the first-order constraints $profAt \implies worksFor$ and $registeredIn \implies livesIn$. Predictions for training cells of factual constraints $[R(p)] = \top$ are shown in bold, and `true` and `false` test cells are denoted by $^{\top}$ and $^{\bot}$ respectively.

and that, when somebody is registered in a city, he or she also lives in that city.

When including such first-order constraints (right part of Table 1), the model's predictions improve concerning different aspects. First, the model gets the implication right, demonstrating that the low-dimensional embeddings encode first-order knowledge. Second, this implication transitively improves the predictions on other columns (*e.g.* SMITH is an employee at CAMBRIDGE). Third, the implication works indeed in an asymmetric way, *e.g.*, the model does not predict that WILLIAMS is a professor at OXFORD just because she is working there.

5 Related Work

The idea of bringing together distributional semantics and formal logic is not new. Lewis and Steedman (2013) improve the generalization performance of a semantic parser via the use of distributional representations. However, their target representation language is still symbolic, and it is unclear how this approach can cope with noise and uncertainty in data.

Another line of work (Clark and Pulman, 2007; Mitchell and Lapata, 2008; Coecke et al., 2010; Socher et al., 2012; Hermann and Blunsom, 2013) uses symbolic representations to guide the composition of distributional representations. Reading a sentence or logical formula there amounts to compositionally mapping it to a k-dimensional vector that then can be used for downstream tasks. We propose a very different approach: Reading a sentence amounts to updating the involved entity pair and relation embeddings such that the sentence evaluates to *true*. Afterwards we cannot use the embeddings to calculate sentence similarities, but to answer relational questions about the world.

Similar to our work, Bowman (2014) provides further evidence that distributed representations can indeed capture logical reasoning. Although Bowman demonstrates this on natural logic expressions without capturing factual statements, one can think of ways to include the latter in his framework as well. However, the approach presented here can conceptually inject complex nested logical statements into embeddings, whereas it is not obvious how this can be achieved in the neural-network based multi-class classification framework proposed by Bowman.

6 Conclusion

We have argued that low dimensional embeddings of entities and relations may be tuned to simulate the behavior of logic and hence combine the advantages of distributional representations with those of their symbolic counterparts. As a first step into this direction we have presented an objective that encourages embeddings to be consistent with a given logical knowledge base that includes facts and first-order rules. On a small synthetic dataset we optimize this objective with SGD to learn low-dimensional embeddings that indeed follow the behavior of the knowledge base.

Clearly we have only scratched the surface here. Besides only using toy data and logical formulae of very limited expressiveness, there are fundamental questions we have yet to address. For example, even if the embeddings could enable perfect logical reasoning, how do we provide provenance or proofs of answers? Moreover, in practice a machine reader (*e.g.* a semantic parser) *incrementally* gathers logical statements from text— how could we *incrementally* inject this knowledge into embeddings without retraining the whole model? Finally, what are the theoretical limits of embedding logic in vector spaces?

Acknowledgments

We would like to thank Giorgos Spithourakis, Thore Graepel, Karl Moritz Hermann and Edward Grefenstette for helpful discussions, and Andreas Vlachos for comments on the manuscript. This work was supported by Microsoft Research through its PhD Scholarship Programme. This work was supported in part by the TerraSwarm Research Center, one of six centers supported by the STARnet phase of the Focus Center Research Program (FCRP) a Semiconductor Research Corporation program sponsored by MARCO and DARPA.

References

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

Johan Bos and Katja Markert. 2005. Recognising textual entailment with logical inference. In *Proc. of HLT/EMNLP*, pages 628–635.

Samuel R Bowman. 2014. Can recursive neural tensor networks learn logical reasoning? In *ICLR'14*.

Stephen Clark and Stephen Pulman. 2007. Combining symbolic and distributional models of meaning. In *AAAI Spring Symposium: Quantum Interaction*, pages 52–55.

Bob Coecke, Mehrnoosh Sadrzadeh, and Stephen Clark. 2010. Mathematical foundations for a compositional distributional model of meaning. *CoRR*, abs/1003.4394.

Edward Grefenstette. 2013. Towards a formal distributional semantics: Simulating logical calculi with tensors. In *Proc. of *SEM*, pages 1–10.

Karl Moritz Hermann and Phil Blunsom. 2013. The role of syntax in vector space models of compositional semantics. In *Proc. of ACL*, pages 894–904.

Tamara G Kolda and Brett W Bader. 2009. Tensor decompositions and applications. *SIAM review*, 51(3):455–500.

Mike Lewis and Mark Steedman. 2013. Combined distributional and logical semantics. In *TACL*, volume 1, pages 179–192.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proc. of ACL-IJCNLP*, pages 1003–1011.

Jeff Mitchell and Mirella Lapata. 2008. Vector-based models of semantic composition. In *Proc. of ACL*, pages 236–244.

Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M Marlin. 2013. Relation extraction with matrix factorization and universal schemas. In *Proc. of NAACL-HLT*, pages 74–84.

Richard Socher, Brody Huval, Christopher D Manning, and Andrew Y Ng. 2012. Semantic compositionality through recursive matrix-vector spaces. In *Proc. of EMNLP*, pages 1201–1211.

Luke S Zettlemoyer and Michael Collins. 2005. Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars. In *Proc. of UAI*, pages 658–666.