Multilingual Relation Extraction using Compositional Universal Schema

Patrick Verga, David Belanger, Emma Strubell, Benjamin Roth & Andrew McCallum
College of Information and Computer Sciences
University of Massachusetts Amherst
{pat, belanger, strubell, beroth, mccallum}@cs.umass.edu

Abstract

Universal schema builds a knowledge base (KB) of entities and relations by jointly embedding all relation types from input KBs as well as textual patterns observed in raw text. In most previous applications of universal schema, each textual pattern is represented as a single embedding, preventing generalization to unseen patterns. Recent work employs a neural network to capture patterns' compositional semantics, providing generalization to all possible input text. In response, this paper introduces significant further improvements to the coverage and flexibility of universal schema relation extraction: predictions for entities unseen in training and multilingual transfer learning to domains with no annotation. We evaluate our model through extensive experiments on the English and Spanish TAC KBP benchmark, outperforming the top system from TAC 2013 slot-filling using no handwritten patterns or additional annotation. We also consider a multilingual setting in which English training data entities overlap with the seed KB, but Spanish text does not. Despite having no annotation for Spanish data, we train an accurate predictor, with additional improvements obtained by tying word embeddings across languages. Furthermore, we find that multilingual training improves English relation extraction accuracy. Our approach is thus suited to broad-coverage automated knowledge base construction in a variety of languages and domains.

1 Introduction

The goal of automatic knowledge base construction (AKBC) is to build a structured knowledge base (KB) of facts using a noisy corpus of raw text evidence, and perhaps an initial seed KB to be augmented (Carlson et al., 2010; Suchanek et al., 2007; Bollacker et al., 2008). AKBC supports downstream reasoning at a high level about extracted entities and their relations, and thus has broad-reaching applications to a variety of domains.

One challenge in AKBC is aligning knowledge from a structured KB with a text corpus in order to perform supervised learning through *distant supervision*. *Universal schema* (Riedel et al., 2013) along with its extensions (Yao et al., 2013; Gardner et al., 2014; Neelakantan et al., 2015; Rocktaschel et al., 2015), avoids alignment by jointly embedding KB relations, entities, and surface text patterns. This propagates information between KB annotation and corresponding textual evidence.

The above applications of universal schema express each text relation as a distinct item to be embedded. This harms its ability to generalize to inputs not precisely seen at training time. Recently, Toutanova et al. (2015) addressed this issue by embedding text patterns using a deep sentence encoder, which captures the compositional semantics of textual relations and allows for prediction on inputs never seen before.

This paper further expands the coverage abilities of universal schema relation extraction by introducing techniques for forming predictions for new entities unseen in training and even for new domains with no associated annotation. In the extreme example of domain adaptation to a completely new language, we may have limited linguistic resources or labeled data such as treebanks, and only rarely a KB with adequate coverage. Our method performs multilingual transfer learning, providing a predictive model for a language with no coverage in an existing KB, by leveraging common representations for shared entities across text corpora. As depicted in Figure 1, we simply require that one language have an available KB of seed facts. We can further improve our models by tying a small set of word embeddings across languages using only simple knowledge about word-level translations, learning to embed semantically similar textual patterns from different languages into the same latent space.

In extensive experiments on the TAC Knowledge Base Population (KBP) slot-filling benchmark we outperform the top 2013 system with an F1 score of 40.7 and perform relation extraction in Spanish with no labeled data or direct overlap between the Spanish training corpus and

Proceedings of NAACL-HLT 2016, pages 886–896,
San Diego, California, June 12-17, 2016.

the training KB, demonstrating that our approach is well-suited for broad-coverage AKBC in low-resource languages and domains. Interestingly, joint training with Spanish improves English accuracy.

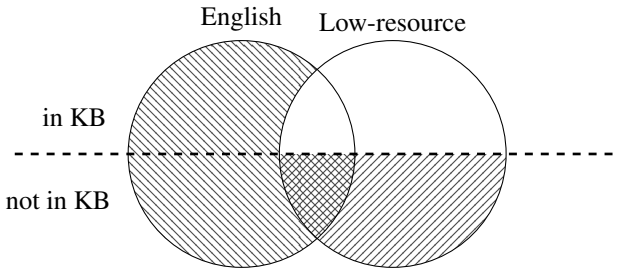


Figure 1: Splitting the entities in a multilingual AKBC training set into parts. We only require that entities in the two corpora overlap. Remarkably, we can train a model for the low-resource language even if entities in the low-resource language do not occur in the KB.

2 Background

AKBC extracts unary attributes of the form (*subject*, *attribute*), typed binary relations of the form (*subject*, *relation*, *object*), or higher-order relations. We refer to subjects and objects as *entities*. This work focuses solely on extracting binary relations, though many of our techniques generalize naturally to unary prediction. Generally, for example in Freebase (Bollacker et al., 2008), higher-order relations are expressed in terms of collections of binary relations.

We now describe prior work on approaches to AKBC. They all aim to predict (*s, r, o*) triples, but differ in terms of: (1) input data leveraged, (2) types of annotation required, (3) definition of relation label schema, and (4) whether they are capable of predicting relations for entities unseen in the training data. Note that all of these methods require pre-processing to detect entities, which may result in additional KB construction errors.

2.1 Relation Extraction as Link Prediction

A knowledge base is naturally described as a graph, in which entities are nodes and relations are labeled edges (Suchanek et al., 2007; Bollacker et al., 2008). In the case of *knowledge graph completion*, the task is akin to link prediction, assuming an initial set of (*s, r, o*) triples. See Nickel et al. (2015) for a review. No accompanying text data is necessary, since links can be predicted using properties of the graph, such as transitivity. In order to generalize well, prediction is often posed as low-rank matrix or tensor factorization. A variety of model variants have been suggested, where the probability of a given edge existing depends on a multi-linear form (Nickel et al., 2011; García-Durán et al., 2015; Yang et al., 2015; Bordes et al., 2013; Wang et al., 2014; Lin et al., 2015), or non-linear interactions between s, r, and o (Socher et al., 2013). Other approaches model the compositionality of multi-hop paths, typically for question answering (Bordes et al., 2014; Gu et al., 2015; Neelakantan et al., 2015).

2.2 Relation Extraction as Sentence Classification

Here, the training data consist of (1) a text corpus, and (2) a KB of seed facts with provenance, i.e. supporting evidence, in the corpus. Given individual an individual sentence, and pre-specified entities, a classifier predicts whether the sentence expresses a relation from a target schema. To train such a classifier, KB facts need to be aligned with supporting evidence in the text, but this is often challenging. For example, not all sentences containing Barack and Michelle Obama state that they are married. A variety of one-shot and iterative methods have addressed the alignment problem (Bunescu and Mooney, 2007; Mintz et al., 2009; Riedel et al., 2010; Yao et al., 2010; Hoffmann et al., 2011; Surdeanu et al., 2012; Min et al., 2013; Zeng et al., 2015). An additional degree of freedom in these approaches is whether they classify individual sentences or predicting at the corpus level by aggregating information from all sentences containing a given pair of entities before prediction. The former approach is often preferable in practice, due to the simplicity of independently classifying individual sentences and the ease of associating each prediction with a provenance. Prior work has applied deep learning to small-scale relation extraction problems, where functional relationships are detected between common nouns (Li et al., 2015; dos Santos et al., 2015). Xu et al. (2015) apply an LSTM to a parse path, while Zeng et al. (2015) use a CNN on the raw text, with a special temporal pooling operation to separately embed the text around each entity.

2.3 Open-Domain Relation Extraction

In the previous two approaches, prediction is carried out with respect to a fixed schema R of possible relations r. This may overlook salient relations that are expressed in the text but do not occur in the schema. In response, *open-domain* information extraction (OpenIE) lets the text speak for itself: R contains all possible patterns of text occurring between entities s and o (Banko et al., 2007; Etzioni et al., 2008; Yates and Etzioni, 2007). These are obtained by filtering and normalizing the raw text. The approach offers impressive coverage, avoids issues of distant supervision, and provides a useful exploratory tool. On the other hand, OpenIE predictions are difficult to use in downstream tasks that expect information from a fixed schema.

Table 1 provides examples of OpenIE patterns. The examples in row two and three illustrate relational contexts

for which similarity is difficult to be captured by an OpenIE approach because of their syntactically complex constructions. This motivates the technique in Section 3.2, which uses a deep architecture applied to raw tokens, instead of rigid rules for normalizing text to obtain patterns.

Sentence (context tokens italicized)	OpenIE pattern
Khan *'s younger sister,* **Annapurna Devi**, who later married Shankar, developed into an equally accomplished master of the surbahar, but custom prevented her from performing in public.	*arg1*'s * sister *arg2*
A professor emeritus at Yale, **Mandelbrot** *was born in Poland but as a child moved with his family to* **Paris** where he was educated.	*arg1* * moved with * family to *arg2*
Kissel *was born in Provo, Utah, but her family also lived in* **Reno**.	*arg1* * lived in *arg2*

Table 1: Examples of sentences expressing relations. Context tokens (italicized) consist of the text occurring between entities (bold) in a sentence. OpenIE patterns are obtained by normalizing the context tokens using hand-coded rules. The top example expresses the per:siblings relation and the bottom two examples both express the per:cities_of_residence relation.

2.4 Universal Schema

When applying Universal Schema (Riedel et al., 2013) (USchema) to relation extraction, we combine the OpenIE and link-prediction perspectives. By jointly modeling both OpenIE patterns and the elements of a target schema, the method captures broader relational structure than multi-class classification approaches that just model the target schema. Furthermore, the method avoids the distant supervision alignment difficulties of Section 2.2.

Riedel et al. (2013) augment a knowledge graph from a seed KB with additional edges corresponding to OpenIE patterns observed in the corpus. Even if the user does not seek to predict these new edges, a joint model over all edges can exploit regularities of the OpenIE edges to improve modeling of the labels from the target schema.

The data still consist of (s, r, o) triples, which can be predicted using link-prediction techniques such as low-rank factorization. Riedel et al. (2013) explore a variety of approximations to the 3-mode (s, r, o) tensor. One such probabilistic model is:

$$\mathbb{P}\left((s, r, o)\right) = \sigma\left(u_{s,o}^{\top} v_r\right), \quad (1)$$

where $\sigma()$ is a sigmoid function, $u_{s,o}$ is an embedding of the entity pair (s, o), and v_r is an embedding of the relation r, which may be an OpenIE pattern or a relation from the target schema. All of the exposition and results in this paper use this factorization, though many of the techniques we present later could be applied easily to the other factorizations described in Riedel et al. (2013). Note that learning unique embeddings for OpenIE relations does not guarantee that similar patterns, such as the final two in Table 1, will be embedded similarly.

As with most of the techniques in Section 2.1, the data only consist of positive examples of edges. The absence of an annotated edge does not imply that the edge is false. In fact, we seek to predict some of these missing edges as true. Riedel et al. (2013) employ the Bayesian Personalized Ranking (BPR) approach of Rendle et al. (2009), which does not explicitly model unobserved edges as negative, but instead seeks to rank the probability of observed triples above unobserved triples.

Recently, Toutanova et al. (2015) extended USchema to not learn individual pattern embeddings v_r, but instead to embed text patterns using a deep architecture applied to word tokens. This shares statistical strength between OpenIE patterns with similar words. We leverage this approach in Section 3.2. Additional work has modeled the regularities of multi-hop paths through knowledge graph augmented with text patterns (Lao et al., 2011; Lao et al., 2012; Gardner et al., 2014; Neelakantan et al., 2015).

2.5 Multilingual Embeddings

Much work has been done on multilingual word embeddings. Most of this work uses aligned sentences from the Europarl dataset (Koehn, 2005) to align word embeddings across languages (Gouws et al., 2015; Luong et al., 2015; Hermann and Blunsom, 2014). Others (Mikolov et al., 2013; Faruqui et al., 2014) align separate single-language embedding models using a word-level dictionary. Mikolov et al. (2013) use translation pairs to learn a linear transform from one embedding space to another.

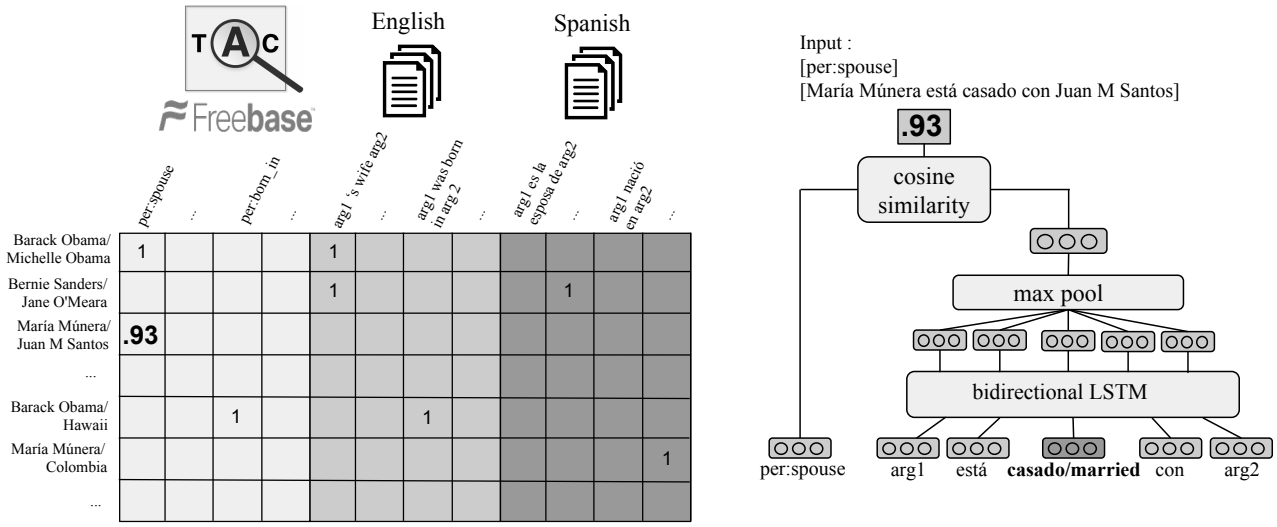
However, very little work exists on multilingual relation extraction. Faruqui and Kumar (2015) perform multilingual OpenIE relation extraction by projecting all languages to English using Google translate. However, as explained in Section 2.3 the OpenIE paradigm is not amenable to prediction within a fixed schema. Further, their approach does not generalize to low-resource languages where translation is unavailable – while we use translation dictionaries to improve our results, our experiments demonstrate that our method is effective even without this resource.

3 Methods

3.1 Universal Schema as Sentence Classifier

Similar to many link prediction approaches, (Riedel et al., 2013) perform transductive learning, where a model is learned jointly over train and test data. Predictions are made by using the model to identify edges that were unobserved in the test data but likely to be true. The approach is vulnerable to the *cold start* problem in collab-

Figure 2: Universal Schema jointly embeds KB and textual relations from Spanish and English, learning dense representations for entity pairs and relations using matrix factorization. Cells with a 1 indicate triples observed during training (left). The bold score represents a test-time prediction by the model (right). Using transitivity through KB/English overlap and English/Spanish overlap, our model can predict that a text pattern in Spanish evidences a KB relation despite no overlap between Spanish/KB entity pairs. At train time we use BPR loss to maximize the inner product of entity pairs with KB relations and text patterns encoded using a bidirectional LSTM. At test time we score compatibility between embedded KB relations and encoded textual patterns using cosine similarity. In our Spanish model we treat embeddings for a small set of English/Spanish translation pairs as a single word, e.g. casado and married.



orative filtering (Schein et al., 2002): it is unclear how to form predictions for unseen entity pairs, without refactorizing the entire matrix or applying heuristics.

In response, this paper re-purposes USchema as a means to train a sentence-level relation classifier, like those in Section 2.2. This allows us to avoid errors from aligning distant supervision to the corpus, but is more deployable for real world applications. It also provides opportunities in Section 3.4 to improve multilingual AKBC.

We produce predictions using a very simple approach: (1) scan the corpus and extract a large quantity of triplets (s, r_{text}, o), where r_{text} is an OpenIE pattern. For each triplet, if the similarity between the embedding of r_{text} and the embedding of a target relation r_{schema} is above some threshold, we predict the triplet $(s, r_{\text{schema}}, o)$, and its provenance is the input sentence containing (s, r_{text}, o). We refer to this technique as *pattern scoring*. In our experiments, we use the cosine distance between the vectors (Figure 2). In Section 7.3, we discuss details for how to make this distance well-defined.

3.2 Using a Compositional Sentence Encoder to Predict Unseen Text Patterns

The pattern scoring approach is subject to an additional cold start problem: input data may contain patterns unseen in training. This section describes a method for using USchema to train a relation classifier that can take arbitrary context tokens (Section 2.3) as input.

Fortunately, the cold start problem for context tokens is more benign than that of entities since we can exploit statistical regularities of text: similar sequences of context tokens should be embedded similarly. Therefore, following Toutanova et al. (2015), we embed raw context tokens compositionally using a deep architecture. Unlike Riedel et al. (2013), this requires no manual rules to map text to OpenIE patterns and can embed any possible input string. The modified USchema likelihood is:

$$\mathbb{P}\left((s, r, o)\right) = \sigma\left(u_{s,o}^{\top}\text{Encoder}(r)\right). \qquad (2)$$

Here, if r is raw text, then $\text{Encoder}(r)$ is parameterized by a deep architecture. If r is from the target schema, $\text{Encoder}(r)$ is a produced by a lookup table (as in traditional USchema). Though such an encoder increases the computational cost of test-time prediction over straightforward pattern matching, evaluating a deep architecture can be done in large batches in parallel on a GPU.

Both convolutional networks (CNNs) and recurrent networks (RNNs) are reasonable encoder architectures, and we consider both in our experiments. CNNs have been useful in a variety of NLP applications (Collobert et al., 2011; Kalchbrenner et al., 2014; Kim, 2014). Unlike Toutanova et al. (2015), we also consider RNNs, specifically Long-Short Term Memory Networks

(LSTMs) (Hochreiter and Schmidhuber, 1997). LSTMs have proven successful in a variety of tasks requiring encoding sentences as vectors (Sutskever et al., 2014; Vinyals et al., 2014). In our experiments, LSTMs outperform CNNs.

There are two key differences between our sentence encoder and that of Toutanova et al. (2015). First, we use the encoder at test time, since we process the context tokens for held-out data. On the other hand, Toutanova et al. (2015) adopt the transductive approach where the encoder is only used to help train better representations for the relations in the target schema; it is ignored when forming predictions. Second, we apply the encoder to the raw text between entities, while Toutanova et al. (2015) first perform syntactic dependency parsing on the data and then apply an encoder to the path between the two entities in the parse tree. We avoid parsing, since we seek to perform multilingual AKBC, and many languages lack linguistic resources such as treebanks. Even parsing non-newswire English text, such as tweets, is extremely challenging.

3.3 Modeling Frequent Text Patterns

Despite the coverage advantages of using a deep sentence encoder, separately embedding each OpenIE pattern, as in Riedel et al. (2013), has key advantages. In practice, we have found that many high-precision patterns occur quite frequently. For these, there is sufficient data to model them with independent embeddings per pattern, which imposes minimal inductive bias on the relationship between patterns. Furthermore, some discriminative phrases are idiomatic, i.e.. their meaning is not constructed compositionally from their constituents. For these, a sentence encoder may be inappropriate.

Therefore, pattern embeddings and deep token-based encoders have very different strengths and weaknesses. One values specificity, and models the head of the text distribution well, while the other has high coverage and captures the tail. In experimental results, we demonstrate that an ensemble of both models performs substantially better than either in isolation.

3.4 Multilingual Relation Extraction with Zero Annotation

The models described in previous two sections provide broad-coverage relation extraction that can generalize to all possible input entities and text patterns, while avoiding error-prone alignment of distant supervision to a corpus. Next, we describe techniques for an even more challenging generalization task: relation classification for input sentences in completely different languages.

Training a sentence-level relation classifier, either using the alignment-based techniques of Section 2.2, or the alignment-free method of Section 3.1, requires an available KB of seed facts that have supporting evidence in the corpus. Unfortunately, available KBs have low overlap with corpora in many languages, since KBs have cultural and geographical biases. In response, we perform multilingual relation extraction by jointly modeling a high-resource language, such as English, and an alternative language with no KB annotation. This approach provides transfer learning of a predictive model to the alternative language, and generalizes naturally to modeling more languages.

Extending the training technique of Section 3.1 to corpora in multiple languages can be achieved by factorizing a matrix that mixes data from a KB and from the two corpora. In Figure 1 we split the entities of a multilingual training corpus into sets depending on whether they have annotation in a KB and what corpora they appear in. We can perform transfer learning of a relation extractor to the low-resource language if there are entity pairs occurring in the two corpora, even if there is no KB annotation for these pairs. Note that we do not use the entity pair embeddings at test time: They are used only to bridge the languages during training. To form predictions in the low-resource language, we can simply apply the pattern scoring approach of Section 3.1.

In Section 5, we demonstrate that jointly learning models for English and Spanish, with no annotation for the Spanish data, provides fairly accurate Spanish AKBC, and even improves the performance of the English model. Note that we are not performing *zero-shot* learning of a Spanish model (Larochelle et al., 2008). The relations in the target schema are language-independent concepts, and we have supervision for these in English.

3.5 Tied Sentence Encoders

The sentence encoder approach of Section 3.2 is complementary to our multilingual modeling technique: we simply use a separate encoder for each language. This approach is sub-optimal, however, because each sentence encoder will have a separate matrix of word embeddings for its vocabulary, despite the fact that there may be considerable shared structure between the languages. In response, we propose a straightforward method for tying the parameters of the sentence encoders across languages.

Drawing on the dictionary-based techniques described in Section 2.5, we first obtain a list of word-word translation pairs between the languages using a translation dictionary. The first layer of our deep text encoder consists of a word embedding lookup table. For the aligned word types, we use a single cross-lingual embedding. Details of our approach are described in Appendix 7.5.

4 Task and System Description

We focus on the TAC KBP slot-filling task. Much related work on embedding knowledge bases evaluates on

the FB15k dataset (Bordes et al., 2013; Wang et al., 2014; Lin et al., 2015; Yang et al., 2015; Toutanova et al., 2015). Here, relation extraction is posed as link prediction on a subset of Freebase. This task does not capture the particular difficulties we address: (1) evaluation on entities and text unseen during training, and (2) zero-annotation learning of a predictor for a low-resource language.

Also, note both Toutanova et al. (2015) and Riedel et al. (2013) explore the pros and cons of learning embeddings for entity pairs vs. separate embeddings for each entity. As this is orthogonal to our contributions, we only consider entity pair embeddings, which performed best in both works when given sufficient data.

4.1 TAC Slot-Filling Benchmark

The aim of the TAC benchmark is to improve both coverage and quality of relation extraction evaluation compared to just checking the extracted facts against a knowledge base, which can be incomplete and where the provenances are not verified. In the slot-filling task, each system is given a set of paired query entities and relations or 'slots' to fill, and the goal is to correctly fill as many slots as possible along with provenance from the corpus. For example, given the query entity/relation pair (*Barack Obama, per:spouse*), the system should return the entity *Michelle Obama* along with sentence(s) whose text expresses that relation. The answers returned by all participating teams, along with a human search (with timeout), are judged manually for correctness, i.e. whether the provenance specified by the system indeed expresses the relation in question.

In addition to verifying our models on the 2013 and 2014 English slot-filling task, we evaluate our Spanish models on the 2012 TAC Spanish slot-filling evaluation. Because this TAC track was never officially run, the coverage of facts in the available annotation is very small, resulting in many correct predictions being marked incorrectly as precision errors. In response, we manually annotated all results returned by the models considered in Table 4. Precision and recall are calculated with respect to the union of the TAC annotation and our new labeling[1].

4.2 Retrieval Pipeline

Our retrieval pipeline first generates all valid slot filler candidates for each query entity and slot, based on entities extracted from the corpus using FACTORIE (McCallum et al., 2009) to perform tokenization, segmentation, and entity extraction. We perform entity linking by heuristically linking all entity mentions from our text corpora to a Freebase entity using anchor text in Wikipedia. Making use of the fact that most Freebase entries contain a link to the corresponding Wikipedia page, we link all entity mentions from our text corpora to a Freebase entity by the following process: First, a set of candidate entities is obtained by following frequent link anchor text statistics. We then select that candidate entity for which the cosine similarity between the respective Wikipedia and the sentence context of the mention is highest, and link to that entity if a threshold is exceeded.

An entity pair qualifies as a candidate prediction if it meets the type criteria for the slot.[2] The TAC 2013 English and Spanish newswire corpora each contain about 1 million newswire documents from 2009–2012. The document retrieval and entity matching components of our relation extraction pipeline are based on RelationFactory (Roth et al., 2014), the top-ranked system of the 2013 English slot-filling task. We also use the English distantly supervised training data from this system, which aligns the TAC 2012 corpus to Freebase. More details on alignment are described in Appendix 7.4.

As discussed in Section 3.3, models using a deep sentence encoder and using a pattern lookup table have complementary strengths and weaknesses. In response, we present results where we ensemble the outputs of the two models by simply taking the union of their individual outputs. Slightly higher results might be obtained through more sophisticated ensembling schemes.

4.3 Model Details

All models are implemented in Torch (code publicly available[3]). Models are tuned to maximize F1 on the 2012 TAC KBP slot-filling evaluation. We additionally tune the thresholds of our pattern scorer on a per-relation basis to maximize F1 using 2012 TAC slot-filling for English and the 2012 Spanish slot-filling development set for Spanish. As in Riedel et al. (2013), we train using the BPR loss of Rendle et al. (2009). Our CNN is implemented as described in Toutanova et al. (2015), using width-3 convolutions, followed by tanh and max pool layers. The LSTM uses a bi-directional architecture where the forward and backward representations of each hidden state are averaged, followed by max pooling over time. See Section 7.2

We also report results including an alternate names (AN) heuristic, which uses automatically-extracted rules to detect the TAC 'alternate name' relation. To achieve this, we collect frequent Wikipedia link anchor texts for

[1] Following Surdeanu et al. (2012) we remove facts about undiscovered entities to correct for recall.

[2] Due to the difficulty of retrieval and entity detection, the maximum recall for predictions is limited. For this reason, Surdeanu et al. (2012) restrict the evaluation to answer candidates returned by their system and effectively rescaling recall. We do not perform such a re-scaling in our English results in order to compare to other reported results. Our Spanish numbers are rescaled. All scores reflect the 'anydoc' (relaxed) scoring to mitigate penalizing effects for systems not included in the evaluation pool.

[3] https://github.com/patverga/torch-relation-extraction

Model	Recall	Precision	F1
CNN	31.6	36.8	34.1
LSTM	32.2	39.6	**35.5**
USchema	29.4	42.6	34.8
USchema+LSTM	34.4	41.9	37.7
USchema+LSTM+Es	38.1	40.2	**39.2**
USchema+LSTM+AN	36.7	43.1	39.7
USchema+LSTM+Es+AN	40.2	41.2	**40.7**
Roth et al. (2014)	35.8	45.7	40.2

Table 2: Precision, recall and F1 on the English TAC 2013 slot-filling task. AN refers to alternative names heuristic and Es refers to the addition of Spanish text at train time. LSTM+USchema ensemble outperforms any single model, including the highly-tuned top 2013 system of Roth et al. (2014), despite using no handwritten patterns.

Model	Recall	Precision	F1
CNN	28.1	29.0	28.5
LSTM	27.3	32.9	**29.8**
USchema	24.3	35.5	28.8
USchema+LSTM	34.1	29.3	31.5
USchema+LSTM+Es	34.4	31.0	**32.6**

Table 3: Precision, recall and F1 on the English TAC 2014 slot-filling task. Es refers to the addition of Spanish text at train time. The AN heuristic is ineffective on 2014 adding only 0.2 to F1. Our system would rank 4/18 in the official TAC 2014 competition behind systems that use hand-written patterns and active learning despite our system using neither of these additional annotations (Surdeanu and Ji., 2014).

each query entity. If a high probability anchor text co-occurs with the canonical name of the query in the same document, we return the anchor text as a slot filler.

5 Experimental Results

In experiments on the English and Spanish TAC KBC slot-filling tasks, we find that both USchema and LSTM models outperform the CNN across languages, and that the LSTM tends to perform slightly better than USchema as the only model. Ensembling the LSTM and USchema models further increases final F1 scores in all experiments, suggesting that the two different types of model compliment each other well. Indeed, in Section 5.3 we present quantitative and qualitative analysis of our results which further confirms this hypothesis: the LSTM and USchema models each perform better on different pattern lengths and are characterized by different precision-recall tradeoffs.

Model	Recall	Precision	F1
LSTM	9.3	12.5	10.7
LSTM+Dict	14.7	15.7	15.2
USchema	15.2	17.5	16.3
USchema+LSTM	21.7	14.5	17.3
USchema+LSTM+Dict	26.9	15.9	**20.0**

Table 4: Zero-annotation transfer learning F1 scores on 2012 Spanish TAC KBP slot-filling task. Adding a translation dictionary improves all encoder-based models. Ensembling LSTM and USchema models performs the best.

5.1 English TAC Slot-filling Results

Tables 2 and 3 present the performance of our models on the 2013 and 2014 English TAC slot-filling tasks. Ensembling the LSTM and USchema models improves F1 by 2.2 points for 2013 and 1.7 points for 2014 over the strongest single model on both evaluations, LSTM. Adding the alternative names (AN) heuristic described in Section 4.3 increases F1 by an additional 2 points on 2013, resulting in an F1 score that is competitive with the state-of-the-art. We also demonstrate the effect of jointly learning English and Spanish models on English slot-filling performance. Adding Spanish data improves our F1 scores by 1.5 points on 2013 and 1.1 on 2014 over using English alone. This places are system higher than the top performer at the 2013 TAC slot-filling task even though our system uses no hand-written rules.

The state of the art systems on this task all rely on matching handwritten patterns to find additional answers while our models use only automatically generated, indirect supervision; even our AN heuristics (Section 4.2) are automatically generated. The top two 2014 systems were Angeli et al. (2014) and RPI Blender (Surdeanu and Ji., 2014) who achieved F1 scores of 39.5 and 36.4 respectively. Both of these systems used additional active learning annotation. The third place team (Lin et al., 2014) relied on highly tuned patterns and rules and achieved an F1 score of 34.4.

Our model performs substantially better on 2013 than 2014 for two reasons. First, our RelationFactory (Roth et al., 2014) retrieval pipeline was a top retrieval pipeline on the 2013 task, but was outperformed on the 2014 task which introduced new challenges such as confusable entities. Second, improved training using active learning gave the top 2014 systems a boost in performance. No 2013 systems, including ours, use active learning. Bentor et al. (2014), the 4th place team in the 2014 evaluation, used the same retrieval pipeline (Roth et al., 2014) as our model and achieved an F1 score of 32.1.

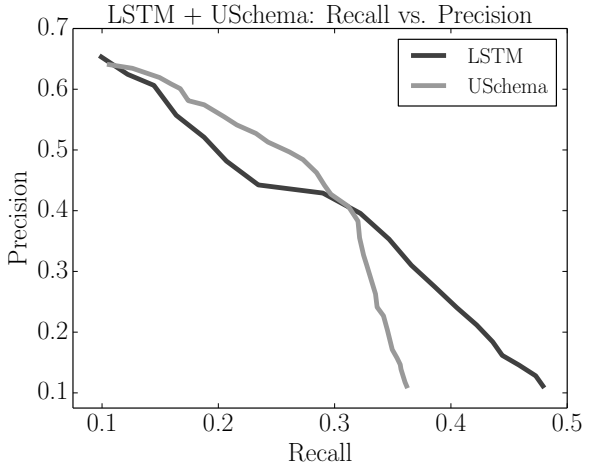


Figure 3: Precision-Recall curves for USchema and LSTM on 2013 TAC slot-filling. USchema achieves higher precision values whereas LSTM has higher recall.

5.2 Spanish TAC Slot-filling Results

Table 4 presents 2012 Spanish TAC slot-filling results for our multilingual relation extractors trained using zero-annotation transfer learning. Tying word embeddings between the two languages results in substantial improvements for the LSTM. We see that ensembling the non-dictionary LSTM with USchema gives a slight boost over USchema alone, but ensembling the dictionary-tied LSTM with USchema provides a significant increase of nearly 4 F1 points over the highest-scoring single model, USchema. Clearly, grounding the Spanish data using a translation dictionary provides much better Spanish word representations. These improvements are complementary to the baseline USchema model, and yield impressive results when ensembled.

In addition to embedding semantically similar phrases from English and Spanish to have high similarity, our models also learn high-quality multilingual word embeddings. In Table 5 we compare Spanish nearest neighbors of English query words learned by the LSTM with dictionary ties versus the LSTM with no ties, using no unsupervised pre-training for the embeddings. Both approaches jointly embed Spanish and English word types, using shared entity embeddings, but the dictionary-tied model learns qualitatively better multilingual embeddings.

5.3 USchema vs LSTM

We further analyze differences between USchema and LSTM in order to better understand why ensembling the models results in the best performing system. Figure 3 depicts precision-recall curves for the two models on the 2013 slot-filling task. As observed in earlier results, the LSTM achieves higher recall at the loss of

CEO	
Dictionary	No Ties
jefe (chief)	CEO
CEO	director (principle)
ejecutivo (executive)	directora (director)
cofundador (co-founder)	firma (firm)
president (chairman)	magnate (tycoon)
headquartered	
Dictionary	No Ties
sede (headquarters)	Geológico (Geological)
situado (located)	Treki (Treki)
selectivo (selective)	Geofísico(geophysical)
profesional (vocational)	Normandía (Normandy)
basándose (based)	emplea (uses)
hubby	
Dictionary	No Ties
matrimonio (marriage)	esposa (wife)
casada (married)	esposo (husband)
esposa (wife)	casada(married)
casó (married)	embarazada (pregnant)
embarazada (pregnant)	embarazo (pregnancy)
alias	
Dictionary	No Ties
simplificado (simplified)	Weaver (Weaver)
sabido (known)	interrogación (question)
seudónimo (pseudonym)	alias
privatización (privatization)	reelecto (reelected)
nombre (name)	conocido (known)

Table 5: Example English query words (not in translation dictionary) in bold with their top nearest neighbors by cosine similarity listed for the dictionary and no ties LSTM variants. Dictionary-tied nearest neighbors are consistently more relevant to the query word than untied.

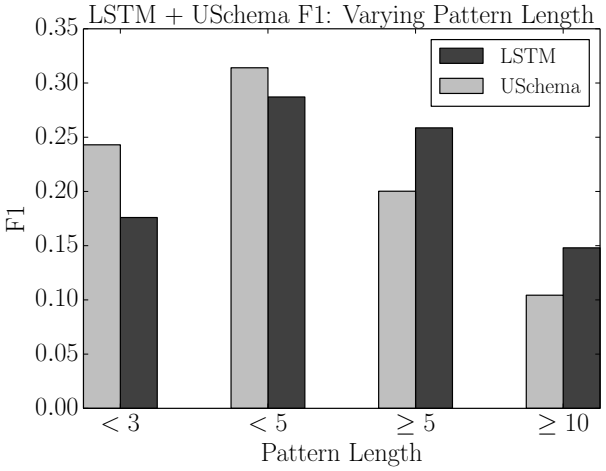


Figure 4: F1 achieved by USchema vs. LSTM models for varying pattern token lengths on 2013 TAC slot-filling. LSTM performs better on longer patterns whereas USchema performs better on shorter patterns.

some precision, whereas USchema can make more precise predictions at a lower threshold for recall. In Figure 4 we observe evidence for these different precision-recall trade-offs: USchema scores higher in terms of F1 on shorter patterns whereas the LSTM scores higher on longer patterns. As one would expect, USchema successfully matches more short patterns than the LSTM, making more precise predictions at the cost of being unable to predict on patterns unseen during training. The LSTM can predict using any text between entities observed at test time, gaining recall at the loss of precision. Combining the two models makes the most of their strengths and weaknesses, leading to the highest overall F1.

Qualitative analysis of our English models also suggests that our encoder-based models (LSTM) extract relations based on a wide range of semantically similar patterns that the pattern-matching model (USchema) is unable to score due to a lack of exact string match in the test data. For example, Table 6 lists three examples of the *per:children* relation that the LSTM finds which USchema does not, as well as three patterns that USchema does find. Though the LSTM patterns are all semantically and syntactically similar, they each contain different specific noun phrases, e.g. *Lori*, *four children*, *toddler daughter*, *Lee and Albert*, etc. Because these specific nouns weren't seen during training, USchema fails to find these patterns whereas the LSTM learns to ignore the specific nouns in favor of the overall pattern, that of a parent-child relationship in an obituary. USchema is limited to finding the relations represented by patterns observed during training, which limits the patterns matched at test-time to short and common patterns; all the USchema patterns matched at test time were similar to those listed in Table 6: variants of *'s son, '*.

LSTM
McGregor *is survived by his wife, Lori, and four children, daughters Jordan,* **Taylor** and Landri, and a son, Logan.
In addition to his wife, **Mays** *is survived by a toddler daughter and a son,* **Billy Mays Jr.**, who is in his 20s.
Anderson *is survived by his wife Carol, sons Lee and Albert, daughter* **Shirley Englebrecht** and nine grandchildren.
USchema
Dio *'s son,* **Dan Padavona**, cautioned the memorial crowd to be screened regularly by a doctor and take care of themselves, something he said his father did not do.
But **Marshall** *'s son,* **Philip**, told a different story.
"I'd rather have Sully doing this than some stranger, or some hotshot trying to be the next Billy Mays," said the guy who actually is the next **Billy Mays***, his son* **Billy Mays III**.

Table 6: Examples of the *per:children* relation discovered by the LSTM and Universal Schema. Entities are bold and patterns italicized. The LSTM models a richer set of patterns

6 Conclusion

By jointly embedding English and Spanish corpora along with a KB, we can train an accurate Spanish relation extraction model using no direct annotation for relations in the Spanish data. This approach has the added benefit of providing significant accuracy improvements for the English model, outperforming the top system on the 2013 TAC KBC slot filling task, without using the hand-coded rules or additional annotations of alternative systems. By using deep sentence encoders, we can perform prediction for arbitrary input text and for entities unseen in training. Sentence encoders also provides opportunities to improve cross-lingual transfer learning by sharing word embeddings across languages. In future work we will apply this model to many more languages and domains besides newswire text. We would also like to avoid the entity detection problem by using a deep architecture to both identify entity mentions and identify relations between them.

Acknowledgments

Many thanks to Arvind Neelakantan for good ideas and discussions. We also appreciate a generous hardware grant from nVidia. This work was supported in part by the Center for Intelligent Information Retrieval, in part by Defense Advanced Research Projects Agency (DARPA) under agreement #FA8750-13-2-0020 and contract #HR0011-15-2-0036, and in part by the National Science Foundation (NSF) grant numbers DMR-1534431, IIS-1514053 and CNS-0958392. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon, in part by DARPA via agreement #DFA8750-13-2-0020 and NSF grant #CNS-0958392. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

References

[Angeli et al.2014] Gabor Angeli, Sonal Gupta, Melvin Jose, Christopher D Manning, Christopher Ré, Julie Tibshirani, Jean Y Wu, Sen Wu, and Ce Zhang. 2014. Stanfords 2014 slot filling systems. *TAC KBP*.

[Banko et al.2007] Michele Banko, Michael J Cafarella, Stephen Soderland, Matt Broadhead, and Oren Etzioni. 2007. Open information extraction from the web. In *International Joint Conference on Artificial Intelligence.*

[Bentor et al.2014] Yinon Bentor, Vidhoon Viswanathan, and Raymond Mooney. 2014. University of texas at austin kbp 2014 slot filling system: Bayesian logic programs for textual inference. In *Proceedings of the Seventh Text Analysis Conference: Knowledge Base Population (TAC 2014)*.

[Bollacker et al.2008] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a

collaboratively created graph database for structuring human knowledge. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*.

[Bordes et al.2013] Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems.*

[Bordes et al.2014] Antoine Bordes, Sumit Chopra, and Jason Weston. 2014. Question answering with subgraph embeddings. *arXiv preprint arXiv:1406.3676*.

[Bunescu and Mooney2007] Razvan Bunescu and Raymond Mooney. 2007. Learning to extract relations from the web using minimal supervision. In *Annual meeting-association for Computational Linguistics*, volume 45, page 576.

[Carlson et al.2010] Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka, and A. 2010. Toward an architecture for never-ending language learning. In *In AAAI*.

[Collobert et al.2011] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537.

[dos Santos et al.2015] Cıcero Nogueira dos Santos, Bing Xiang, and Bowen Zhou. 2015. Classifying relations by ranking with convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, volume 1, pages 626–634.

[Etzioni et al.2008] Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel S Weld. 2008. Open information extraction from the web. *Communications of the ACM*, 51(12):68–74.

[Faruqui and Kumar2015] Manaal Faruqui and Shankar Kumar. 2015. Multilingual open relation extraction using cross-lingual projection. *arXiv preprint arXiv:1503.06450*.

[Faruqui et al.2014] Manaal Faruqui, Jesse Dodge, Sujay K Jauhar, Chris Dyer, Eduard Hovy, and Noah A Smith. 2014. Retrofitting word vectors to semantic lexicons. *arXiv preprint arXiv:1411.4166*.

[García-Durán et al.2015] Alberto García-Durán, Antoine Bordes, Nicolas Usunier, and Yves Grandvalet. 2015. Combining two and three-way embeddings models for link prediction in knowledge bases. *CoRR*, abs/1506.00999.

[Gardner et al.2014] Matt Gardner, Partha Talukdar, Jayant Krishnamurthy, and Tom Mitchell. 2014. Incorporating vector space similarity in random walk inference over knowledge bases. In *Empirical Methods in Natural Language Processing*.

[Gouws et al.2015] Stephan Gouws, Yoshua Bengio, and Greg Corrado. 2015. B IL BOWA : Fast Bilingual Distributed Representations without Word Alignments. *Icml*, pages 1–10.

[Gu et al.2015] Kelvin Gu, John Miller, and Percy Liang. 2015. Traversing knowledge graphs in vector space. *arXiv preprint arXiv:1506.01094*.

[Hermann and Blunsom2014] Karl Moritz Hermann and Phil Blunsom. 2014. Multilingual models for compositional distributed semantics. *arXiv preprint arXiv:1404.4641*.

[Hochreiter and Schmidhuber1997] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. In *Neural Computation.*

[Hoffmann et al.2011] Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke Zettlemoyer, and Daniel S Weld. 2011. Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 541–550. Association for Computational Linguistics.

[Kalchbrenner et al.2014] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, June.

[Kim2014] Yoon Kim. 2014. Convolutional neural networks for sentence classification. *EMNLP*.

[Kingma and Ba2015] Diederik Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference for Learning Representations (ICLR)*.

[Koehn2005] Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5, pages 79–86. Citeseer.

[Lao et al.2011] Ni Lao, Tom Mitchell, and William W. Cohen. 2011. Random walk inference and learning in a large scale knowledge base. In *Conference on Empirical Methods in Natural Language Processing*.

[Lao et al.2012] Ni Lao, Amarnag Subramanya, Fernando Pereira, and William W. Cohen. 2012. Reading the web with learned syntactic-semantic inference rules. In *Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*.

[Larochelle et al.2008] Hugo Larochelle, Dumitru Erhan, and Yoshua Bengio. 2008. Zero-data learning of new tasks. In *National Conference on Artificial Intelligence.*

[Li et al.2015] Jiwei Li, Dan Jurafsky, and Eudard Hovy. 2015. When are tree structures necessary for deep learning of representations? *arXiv preprint arXiv:1503.00185*.

[Lin et al.2014] Hailun Lin, Zeya Zhao, Yantao Jia, Yuanzhuo Wang, Jinhua Xiong, and Xiaojing Li. 2014. OpenKN at TAC KBP 2014.

[Lin et al.2015] Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of AAAI*.

[Luong et al.2015] Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Bilingual word representations with monolingual quality in mind. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 151–159.

[McCallum et al.2009] Andrew McCallum, Karl Schultz, and Sameer Singh. 2009. FACTORIE: Probabilistic programming via imperatively defined factor graphs. In *Neural Information Processing Systems (NIPS)*.

[Mikolov et al.2013] Tomas Mikolov, Quoc V Le, and Ilya Sutskever. 2013. Exploiting Similarities among Languages for Machine Translation. In *arXiv preprint arXiv:1309.4168v1*, pages 1–10.

[Min et al.2013] Bonan Min, Ralph Grishman, Li Wan, Chang Wang, and David Gondek. 2013. Distant supervision for

relation extraction with an incomplete knowledge base. In *HLT-NAACL*, pages 777–782.

[Mintz et al.2009] Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Association for Computational Linguistics and International Joint Conference on Natural Language Processing*.

[Neelakantan et al.2015] Arvind Neelakantan, Benjamin Roth, and Andrew McCallum. 2015. Compositional vector space models for knowledge base completion. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*.

[Nickel et al.2011] Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2011. A three-way model for collective learning on multi-relational data. In *International Conference on Machine Learning*.

[Nickel et al.2015] Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. 2015. A review of relational machine learning for knowledge graphs: From multi-relational link prediction to automated knowledge graph construction. *arXiv preprint arXiv:1503.00759*.

[Rendle et al.2009] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 452–461. AUAI Press.

[Riedel et al.2010] Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *Machine Learning and Knowledge Discovery in Databases*, pages 148–163. Springer.

[Riedel et al.2013] Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M. Marlin. 2013. Relation extraction with matrix factorization and universal schemas. In *HLT-NAACL*.

[Rocktaschel et al.2015] Tim Rocktaschel, Sameer Singh, and Sebastian Riedel. 2015. Injecting logical background knowledge into embeddings for relation extraction. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*.

[Roth et al.2014] Benjamin Roth, Tassilo Barth, Grzegorz Chrupała, Martin Gropp, and Dietrich Klakow. 2014. Relationfactory: A fast, modular and effective system for knowledge base population. *EACL 2014*, page 89.

[Schein et al.2002] Andrew I Schein, Alexandrin Popescul, Lyle H Ungar, and David M Pennock. 2002. Methods and metrics for cold-start recommendations. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 253–260. ACM.

[Socher et al.2013] Richard Socher, Danqi Chen, Christopher D Manning, and Andrew Ng. 2013. Reasoning with neural tensor networks for knowledge base completion. In *Advances in Neural Information Processing Systems*.

[Suchanek et al.2007] Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: A core of semantic knowledge. In *Proceedings of the 16th International Conference on World Wide Web*.

[Surdeanu and Ji.2014] Mihai Surdeanu and Heng Ji. 2014. Overview of the english slot filling track at the tac2014 knowledge base population evaluation. *Proc. Text Analysis Conference (TAC2014)*.

[Surdeanu et al.2012] Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D Manning. 2012. Multi-instance multi-label learning for relation extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 455–465. Association for Computational Linguistics.

[Sutskever et al.2014] Ilya Sutskever, Oriol Vinyals, and Quoc V. V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*.

[Toutanova et al.2015] Kristina Toutanova, Danqi Chen, Patrick Pantel, Hoifung Poon, Pallavi Choudhury, and Michael Gamon. 2015. Representing text for joint embedding of text and knowledge bases. In *Empirical Methods in Natural Language Processing (EMNLP)*.

[Vinyals et al.2014] Oriol Vinyals, Lukasz Kaiser, Terry Koo, Slav Petrov, Ilya Sutskever, and Geoffrey Hinton. 2014. Grammar as a foreign language. In *CoRR*.

[Wang et al.2014] Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, pages 1112–1119. Citeseer.

[Xu et al.2015] Yan Xu, Lili Mou, Ge Li, Yunchuan Chen, Hao Peng, and Zhi Jin. 2015. Classifying relations via long short term memory networks along shortest dependency paths. In *Proceedings of Conference on Empirical Methods in Natural Language Processing (to appear)*.

[Yang et al.2015] Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015. Embedding entities and relations for learning and inference in knowledge bases. *International Conference on Learning Representations 2014*.

[Yao et al.2010] Limin Yao, Sebastian Riedel, and Andrew McCallum. 2010. Collective cross-document relation extraction without labelled data. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1013–1023. Association for Computational Linguistics.

[Yao et al.2013] Limin Yao, Sebastian Riedel, and Andrew McCallum. 2013. Universal schema for entity type prediction. In *Proceedings of the 2013 workshop on Automated knowledge base construction*, pages 79–84. ACM.

[Yates and Etzioni2007] Alexander Yates and Oren Etzioni. 2007. Unsupervised resolution of objects and relations on the web. In *North American Chapter of the Association for Computational Linguistics*.

[Zeng et al.2015] Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. *EMNLP*.