TRANSFER LEARNING FOR MULTILINGUAL KNOWLEDGE BASE CONSTRUCTION

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

While existing embedding approaches have shown success for closed-domain knowledge base construction, they suffer from the cold-start problem, where it is unclear how to form predictions for entities or text unseen in the training data. Addtionally they are typically confined to a single language with available linguistic resources such as parsers. We address both problems, using deep sentence encoders with word and entity embeddings tied across languages. By jointly modeling English and Spanish, we perform zero-shot learning of an effective Spanish model, using no annotation for relations in Spanish text. In fact, using sentence encoders and multilingual modeling provides complementary accuracy improvements for the English model as well. We encourage further application of our method to knowledge base construction in low-resource languages.

1 INTRODUCTION

The goal of automatic knowledge base construction (AKBC) is building a structured knowledge base (KB) (Carlson et al., 2010; Suchanek et al., 2007; Bollacker et al., 2008) of facts using a noisy corpus of raw text evidence, and perhaps an initial seed KB to be augmented. AKBC supports downstream reasoning at a high level about extracted entities and their relations, and thus has broad-reaching applications [[todo: examples]]. One challenge in AKBC is defining a schema of relations between entities. Not only is a fixed schema finite and brittle, but performing AKBC requires alignment of text corpora with the schema which can be challenging in itself.

[[todo: remove this]] The *universal schema* Riedel et al. (2013) approach to AKBC embeds KB relations, entities and text corpora together to model the union of all schemas, as defined implicitly in text and explicitly in a known KB. Universal schema and its extensions [[todo: cite: entity types, chains of reasoning...]] have proven successful for AKBC [[todo: more about this?]].

One drawback to universal schema is that it expresses each relation as a distinct item to be embedded. This harms its ability to generalize, making it impossible to process inputs not precisely seen at training time. We are most interested, however, in generalizing to new text patterns, new entities, and even new languages unseen during training.

The most extreme case of generalization is to a completely new language. Many languages have limited resources or labeled data including a sufficiently complete KB. We would like to extract relations in a new languages without any labeled data, entities appearing in KB, or linguistic resources such as treebanks. We demonstrate that we can accomplish this given only simple knowledge about word-word translations, and unlabeled text having overlapping entities.

This paper presents an LSTM word model that captures textual relations through their compositional semantics, allowing for prediction on new textual relations never seen before. We demonstrate this dramatically by running on a completely new language, with no training or KB overlap with the base language. We embed words from multiple languages to a common space and improve these representations further by tying a small set of dictionary entries across languages.

[[todo: remove this]] Our work differs substantially from recent related work on convolutional neural networks (CNNs) for relation expression modeling (Toutanova et al., 2015) in many ways: (1) they work exclusively in English whereas our main motivation is multi-lingual KBC; (2) their CNN is used only at train time; (3) their evaluation is limited to entity-pair classification whereas we operate on mention-pairs allowing us to generalize to unseen entities; (4) they use syntactic parses between entities where we operate on raw text; and (5) our experiments demonstrate that LSTMs yield better accuracy than CNNs.

We present extensive experiments on the TAC Knowledge Base Population (KBP) slot-filling benchmark in which we perform relation extraction in Spanish with no labeled data or KB overlap. Interestingly, we also find that joint training with Spanish improves English accuracy.

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., 2013a).

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, ie. 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 & 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 form predictions at the corpus level by aggregating information from all sentences containing a given pair of entities before making a prediction. The former approach is often used in practice, due to the simplicity of independently classifying individual sentence and the ease of associating each prediction with a provenance.

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 (open IE) lets the text speak for itself: R contains all possible patterns of surface text occuring between entities s and o (Banko et al., 2007; Etzioni et al., 2008; Yates & Etzioni, 2007). Coupled with a method for filtering and clustering patterns, such an approach offers impressive coverage, avoids issues of distant supervision, and provides a useful exploratory tool. On the other hand, open IE predictions are difficult to use in downstream tasks that expect information from a fixed schema.

Table 1 provide examples of open IE text patterns. The examples in row two and three illustrate relational contexts for which similarity is diffult to be captured by an open IE approach because of their syntactically complex constructions. This motivates the technique in Section 4.

Table 1: Examples of sentences expressing relations. Open IE patterns (italicized) consist of text ocurring between entities (bold) in a sentence. **[[**todo: add extra column for open IE pattern. Also, discuss how this requires rules. instead we use deep learning.**]]**

Relation	Sentence
per:siblings	**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.
per:cities_of_residence	A professor emeritus at Yale, **Mandelbrot** *was born in Poland but as a child moved with his family to* **Paris** where he was educated.
per:cities_of_residence	**Kissel** *was born in Provo, Utah, but her family also lived in* **Reno**.

2.4 UNIVERSAL SCHEMA

[[todo: be more careful about patterns vs. intervening text**]]**

Universal Schema (Riedel et al., 2013) (US) combines the open IE perspective with link-prediction-based relation extraction. This provides predictions from a target schema, while avoiding errors from aligning distant supervision annotation to text. The knowledge graph considered in link-prediction approaches is augmented with additional edges with labels from open IE patterns. Even if the user does not seek to predict these new edges, a joint model over all edges may be able to exploit regularities of the open IE 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:

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

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 open IE pattern or a relation from the target schema. All of the exposition and results in this paper use this factorization, though many of the modeling techniques we present later could be applied easily to the other factorizations described in Riedel et al. (2013). Note that learning unique embeddings for each open IE 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. As with Riedel et al. (2013), all of our experiments employ SGD coupled with 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 US 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 open IE patterns with similar words, and we leverage this approach in Section 4. Additional work has modeled the regularities of multi-hop paths through knowledge graph augmented with text patterns (Lao et al., 2011; 2012; Gardner et al., 2014; Neelakantan et al., 2015).

3 Training a Sentence Classifier without Alignment

Similar to many link prediction approaches, US performs 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 collaborative filtering (Schein et al., 2002): given new data, it is unclear how to form predictions for entity pairs not seen so far, without re-factorizing the entire matrix or applying various heuristics.

In response, this paper re-purposes US as a means to train a sentence-level relation classifier, like those in Section 2.2, which allows us to avoid errors from aligning distant supervision to the corpus. It provides improved accuracy, is more deployable for real-world applications, and provide opportunities in Section 5 to improve multi-lingual AKBC.

We produce predictions using a very simple approach: (1) scan the corpus and extract a large quantity of triples (s, r_{text}, o), where r_{text} is a surface 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 cosince distance between the vectors. In Section A.1, we discuss details for how to make this distance well-defined.

4 Predictions for Unseen Text Patterns

[[todo: make it clear that we're processing the token context words**]]**

The pattern scoring approach described in the previous section is subject to an additional cold start problem: input data may contain patterns unseen in training. This section describes a method for using US to train a relation classifier that can take arbitrary text as input.

Fortunately, the cold start problem for text patterns is more benign than the cold start problem for entities. For text patterns, we can exploit statistical regularities of text: similar patterns should be embedded similarly. Therefore, following (Toutanova et al., 2015), we avoid learning unique text patterns for each open IE pattern, and instead embed patterns compositionally from their word tokens. The modified US likelihood is:

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

Here, if r is an open IE surface pattern, then Encoder(r) is parametrized by a deep architecture applied to the word tokens in the pattern. Otherwise, ie. if r is from the target schema, Encoder(r) is a produced by a lookup table (as in traditional US). Using such an encoder increases the computational cost of test-time prediction from straightforward pattern matching to evaluating a deep architecture. However, encoding can be done in large batches in parallel on a GPU.

Both convolutional networks (CNNs) and recurrent networks (RNNs) are reasonable architectures for the encoder, 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 & Schmidhuber, 1997). These have been very successful in a variety of tasks that require 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, by performing pattern scoring on held out data. On the other hand, Toutanova et al. (2015) adopt the transductive approach. Here, 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 multi-lingual AKBC, and many languages lack linguistic resources such as treebanks. Even parsing non-newswire English text, such as tweets, is extremely challenging.

Prior work has applied deep learning to small-scale relation extraction problem, where functional relationships are detected between common nouns. 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.

4.1 Modeling Idiomatic Patterns

Deep architectures such as CNNs and RNNs are succesful in NLP because most text is compositional: a passage's meaning is defined bottom-up from its constituents. However, some key phrases in language are non-compositional, ie. idiomatic. In particular, we have observed that these include many of the discriminative text patterns useful for AKBC. Table 2 provides some examples of compositional and idiomatic patterns.

[[todo: change this to be common patterns and tail patterns]]

Table 2: Examples of compositional vs. idiomatic text patterns for relation extraction.

compositional	*arg1* will be cremated on Friday just outside the city of *arg2* *arg1* has pleaded not guilty to charges of *arg2* *arg1* was convicted of all five counts against him, including *arg2*
idiomatic	*arg1*, aka *arg2* *arg1* hit the field in *arg2* *arg1* didn't relish the idea of pulling the managerial rug out from under *arg2*

With this in mind, the approach of Riedel et al. (2013) to separately embed each open IE pattern is not an unreasonable modeling technique compared to using text encoder operating on tokens. While the pattern lookup table does not share statistical strength across patterns with similar words, it has the modeling capacity to separately represent discriminative patterns that are short and non-compositional. In addition, many high-precision idiomatic patterns are relatively frequent, and thus their embeddings can be fit well; the parameter sharing of using a single deep architecture as an encoder is unnecessary.

Using pattern embeddings and a using a deep token-based encoder have very different inductive biases. One values specificity, while the other values coverage. In our experiments, we demonstrate that the lookup table approach outperforms using an encoder, but that an ensemble of both models is substantially better than either in isolation.

5 Multilingual Relation Extraction with Zero Annotation

The modeling techniques of the 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 provide modeling techniques for a substantially 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 , requires an available KB of seed facts that have supporting evidence in the corpus. Unfortunately, for many languages available KBs have low overlap with corpora since KBs have cultural and geographical biases. [[todo: Ben: citation?]]

In response, we jointly model relation extraction in a high-resource language, such as English, and an alternative language with no such annotation available. The approach provides transfer learning of a predictive model to the alternative language, and generalizes naturally to modeling of more than two languages.

[[todo: fix this]] Extending the training technique of Section 3 to corpora in multiple languages can be achieved by factorizing a matrix that mixes data from the two corpora and linking the entities across corpora. Without such entity linking, learning degenerates into independent problems for each language. Furthermore, if there is no supervision for the non-English language, then this learning problem will not return a usable sentence classifier. This parameter tying scheme jointly embeds the AKBC problems for the two languages. To form predictions in the low-resource language, we simply apply the pattern scoring approach of Section 3.

[[todo: switch to figure, and change to describe it in text]] In Table 3 we split the entities of a multi-lingual training corpus into sets depending on whether they have annotation in a KB and sets depending on the language of the corpus and the type of annotation. We can perform zero-annotation transfer learning of a relation extractor for the low-resource language if there are entity pairs occuring 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 simply used to bridge the languages during training. In Section 6.2, 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.

Language	Has annotation in the seed KB	No annotation in the seed KB
English	A	B
Low-Resource	C	D

Table 3: Splitting the entities in a multilingul AKBC training set into parts. We only require that sets B and D overlap. Remarkably, we can train a model for the low-resource language even if C is empty.

5.1 Tied Sentence Encoders

The sentence encoder approach of Section 4 is complementary to our multi-lingual modeling technique: we simply use a seperate 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 simple way for tying the parameters of the sentence encoders across languages.

Most work on learning multilingual word embeddings uses aligned sentences from the Europarl dataset (Koehn, 2005) to align words across languages in the embedded space (Gouws et al., 2015; Luong et al., 2015; Hermann & Blunsom, 2014). Others (Mikolov et al., 2013; Faruqui et al., 2014) have aligned seperate single-language embedding models using a word-level alignment dictionary. Notably, Mikolov et al. (2013) use translation pairs to learn a linear transform from one word embedding space to another. We borrow from both of these techniques and use dictionary alignment to tie the joint embedding space beyond entity co-ocurrence by representing word pairs in the dictionary each as a single embedding. Details for our approach are described in Section A.3.

6 Experiments

6.1 Task

Much of the 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 following particular difficulties we address in this work: (1) the cold start problem and (2) evaluation on entities and text unseen during training, and (3) zero-annotation learning of a predictor for a low-resource language.

Instead, we focus on the 2013 TAC KBP slot-filling task. The aim of the TAC KBP 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.

[[todo: Emma: mention what the corpora are and how big they are.]] [[todo: mention best on task (relationfactory)]] [[todo: mention stanford SOA number, and how they use active learning.]] [[todo: all use handwritten patterns.]]

Our retrieval pipeline works by first generating 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. An entity pair qualifies as a candidate prediction if it meets the type criteria for the slot.[1] 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, and heuristically link all entity mentions from our text corpora to a Freebase entity using anchor text in Wikipedia. More details on alignment and linking are described in Appendix A.4.

We evaluate our Spanish models on the 2012 TAC Spanish slot-filling evaluation. However, because this TAC track was never officially run, the coverage of facts in the available annotation is very small. This results in many correct predictions being marked incorrectly as precision errors. In response, we manually annotated all results returned by the 5 models considered in Table 6. Precision and recall are calculated with respect to the union of the TAC annotation and our new labeling.

As discussed in Section 4.1, models using a deep sentence encoder and using a pattern lookup table have complementary strengths and weaknesses. In response, we also present results where we ensemble the outputs of the two models. We manually shift the models' thresholds to be more precision-biased, and take the union of the predictions returned by the two models. This simple heuristic was effective for our application, though future work could explore more sophisticated ensembling methods. Our approach differs from the ensembling technique of Toutanova et al. (2015), who add the confidence scores of the systems and then apply a threshold. We found that this ensembling approach does not adequately account for the qualitative distinction in types of prediction that each technique can make accurately.

Finally, 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.

6.2 RESULTS

[[todo: make tables at top and bottom. make them appear before they are discussed.]] See Section A.6 for a thorough discussion of the hyperparameters, optimization techniques, etc. used in all experiments. As with Riedel et al. (2013), we train our model using the BPR loss of Rendle et al. (2009).

In experiments on the English and Spanish TAC KBC slot-filling tasks, we find that both US and LSTM models outperform the CNN across languages, and that US tends to perform slightly better than the LSTM as the only model. Ensembling the LSTM and US models further increases final F1 scores in all experiments, suggesting that the two different types of model compliment each other well. In Section 6.3 we present a qualitative analysis of our results which further confirms this hypothesis.

Table 4 presents the performance of our English models. First, observe that the LSTM substantially outperforms a CNN. Second, note that the LSTM achieves higher recall than US whereas US is more precision-biased. This confirms our hypothesis in Section 4.1 about the strengths and weaknesses of the two approaches. Unsuprisingly, ensembling the LSTM and US improves F1 by nearly 2 points over the strongest single model, US. Adding the alternative names heuristic described in Section

[1] Note that because of the difficult retrieval and entity detection step, the maximum recall achievable by the prediction component is limited even for perfect predictions. 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. All scores reported in this paper are evaluated in the 'anydoc' (relaxed) scoring, for mitigating penalizing effects for systems not included in the evaluation pool.

A.2 increases F1 by an additional 2 points, resulting in an F1 score that is competitive with the state-of-the-art.

Table 4: Precision, recall and F1 of English-only models on the English TAC 2013 slot-filling task. LSTM+US ensemble outperforms any single model.

Model	Recall	Precision	F1
CNN	28.9	35.9	32.0
LSTM	**34.3**	32.7	33.5
US	29.4	**42.6**	34.8
US+LSTM	32.1	42.6	36.6
US+LSTM+AN	34.4	43.9	**38.6**

In Table 5, we analyze the effect of jointly learning English and Spanish models on English slot filling performance. Adding Spanish data improves scores of US and CNN, though the LSTM remains unaffected. Further tying the parameters of English and Spanish data by adding a translation dictionary further improves the CNN, and greatly improves the ensemble of US and LSTM, leading to 1.5 point increase in F1 over the ensemble of models trained on English alone. The boost in score resulting from dictionary typing suggests that with dictionary-tied parameters the LSTM can better leveraging the Spanish data to find good relations that US is unable to find with only parameter tying through entities. Since US models only embeddings of entire patterns and not single words, parameters cannot be tied at the token-level and so dictionary-tied results are not applicable to this model. As for the English-only results, adding the alternate names dictionary further increases F1 by more than 1.5 points to a final score about one F1 point higher than the English-only ensemble.

Table 5: F1 scores of multilingual models on the English TAC 2013 slot-filling task. Jointly embedding English and Spanish entity pairs results in higher scores on the English evaluation.

Model	En Only	En (+Es)	En (+Es+dict)
CNN	32.0	32.2	32.6
LSTM	33.5	33.5	33.5
US	34.8	35.7	—
US+LSTM	36.6	36.5	38.1
US+LSTM+AN	38.6	38.1	**39.7**

Table 6 presents results for our Spanish relation extractors trained using zero-shot learning. For both the CNN and LSTM, tying word embeddings between the two languages results in substantial improvements. We see that ensembling the non-dictionary LSTM with US leads to a lower score than just US alone, but ensembling the dictionary-tied LSTM with US provides a significant increase of nearly 4 F1 points over the highest-scoring single model, US. Clearly, grounding the Spanish data using a translation dictionary provides much better Spanish word representations. These improvements are complementary to the non-compositional baseline US model, and yield impressive results when ensembled.

Table 6: Zero-shot results (F1 scores) of multilingual models on 2012 Spanish TAC KBP slot-filling task. Adding a translation dictionary improves all compositional models. Ensembling LSTM and US models performs the best.

Model	Es (+En)	Es (+En+dict)
CNN	8.1	12.0
LSTM	8.4	15.3
US	18.6	—
US+LSTM	18.0	**22.4**

6.3 Qualitative Analysis

Qualitative analysis of our English models suggests that our compositional models (LSTM) extract relations based on a wide range of semantically similar patterns that the pattern-matching model (US) is unable to score due to a lack of exact string match in the test data. For example, Table 7 lists three examples of the *per:children* relation that the LSTM finds which US does not, as well as three patterns that US 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, US 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. US 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 US patterns matched at test time were similar to those listed in Table 7: variants of *’s son, ’*.

Table 7: Examples of the *per:children* relation discovered by the LSTM and Universal Schema. Entities are bold and patterns italicized. The LSTM can model a richer set of patterns

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.
US	**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**.

Analysis of our mutlilingual models also suggests that they successfully embed semantically similar relations across languages using tied entity pairs and translation dictionary as grounding. Table 8 lists three top nearest neighbors in English for several Spanish patterns from the text. In each case, the English patterns do capture the relation represented in the Spanish text.

In addition to embedding semantically similar phrases from English and Spanish to have high similarity, our mutlilingual models also learn high-quality multilingual word embeddings. By tying the parameters of words contained in an automatically generated translation dictionary along with entities common between the English and Spanish corpora, our LSTM learns a shared embedding space. In Table 9 we compare Spanish nearest neighbors of English query words learned by the LSTM with dictionary ties versus the LSTM with no ties. Both LSTM models learn to align the Spanish and English tokens using shared entity embeddings. However, the dictionary-tied model learns higher quality multilingual embeddings.

7 Conclusion and Future Work

By using jointly embedding English and Spanish knowledge bases, 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, obtaining nearly state-of-the-art accuracy on the 2013 TAC KBC slot filling task, while using substantially fewer hand-coded rules than 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 cros;ws-lingual transfer learning by sharing word embeddings across languages.

In future work, we will consider using a sentence encoder that considers a larger context window, such as the entire sentence containing two entities or even crossing sentence boundaries to consider multiple sentences within a document or corpus. We are also interested to apply this model to many more languages and more domains besides newswire text. We would also like to avoid the

Table 8: **[[**todo: remove some of these examples**]]**. Top English patterns for a Spanish query pattern encoded using the dictionary LSTM: For each Spanish query (English translation in italics), a list of English nearest neighbors.

arg1 y cuatro de sus familias, incluidos su esposa, Wu Shu-chen, su hijo, *arg2* arg1 *and four of his family members, including his wife, Wu Shu-chen, his son,* arg2
arg1 Bachchan and his son *arg2* *arg1* C. MacKenzie is survived by his wife, Sybil MacKenzie and a son, *arg2* *arg1* Spelling give birth to a baby last week – son *arg2*
arg1 (Puff Daddy, cuyos verdaderos nombre sea *arg2* arg1 *(Puff Daddy, whose real name is* arg2
arg1 (usually credited as *E1* *arg1* (also known as Gero ##, real name *arg2* *arg1* and (after changing his name to *arg2*
arg1, Tian Tian, de ## años de edad, y su madre *arg2* arg1, *Tian Tian, ## years old, and his mother* arg2
arg1 Gyllenhaal's parents – screenwriter Naomi Foner and director *arg2* *arg1* Brando's mother, actress Anna Kashfi, divorced *arg2* *arg1* Cash, his mom was *arg2*
arg1 llegó a la alfombra roja en compañía de su esposa, la actriz Suzy Amis, casi una hora antes que su ex esposa, *arg2* arg1 *arrived on the red carpet with his wife, actress Suzy Amis, nearly an hour before his ex-wife* , arg2
arg1, who may or may not be having twins with husband *arg2* *arg1*, aged twenty, Kirk married *arg2* *arg1* went to elaborate lengths to keep his wedding to former supermodel *arg2*

Table 9: The compositional models jointly embed Spanish and English words into a shared space. Example English query words (not in translation dictionary) are bold and top nearest neighbors by cosine similarity are listed for the dictionary and no ties LSTM variants. Dictionary-tied nearest neighbors are consistently more relevant to the query word than non-tied.

CEO	
Dictionary	No Ties
jefe (chief)	CEO
CEO	director (principle)
ejecutivo (executive)	directora (director)
cofundador (cofounder)	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)
basndose (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 (privatisation)	reelecto (reelected)
nombre (name)	conocido (known)

entity detection problem, by using a deep architecture to both identify entity mentions and identify relations between them.

Acknowledgments

We thank Nvidia for a generous hardware grant. **[[**todo: Darpa**]]**. Many thanks to Arvind Neelakantan for good ideas and discussions.

References

Banko, Michele, Cafarella, Michael J, Soderland, Stephen, Broadhead, Matt, and Etzioni, Oren. Open information extraction from the web. In *International Joint Conference on Artificial Intel-*

ligence., 2007.

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

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

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

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

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

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

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

García-Durán, Alberto, Bordes, Antoine, Usunier, Nicolas, and Grandvalet, Yves. Combining two and three-way embeddings models for link prediction in knowledge bases. *CoRR*, abs/1506.00999, 2015. URL `http://arxiv.org/abs/1506.00999`.

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

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

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

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

Hoffmann, Raphael, Zhang, Congle, Ling, Xiao, Zettlemoyer, Luke, and Weld, Daniel S. 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*, pp. 541–550. Association for Computational Linguistics, 2011.

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

Kim, Yoon. Convolutional neural networks for sentence classification. *EMNLP*, 2014.

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

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

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

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

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

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

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

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

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

Min, Bonan, Grishman, Ralph, Wan, Li, Wang, Chang, and Gondek, David. Distant supervision for relation extraction with an incomplete knowledge base. In *HLT-NAACL*, pp. 777–782, 2013.

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

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

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

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

Norouzi, Mohammad, Mikolov, Tomas, Bengio, Samy, Singer, Yoram, Shlens, Jonathon, Frome, Andrea, Corrado, Greg, and Dean, Jeffrey. Zero-shot learning by convex combination of semantic embeddings. In *International Conference on Learning Representations.*, 2014.

Palatucci, Mark, Pomerleau, Dean, Hinton, Geoffrey, and Mitchell, Tom. Zero-shot learning with semantic output codes. In *Neural Information Processing Systems.*, 2009.

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

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

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

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

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

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

Socher, Richard, Ganjoo, Milind, Manning, Christopher D, and Ng, Andrew. Zero-shot learning through cross-modal transfer. In *Neural Information Processing Systems.*, 2013b.

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

Surdeanu, Mihai, Tibshirani, Julie, Nallapati, Ramesh, and Manning, Christopher D. 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*, pp. 455–465. Association for Computational Linguistics, 2012.

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

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

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

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

Xu, Yan, Mou, Lili, Li, Ge, Chen, Yunchuan, Peng, Hao, and Jin, Zhi. 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)*, 2015.

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

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

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

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

A Appendix

A.1 Details Concerning Cosine Similarity Computation

We measure the similarity between r_{text} and r_{schema} by computing the vectors' cosine similarity. However, such a distance is not well-defined, since the model was trained using inner products between entity vectors and relation vectors, not between two relation vectors. The US likelihood is invariant to invertible transformations of the latent coordinate system, since $\sigma\left(u_{s,o}^\top v_r\right) = \sigma\left((A^\top u_{s,o})^\top A^{-1} v_r\right)$ for any invertible A. When taking inner products between two v terms, however, the implicit A^{-1} terms do not cancel out. We found that this issue can be minimized, and high quality predictive accuracy can be achieved, simply by using sufficient ℓ_2 regularization to avoid implicitly learning an A that substantially stretches the space.

A.2 Data Pre-processing

The TAC 2013 English and Spanish newswire corpora each contain about 1 million newswire documents from 2009–2012. We replace tokens occuring less than 5 times in the corpus with UNK and normalize all digits to # (e.g. Oct-11-1988 becomes Oct-##-####). For each sentence, we then extract all entity pairs and the text between them as open IE surface patterns, ignoring patterns longer than 15 tokens. This results in 48 million English 'relations'. In Section A.5, we describe a technique for normalizing the open IE patterns. We filter out entity pairs that occured less than 10 times in the data and extract the largest connected component in this entity co-ocurrence graph. This is necessary for the baseline US model, as otherwise learning decouples into independent problems per connected component. Though the components are connected when using sentence encoders, we use only a single component to facilitate a fair comparison between modeling approaches. Finally, we add 352,236 distant supervision facts extracted from a combination of Freebase and hand-coded high precision text patterns for a final set of 3,980,164 facts made up of 549,760 unique entity pairs, 1,285,258 unique relations and 62,841 unique tokens. We also report results including an alternate names (AN) heuristic from RelationFactory that uses hand-engineered rules for the 'alternate name' relation.

We perform the same preprocessing on the Spanish data, resulting in 34 million raw surface patterns between entities. We then filter patterns that never occur with an entity pair found in the English data. This yields 860,502 Spanish patterns. Our multilingual model is trained on a combination of these Spanish patterns, the English surface patterns, and the distant supervision data described above. We learn word embeddings for 39,912 unique Spanish word types. After parameter tying for translation pairs (Section 5.1), there are 33,711 additional Spanish words not tied to English.

A.3 Generation of Cross-Lingual Tied Word Types

We follow the same procedure for generating translation pairs as Mikolov et al. (2013). First, we select the top 6000 words occuring in the lowercased Europarl dataset for each language and obtain a Google translation. We then filter duplicates and translations resulting in multi-word phrases. We also remove English past participles (ending in -ed) as we found the Google translation interprets these as adjectives (eg, 'she read the borrowed book' rather than 'she borrowed the book') and much of the relational structure in language we seek to model is captured by verbs. This resulted in 6201 translation pairs that occurred in our text corpus. Though higher quality translation dictionaries would likely improve this technique, our experimental results show that such automatically generated dictionaries perform well.

A.4 Distant supervision and entity linking

Our RelationFactory distant supervision data consists of roughly 2 million training sentences obtained from aligning 360 thousand entity pairs (with existing Freebase relations) to the TAC 2012 corpus. For entity linking, we make use of the fact that most Freebase entries contain a link to the corresponding Wikipedia page, and we heuristically 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.

A.5 Open IE Pattern Normalization

To improve US generalization, our US relations use log-shortened patterns where the middle tokens in patterns longer than five tokens are simplified. For each long pattern we take the first two tokens and last two tokens, and replace all k remaining tokens with the number $\log k$. For example, the pattern **Barack Obama** *is married to a person named* **Michell Obama** would be converted to: **Barack Obama** *is married [1] person named* **Michell Obama**. This shortening performs slightly better than whole patterns. LSTM and CNN variants use the entire sequence of tokens.

A.6 Implementation and Hyperparameters

All models are implemented in Torch[2] and tuned to maximzize F1 on the TAC 2012 slotfilling evaluation. We additionally tune the thresholds of our pattern scorer on a per-relation basis to maximize F1 using the 2012 TAC KBP slot filling evaluation as a validation set. All experiments use 50-dimensional relation and entity pair embeddings. Our CNN is implemented as described in Toutanova et al. (2015), using width-3 convolutions, followed by tanh and max pool layers. The CNN and LSTM both learned 100-dimensional word embeddings, which were randomly initialized. We found that pre-trained word embeddings did not substantially affect the results. Entity pair embeddings for the baseline US model are randomly initialized. For the models with LSTM and CNN text encoders, entity pair embeddings are initialized using vectors from the baseline US model. This performs better than random initialization. We perform ℓ_2 gradient clipping to 1 on all models. Universal Schema uses a batch size of 1024 while the CNN and LSTM use 128. All models are optimized using ADAM (Kingma & Ba, 2015) with $\epsilon = 1e - 8$, $\beta_1 = 0.9$, and $\beta_2 = 0.999$ with a learning rate of .001 for US and .0001 for CNN and LSTM. The CNN and LSTM also use dropout of 0.1 after the embedding layer. All models are trained for a maximum of 15 epochs. We also experimented with bidirectional LSTMs which did not perform better.

A.7 Additional Qualitative Results

Table 10: Top scoring patterns for both Spanish and English given TAC relations.

per:sibling	ES	*arg1*, según petición the primeros ministro, su hermano gemelo *arg2* *arg1*, sea the principal favorito para esto oficina que también ambiciona su hermano *arg2* *arg1*, y su hermano gemelo, the primeros ministro *arg2*
	EN	*arg1*, for whose brother *arg2* *arg1* inherited his brother *arg2* *arg1* on saxophone and brother *arg2*
org:top_members_employees	ES	*arg2*, presidente y director generales the *arg1* *arg2*, presidente of the negocios especializada *arg1* *arg2* (CIA), the director of the entidad, *arg1*
	EN	*arg2*, vice president and policy director of the *arg1* *arg2*, president of the German Soccer *arg1* *arg2*, president of the quasi-official *arg1*
per:alternate_names	ES	*arg1* (como también son sabido para *arg2* *arg2*-cuyos verdaderos nombre sea *arg1* *arg1* también sabido como *arg2*
	EN	*arg1* aka *arg2* *arg1*, who also creates music under the pseudonym *arg2* *arg1* (of Modern Talking fame) aka *arg2*
per:cities_of_residence	ES	*arg1*, poblado dónde vive *arg2* *arg1*, una ciudadano naturalizado american y nacido in *arg2* *arg1*, que vive in *arg2*
	EN	*arg1* was born Jan. # , #### in *arg2* *arg1* was born on Monday in *arg2* *arg1* was born at Keighley in *arg2*

[2] http://torch.ch