

# Relation Schema Induction using Tensor Factorization with Side Information

**Madhav Nimishakavi**

Indian Institute of Science  
Bangalore, India

madhav@csa.iisc.ernet.in

**Uday Singh Saini**

Indian Institute of Science  
Bangalore, India

uday.s.saini@gmail.com

**Partha Talukdar**

Indian Institute of Science  
Bangalore, India

ppt@cds.iisc.ac.in

## Abstract

Given a set of documents from a specific domain (e.g., medical research journals), how do we automatically build a Knowledge Graph (KG) for that domain? Automatic identification of relations and their schemas, i.e., type signature of arguments of relations (e.g., *undergo(Patient, Surgery)*), is an important first step towards this goal. We refer to this problem as *Relation Schema Induction (RSI)*. In this paper, we propose Schema Induction using Coupled Tensor Factorization (SICTF), a novel tensor factorization method for relation schema induction. SICTF factorizes Open Information Extraction (OpenIE) triples extracted from a domain corpus along with additional side information in a principled way to induce relation schemas. To the best of our knowledge, this is the first application of tensor factorization for the RSI problem. Through extensive experiments on multiple real-world datasets, we find that SICTF is not only more accurate than state-of-the-art baselines, but also significantly faster (about 14x faster).

## 1 Introduction

Over the last few years, several techniques to build Knowledge Graphs (KGs) from large unstructured text corpus have been proposed, examples include NELL (Mitchell et al., 2015) and Google Knowledge Vault (Dong et al., 2014). Such KGs consist of millions of entities (e.g., *Oslo, Norway*, etc.),

their types (e.g., *isA(Oslo, City)*, *isA(Norway, Country)*), and relationships among them (e.g., *cityLocatedInCountry(Oslo, Norway)*). These KG construction techniques are called ontology-guided as they require as input list of relations, their schemas (i.e., their type signatures, e.g., *cityLocatedInCountry(City, Country)*), and seed instances of each such relation. Listing of such relations and their schemas are usually prepared by human domain experts.

The reliance on domain expertise poses significant challenges when such ontology-guided KG construction techniques are applied to domains where domain experts are either not available or are too expensive to employ. Even when such a domain expert may be available for a limited time, she may be able to provide only a partial listing of relations and their schemas relevant to that particular domain. Moreover, this expert-mediated model is not scalable when new data in the domain becomes available, bringing with it potential new relations of interest. In order to overcome these challenges, we need automatic techniques which can discover relations and their schemas from unstructured text data itself, without requiring extensive human input. We refer to this problem as *Relation Schema Induction (RSI)*.

In contrast to ontology-guided KG construction techniques mentioned above, Open Information Extraction (OpenIE) techniques (Etzioni et al., 2011) aim to extract surface-level triples from unstructured text. Such OpenIE triples may provide a suitable starting point for the RSI problem. In fact, KB-LDA,

	Target task	Interpretable latent factors?	Can induce relation schema?	Can use NP side info?	Can use relation side info?
Typed RESCAL (Chang et al., 2014a)	Embedding	No	No	Yes	No
Universal Schema (Singh et al., 2015)	Link Prediction	No	No	No	No
KB-LDA (Movshovitz-Attias and Cohen, 2015)	Ontology Induction	Yes	Yes	Yes	No
SICTF (this paper)	Schema Induction	Yes	Yes	Yes	Yes

**Table 1:** Comparison among SICTF (this paper) and other related methods. KB-LDA is the most related prior method which is extensively compared against SICTF in Section 4

a topic modeling-based method for inducing an ontology from SVO (Subject-Verb-Object) triples was recently proposed in (Movshovitz-Attias and Cohen, 2015). We note that ontology induction (Velardi et al., 2013) is a more general problem than RSI, as we are primarily interested in identifying categories and relations from a domain corpus, and not necessarily any hierarchy over them. Nonetheless, KB-LDA maybe used for the RSI problem and we use it as a representative of the state-of-the-art of this area.

Instead of a topic modeling approach, we take a tensor factorization-based approach for RSI in this paper. Tensors are a higher order generalization of matrices and they provide a natural way to represent OpenIE triples. Applying tensor factorization methods over OpenIE triples to identify relation schemas is a natural approach, but one that has not been explored so far. Also, a tensor factorization-based approach presents a flexible and principled way to incorporate various types of side information. Moreover, as we shall see in Section 4, compared to state-of-the-art baselines such as KB-LDA, tensor factorization-based approach results in better and faster solution for the RSI problem. In this paper, we make the following contributions:

- We present Schema Induction using Coupled Tensor Factorization (SICTF), a novel and principled tensor factorization method which jointly factorizes a tensor constructed out of OpenIE triples extracted from a domain corpus, along with various types of additional side information for relation schema induction.
- We compare SICTF against state-of-the-art baseline on various real-world datasets from diverse domains. We observe that SICTF is not only significantly more accurate than such

baselines, but also much faster. For example, SICTF achieves 14x speedup over KB-LDA (Movshovitz-Attias and Cohen, 2015).

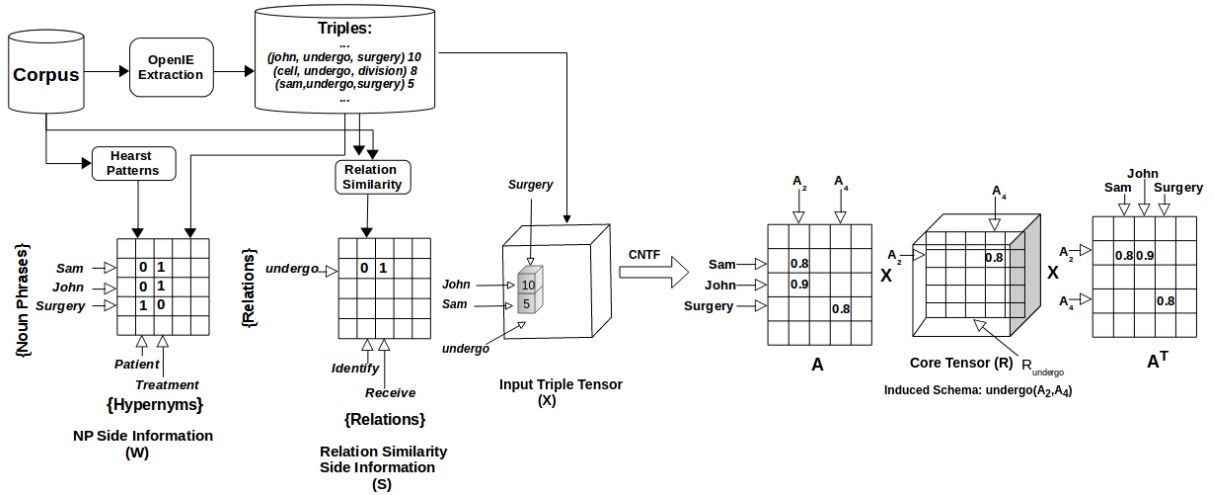
- We have made the data and code available <sup>1</sup>.

## 2 Related Work

**Schema Induction:** Properties of SICTF and other related methods are summarized in Table 1<sup>2</sup>. A method for inducing (binary) relations and the categories they connect was proposed by (Mohamed et al., 2011). However, in that work, categories and their instances were known a-priori. In contrast, in case of SICTF, both categories and relations are to be induced. A method for event schema induction, the task of learning high-level representations of complex events and their entity roles from unlabeled text, was proposed in (Chambers, 2013). This gives the schemas of slots per event, but our goal is to find schemas of relations. (Chen et al., 2013) and (Chen et al., 2015) deal with the problem of finding semantic slots for unsupervised spoken language understanding, but we are interested in finding schemas of relations relevant for a given domain. Methods for link prediction in the Universal Schema setting using matrix and a combination of matrix and tensor factorization are proposed in (Riedel et al., 2013) and (Singh et al., 2015), respectively. Instead of link prediction where relation schemas are assumed to be given, SICTF focuses on discovering such relation schemas. Moreover, in contrast to such

<sup>1</sup><https://github.com/mallabiisc/sictf>

<sup>2</sup>Please note that not all methods mentioned in the table are directly comparable with SICTF, the table only illustrates the differences. KB-LDA is the only method which is directly comparable.



**Figure 1:** Relation Schema Induction (RSI) by SICTF, the proposed method. First, a tensor ( $X$ ) is constructed to represent OpenIE triples extracted from a domain corpus. Noun phrase side information in the form of (noun phrase, hypernym), and relation-relation similarity side information are separately calculated and stored in two separate matrices ( $W$  and  $S$ , respectively). SICTF then performs coupled factorization of the tensor and the two side information matrices to identify relation schemas which are stored in the core tensor ( $R$ ) in the output. Please see Section 3 for details.

methods which assume access to existing KGs, the setting in this paper is unsupervised.

**Tensor Factorization:** Due to their flexibility of representation and effectiveness, tensor factorization methods have seen increased application in Knowledge Graph (KG) related problems over the last few years. Methods for decomposing ontological KGs such as YAGO (Suchanek et al., 2007) were proposed in (Nickel et al., 2012; Chang et al., 2014b; Chang et al., 2014a). In these cases, relation schemas are known in advance, while we are interested in inducing such relation schemas from unstructured text. A PARAFAC (Harshman, 1970) based method for jointly factorizing a matrix and tensor for data fusion was proposed in (Acar et al., 2013). In such cases, the matrix is used to provide auxiliary information (Narita et al., 2012; Erdos and Miettinen, 2013). Similar PARAFAC-based ideas are explored in Rubik (Wang et al., 2015) to factorize structured electronic health records. In contrast to such structured data sources, SICTF aims at inducing relation schemas from unstructured text data. Propstore, a tensor-based model for distributional semantics, a problem different from RSI, was presented in (Goyal et al., 2013). Even though coupled factorization of tensor and matrices constructed out of unstructured text corpus provide a natural and

plausible approach for the RSI problem, they have not yet been explored – we fill this gap in this paper.

**Ontology Induction:** Relation Schema Induction can be considered a sub problem of Ontology Induction (Velardi et al., 2013). Instead of building a full-fledged hierarchy over categories and relations as in ontology induction, we are particularly interested in finding relations and their schemas from unstructured text corpus. We consider KB-LDA<sup>3</sup> (Movshovitz-Attias and Cohen, 2015), a topic-modeling based approach for ontology induction, as a representative of this area. Among all prior work, KB-LDA is most related to SICTF. While both KB-LDA and SICTF make use of noun phrase side information, SICTF is also able to exploit relational side information in a principled manner. In Section 4, through experiments on multiple real-world datasets, we observe that SICTF is not only more accurate than KB-LDA but also significantly faster with a speedup of 14x.

A method for canonicalizing noun and relation phrases in OpenIE triples was recently proposed in (Galárraga et al., 2014). The main focus of this approach is to cluster lexical variants of a *single* entity or relation. This is not directly relevant for RSI, as

<sup>3</sup>In this paper, whenever we refer to KB-LDA, we only refer to the part of it that learns relations from unstructured data.

we are interested in grouping *multiple* entities of the same type into one cluster, and use that to induce relation schema.

### 3 Our Approach: Schema Induction using Coupled Tensor Factorization (SICTF)

#### 3.1 Overview

SICTF poses the relation schema induction problem as a coupled factorization of a tensor along with matrices containing relevant side information. Overall architecture of the SICTF system is presented in Figure 1. First, a tensor  $X \in \mathbb{R}_+^{n \times n \times m}$  is constructed to store OpenIE triples and their scores extracted from the text corpus<sup>4</sup>. Here,  $n$  and  $m$  represent the number of NPs and relation phrases, respectively. Following (Movshovitz-Attias and Cohen, 2015), SICTF makes use of noun phrase (NP) side information in the form of (noun phrase, hypernym). Additionally, SICTF also exploits relation-relation similarity side information. These two side information are stored in matrices  $W \in \{0, 1\}^{n \times h}$  and  $S \in \{0, 1\}^{m \times m}$ , where  $h$  is the number of hypernyms extracted from the corpus. SICTF then performs collective non-negative factorization over  $X$ ,  $W$ , and  $S$  to output matrix  $A \in \mathbb{R}_+^{n \times c}$  and the core tensor  $R \in \mathbb{R}_+^{c \times c \times m}$ . Each row in  $A$  corresponds to an NP, while each column corresponds to an induced category (latent factor). For brevity, we shall refer to the induced category corresponding to the  $q^{th}$  column of  $A$  as  $A_q$ . Each entry  $A_{pq}$  in the output matrix provides a membership score for NP  $p$  in induced category  $A_q$ . Please note that each induced category is represented using the NPs participating in it, with the NPs ranked by their membership scores in the induced category. In Figure 1,  $A_2 = [(John, 0.9), (Sam, 0.8), \dots]$  is an induced category.

Each slice of the core tensor  $R$  is a matrix which corresponds to a specific relation, e.g., the matrix  $R_{undergo}$  highlighted in Figure 1 corresponds to the relation *undergo*. Each cell in this matrix corresponds to an induced schema connecting two induced categories (two columns of the  $A$  matrix), with the cell value representing model’s score of the induced schema. For example, in Figure 1,  $undergo(A_2, A_4)$  is an induced relation schema with

<sup>4</sup> $\mathbb{R}_+$  is the set of non-negative reals.

MEDLINE
(hypertension, disease), (hypertension, state), (hypertension, disorder), (neutrophil, blood element), (neutrophil, effector cell), (neutrophil, cell type)
StackOverflow
(image, resource), (image, content), (image, file), (perl, language), (perl, script), (perl, programs)

**Table 2:** Noun Phrase (NP) side information in the form of (Noun Phrase, Hypernym) pairs extracted using Hearst patterns from two different datasets. Please see Section 3.2 for details.

MEDLINE	StackOverflow
(evaluate, analyze), (evaluate, examine), (indicate, confirm), (indicate, suggest)	(provides, confirms), (provides, offers), (allows, lets), (allows, enables)

**Table 3:** Examples of relation similarity side information in the form of automatically identified similar relation pairs. Please see Section 3.2 for details.

score 0.8 involving relation *undergo* and induced categories  $A_2$  and  $A_4$ .

In Section 3.2, we present details of the side information used by SICTF, and then in Section 3.3 present details of the optimization problem solved by SICTF.

#### 3.2 Side Information

- **Noun Phrase Side Information:** Through this type of side information, we would like to capture type information of as many noun phrases (NPs) as possible. We apply Hearst patterns (Hearst, 1992), e.g., "*<Hypernym> such as <NP>*", over the corpus to extract such (NP, Hypernym) pairs. Please note that neither hypernyms nor NPs are pre-specified, and they are all extracted from the data by the patterns. Examples of a few such pairs extracted from two different datasets are shown in Table 2. These extracted tuples are stored in a matrix  $W_{n \times h}$  whose rows correspond to NPs and columns correspond to extracted hypernyms. We define,

$$W_{ij} = \begin{cases} 1, & \text{if NP}_i \text{ belongs to Hypernym}_j \\ 0, & \text{otherwise} \end{cases}$$

Please note that we don’t expect  $W$  to be a fully specified matrix, i.e., we don’t assume that we know all possible hypernyms for a given NP.

- **Relation Side Information:** In addition to the side information involving NPs, we would also

like to take prior knowledge about textual relations into account during factorization. For example, if we know two relations to be similar to one another, then we also expect their induced schemas to be similar as well. Consider the following sentences "Mary purchased a stuffed animal toy." and "Janet bought a toy car for her son.". From these we can say that both relations *purchase* and *buy* have the schema (Person, Item). Even if one of these relations is more abundant than the other in the corpus, we still want to learn similar schemata for both the relations. As mentioned before,  $S \in \mathbb{R}_+^{m \times m}$  is the relation similarity matrix, where  $m$  is the number of textual relations. We define,

$$S_{ij} = \begin{cases} 1, & \text{if Similarity(Rel}_i, \text{Rel}_j) \geq \gamma \\ 0, & \text{otherwise} \end{cases}$$

where  $\gamma$  is a threshold<sup>5</sup>. For the experiments in this paper, we use cosine similarity over word2vec (Mikolov et al., 2013) vector representations of the relational phrases. Examples of a few similar relation pairs are shown in Table 3.

### 3.3 SICTF Model Details

SICTF performs coupled non-negative factorization of the input triple tensor  $X_{n \times n \times m}$  along with the two side information matrices  $W_{n \times h}$  and  $S_{m \times m}$  by solving the following optimization problem.

$$\min_{A, V, R} \sum_{k=1}^m f(X_k, A, R_k) + f_{np}(W, A, V) + f_{rel}(S, R) \quad (1)$$

where,

$$f(X_k, A, R_k) = \|X_{:, :, k} - AR_{:, :, k}A^T\|_F^2 + \lambda_R \|R_{:, :, k}\|_F^2$$

$$f_{np}(W, A, V) = \lambda_{np} \|W - AV\|_F^2 + \lambda_A \|A\|_F^2 + \lambda_V \|V\|_F^2$$

$$f_{rel}(S, R) = \lambda_{rel} \sum_{i=1}^m \sum_{j=1}^m S_{ij} \|R_{:, :, i} - R_{:, :, j}\|_F^2$$

$$A_{i,j} \geq 0, V_{j,r} \geq 0, R_{p,q,k} \geq 0 \quad (\text{non negative})$$

$$\forall 1 \leq i \leq n, 1 \leq r \leq h,$$

$$1 \leq j, p, q \leq c, 1 \leq k \leq m$$

<sup>5</sup>For the experiments in this paper, we set  $\gamma = 0.7$ , a relatively high value, to focus on highly similar relations and thereby justifying the binary  $S$  matrix.

In the objective above, the first term  $f(X_k, A, R_k)$  minimizes reconstruction error for the  $k^{th}$  relation, with additional regularization on the  $R_{:, :, k}$  matrix<sup>6</sup>. The second term,  $f_{np}(W, A, V)$ , factorizes the NP side information matrix  $W_{n \times h}$  into two matrices  $A_{n \times c}$  and  $V_{c \times h}$ , where  $c$  is the number of induced categories. We also enforce  $A$  to be non-negative. Typically, we require  $c \ll h$  to get a lower dimensional embedding of each NP (rows of  $A$ ). Finally, the third term  $f_{rel}(S, R)$  enforces the requirement that two similar relations as given by the matrix  $S$  should have similar signatures (given by the corresponding  $R$  matrix). Additionally, we require  $V$  and  $R$  to be non-negative, as marked by the (non-negative) constraints. In this objective,  $\lambda_R$ ,  $\lambda_{np}$ ,  $\lambda_A$ ,  $\lambda_V$ , and  $\lambda_{rel}$  are all hyper-parameters.

We derive non-negative multiplicative updates for  $A$ ,  $R_k$  and  $V$  following the rules proposed in (Lee and Seung, 2000), which has the following general form:

$$\theta_i = \theta_i \left( \frac{\frac{\partial C(\theta)^-}{\partial \theta_i}}{\frac{\partial C(\theta)^+}{\partial \theta_i}} \right)^\alpha$$

Here  $C(\theta)$  represents the cost function of the non-negative variables  $\theta$  and  $\frac{\partial C(\theta)^-}{\partial \theta_i}$  and  $\frac{\partial C(\theta)^+}{\partial \theta_i}$  are the negative and positive parts of the derivative of  $C(\theta)$  (Mørup et al., 2008). (Lee and Seung, 2000) proved that for  $\alpha = 1$ , the cost function  $C(\theta)$  monotonically decreases with the multiplicative updates<sup>7</sup>.  $C(\theta)$  for SICTF is given in equation (1). The above procedure will give the following updates:

$$A \leftarrow A * \frac{\sum_k (X_k AR_k^T + X_k^T AR_k) + \lambda_{np} WV^T}{A(\tilde{B} + \lambda_A I + \lambda_{np} VV^T)}$$

$$\tilde{B} = \sum_k (R_k A^T AR_k^T + R_k^T A^T AR_k)$$

$$R_k \leftarrow R_k * \frac{A^T X_k A + 2 \lambda_{rel} \sum_{j=1}^m R_j S_{kj}}{A^T AR_k A^T A + \tilde{D}}$$

$$\tilde{D} = 2 \lambda_{rel} R_k \sum_{j=1}^m S_{kj} + \lambda_R R_k$$

$$V \leftarrow V * \frac{\lambda_{np} A^T W}{\lambda_{np} A^T AV + \lambda_V V}$$

<sup>6</sup>For brevity, we also refer to  $R_{:, :, k}$  as  $R_k$ , and similarly  $X_{:, :, k}$  as  $X_k$

<sup>7</sup>We also use  $\alpha = 1$ .

Dataset	# Docs	# Triples
MEDLINE	50,216	2,499
StackOverflow	5.5m	37,439

**Table 4:** Datasets used in the experiments.

In the equations above,  $*$  is the Hadamard or element-wise product<sup>8</sup>. In all our experiments, we find the iterative updates above to converge in about 10-20 iterations.

## 4 Experiments

In this section, we evaluate performance of different methods on the Relation Schema Induction (RSI) task. Specifically, we address the following questions.

- Which method is most effective on the RSI task? (Section 4.3.1)
- How important are the additional side information for RSI? (Section 4.3.2)
- What is the importance of non-negativity in RSI with tensor factorization? (Section 4.3.3)

### 4.1 Experimental Setup

**Datasets:** We used two datasets for the experiments in this paper, they are summarized in Table 4. For MEDLINE dataset, we used Stanford CoreNLP (Manning et al., 2014) for coreference resolution and Open IE v4.0<sup>9</sup> for triple extraction. Triples with Noun Phrases that have Hypernym information were retained. We obtained the StackOverflow triples directly from the authors of (Movshovitz-Attias and Cohen, 2015), which were also prepared using a very similar process. In both datasets, we use corpus frequency of triples for constructing the tensor.

**Side Information:** Seven Hearst patterns such as "*<hyponym> such as <NP>*", "*<NP> or other <hyponym>*" etc., given in (Hearst, 1992) were used to extract NP side information from the MEDLINE documents. NP side information for the StackOverflow dataset was obtained from the authors of (Movshovitz-Attias and Cohen, 2015).

As described in Section 3, word2vec embeddings of the relation phrases were used to extract relation-similarity based side-information. This was done for

<sup>8</sup> $(A * B)_{i,j} = A_{i,j} \times B_{i,j}$

<sup>9</sup>Open IE v4.0: <http://knowitall.github.io/openie/>

both datasets. Cosine similarity threshold of  $\gamma = 0.7$  was used for the experiments in the paper.

Samples of side information used in the experiments are shown in Table 2 and Table 3. A total of 2067 unique NP-hypernym pairs were extracted from MEDLINE data and 16,639 were from StackOverflow data. 25 unique pairs of relation phrases out of 1172 were found to be similar in MEDLINE data, whereas 280 unique pairs of relation phrases out of approximately 3200 were found similar in StackOverflow data.

Hyperparameters were tuned using grid search and the set which gives minimum reconstruction error for both  $X$  and  $W$  was chosen. We set  $\lambda_{np} = \lambda_{rel} = 100$  for StackOverflow, and  $\lambda_{np} = 0.05$  and  $\lambda_{rel} = 0.001$  for Medline and we use  $c = 50$  for our experiments. Please note that our setting is unsupervised, and hence there is no separate train, dev and test sets.

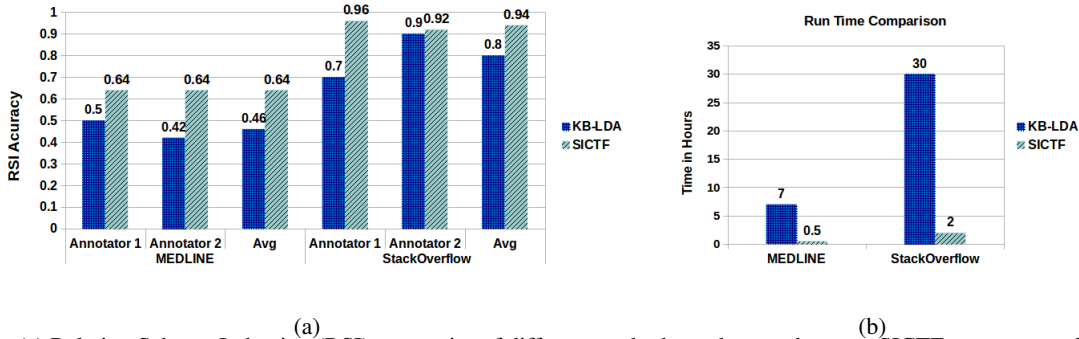
### 4.2 Evaluation Protocol

In this section, we shall describe how the induced schemas are presented to human annotators and how final accuracies are calculated. In factorizations produced by SICTF and other ablated versions of SICTF, we first select a few top relations with best reconstruction score. The schemas induced for each selected relation  $k$  is represented by the matrix slice  $R_k$  of the core tensor obtained after factorization (see Section 3). From each such matrix, we identify the indices  $(i, j)$  with highest values. The indices  $i$  and  $j$  select columns of the matrix  $A$ . A few top ranking NPs from the columns  $A_i$  and  $A_j$  along with the relation  $k$  are presented to the human annotator, who then evaluates whether the tuple  $\text{Relation}_k(A_i, A_j)$  constitutes a valid schema for relation  $k$ . Examples of a few relation schemas induced by SICTF are presented in Table 5. A human annotator would see the first and second columns of this table and then offer judgment as indicated in the third column of the table. All such judgments across all top-reconstructed relations are aggregated to get the final accuracy score. This evaluation protocol was also used in (Movshovitz-Attias and Cohen, 2015) to measure learned relation accuracy.

All evaluations were blind, i.e., the annotators were not aware of the method that generated the output they were evaluating. Moreover, the anno-

Relation Schema	Top 3 NPs in Induced Categories which were presented to annotators	Annotator Judgment
StackOverflow		
$clicks(A_0, A_1)$	$A_0$ : users, client, person $A_1$ : link, image, item	valid
$refreshes(A_{19}, A_{13})$	$A_{19}$ : browser, window, tab $A_{13}$ : page, activity, app	valid
$can\_parse(A_{41}, A_{17})$	$A_{41}$ : access, permission, ability $A_{17}$ : image file, header file, zip file	invalid
MEDLINE		
$suffer\_from(A_{38}, A_{40})$	$A_{38}$ : patient, first patient, anesthetized patient $A_{40}$ : viral disease, renal disease, von recklin ghausen's disease	valid
$have\_undergo(A_3, A_{37})$	$A_3$ : fifth patient, third patient, sixth patient $A_{37}$ : initial liver biopsy, gun biopsy, lymph node biopsy	valid
$have\_discontinue(A_{41}, A_{20})$	$A_{41}$ : patient, group, no patient $A_{20}$ : endemic area, this area, fiber area	invalid

**Table 5:** Examples of relation schemas induced by SICTF from the StackOverflow and MEDLINE datasets. Top NPs from each of the induced categories, along with human judgment of the induced schema are also shown. See Section 4.3.1 for more details.



**Figure 2:** (a) Relation Schema Induction (RSI) accuracies of different methods on the two datasets. SICTF, our proposed method, significantly outperforms state-of-the-art method KBLDA. This is the main result of the paper. Results for KB-LDA on StackOverflow are directly taken from the paper. Please see Section 4.3.1 for details. (b) Runtime comparison between KB-LDA and SICTF. We observe that SICTF results in 14x speedup over KB-LDA. Please see Section 4.3.1 (Runtime Comparison) for details.

tators are experts in software domain and has high-school level knowledge in medical domain. Though recall is a desirable statistic to measure, it is very challenging to calculate it in our setting due to the non-availability of relation schema annotated text on large scale.

## 4.3 Results

### 4.3.1 Effectiveness of SICTF

Experimental results comparing performance of various methods on the RSI task in the two datasets are presented in Figure 2(a). RSI accuracy is calculated based on the evaluation protocol described in Section 4.2. Performance number of KB-LDA for StackOverflow dataset is taken directly from the (Movshovitz-Attias and Cohen, 2015) paper, we used our implementation of KB-LDA for the MEDLINE dataset. Annotation accuracies from two annotators were averaged to get the final accuracy.

From Figure 2(a), we observe that SICTF outperforms KB-LDA on the RSI task. Please note that the inter-annotator agreement for SICTF is 88% and 97% for MEDLINE and StackOverflow datasets respectively. This is the main result of the paper.

In addition to KB-LDA, we also compared SICTF with PARAFAC, a standard tensor factorization method. PARAFAC induced extremely poor and small number of relation schemas, and hence we didn't consider it any further.

**Runtime comparison:** Runtimes of SICTF and KB-LDA over both datasets are compared in Figure 2(b). From this figure, we find that SICTF is able to achieve a 14x speedup on average over KB-LDA<sup>10</sup>. In other words, SICTF is not only able to

<sup>10</sup>Runtime of KB-LDA over the StackOverflow dataset was obtained from the authors of (Movshovitz-Attias and Cohen, 2015) through personal communication. Our own implementation also resulted in similar runtime over this dataset.

Ablation	MEDLINE			StackOverflow		
	A1	A2	Avg	A1	A2	Avg
SICTF	0.64	0.64	<b>0.64</b>	0.96	0.92	<b>0.94</b>
SICTF ( $\lambda_{rel} = 0$ )	0.60	0.56	0.58	0.83	0.70	0.77
SICTF ( $\lambda_{np} = 0$ )	0.46	0.40	0.43	0.89	0.90	0.90
SICTF ( $\lambda_{rel}=0, \lambda_{np} = 0$ )	0.46	0.50	0.48	0.84	0.33	0.59
SICTF ( $\lambda_{rel}=0, \lambda_{np} = 0$ , and no non-negativity constraints )	0.14	0.10	0.12	0.20	0.14	0.17

**Table 6:** RSI accuracy comparison of SICTF with its ablated versions when no relation side information is used ( $\lambda_{rel} = 0$ ), when no NP side information is used ( $\lambda_{np} = 0$ ), when no side information of any kind is used ( $\lambda_{rel} = 0, \lambda_{np} = 0$ ), and when additionally there are no non-negative constraints. From this, we observe that additional side information improves performance, validating one of the central thesis of this paper. Please see Section 4.3.2 and Section 4.3.3 for details.

induce better relation schemas, but also do so at a significantly faster speed.

### 4.3.2 Importance of Side Information

One of the central hypothesis of our approach is that coupled factorization through additional side information should result in better relation schema induction. In order to evaluate this thesis further, we compare performance of SICTF with its ablated versions: (1) SICTF ( $\lambda_{rel} = 0$ ), which corresponds to the setting when no relation side information is used, (2) SICTF ( $\lambda_{np} = 0$ ), which corresponds to the setting when no noun phrases side information is used, and (3) SICTF ( $\lambda_{rel} = 0, \lambda_{np} = 0$ ), which corresponds to the setting when no side information of any kind is used. Hyperparameters are separately tuned for the variants of SICTF. Results are presented in the first four rows of Table 6. From this, we observe that additional coupling through the side information significantly helps improve SICTF performance. This further validates the central thesis of our paper.

### 4.3.3 Importance of Non-Negativity on Relation Schema Induction

In the last row of Table 6, we also present an ablated version of SICTF when no side information no non-negativity constraints are used. Comparing the last two rows of this table, we observe that non-negativity constraints over the  $A$  matrix and core tensor  $R$  result in significant improvement in performance. We note that the last row in Table 6 is equivalent to RESCAL (Nickel et al., 2011) and the fourth row is equivalent to Non-Negative RESCAL (Krompaß et al., 2013), two tensor factor-

ization techniques. We also note that none of these tensor factorization techniques have been previously used for the relation schema induction problem.

The reason for this improved performance may be explained by the fact that absence of non-negativity constraint results in an under constrained factorization problem where the model often overgenerates incorrect triples, and then compensates for this over-generation by using negative latent factor weights. In contrast, imposition of non-negativity constraints restricts the model further forcing it to commit to specific semantics of the latent factors in  $A$ . This improved interpretability also results in better RSI accuracy as we have seen above. Similar benefits of non-negativity on interpretability have also been observed in matrix factorization (Murphy et al., 2012).

## 5 Conclusion

Relation Schema Induction (RSI) is an important first step towards building a Knowledge Graph (KG) out of text corpus from a given domain. While human domain experts have traditionally prepared listing of relations and their schemas, this expert-mediated model poses significant challenges in terms of scalability and coverage. In order to overcome these challenges, in this paper, we present SICTF, a novel non-negative coupled tensor factorization method for relation schema induction. SICTF is flexible enough to incorporate various types of side information during factorization. Through extensive experiments on real-world datasets, we find that SICTF is not only more accurate but also significantly faster (about 14x speedup) compared to state-of-the-art baselines. As part of future work, we hope to analyze SICTF further, as-



sign labels to induced categories, and also apply the model to more domains.

## Acknowledgement

Thanks to the members of MALL Lab, IISc who read our drafts and gave valuable feedback and we also thank the reviewers for their constructive reviews. This research has been supported in part by Bosch Engineering and Business Solutions and Google.

## References

- Evrin Acar, Morten Arendt Rasmussen, Francesco Savarini, Tormod Ns, and Rasmus Bro. 2013. Understanding data fusion within the framework of coupled matrix and tensor factorizations. *Chemometrics and Intelligent Laboratory Systems*, 129(Complete):53–63.
- Nathanael Chambers. 2013. Event schema induction with a probabilistic entity-driven model. In *EMNLP*, pages 1797–1807. ACL.
- Kai-Wei Chang, Wen tau Yih, Bishan Yang, and Christopher Meek. 2014a. Typed tensor decomposition of knowledge bases for relation extraction. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. ACL Association for Computational Linguistics, October.
- Kai-Wei Chang, Wen-tau Yih, Bishan Yang, and Christopher Meek. 2014b. Typed tensor decomposition of knowledge bases for relation extraction. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1568–1579.
- Yun-Nung Chen, William Y. Wang, and Alexander I. Rudnicky. 2013. Unsupervised induction and filling of semantic slots for spoken dialogue systems using frame-semantic parsing. In *2013 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 120–125. IEEE.
- Yun-Nung Chen, William Yang Wang, Anatole Gershan, and Alexander I. Rudnicky. 2015. Matrix factorization with knowledge graph propagation for unsupervised spoken language understanding. In *ACL (1)*, pages 483–494. The Association for Computer Linguistics.
- Xin Dong, Evgeniy Gabrilovich, Jeremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmman, Shaohua Sun, and Wei Zhang. 2014. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 601–610. ACM.
- Dora Erdos and Pauli Miettinen. 2013. Discovering facts with boolean tensor tucker decomposition. In *Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management, CIKM '13*, pages 1569–1572, New York, NY, USA. ACM.
- Oren Etzioni, Anthony Fader, Janara Christensen, Stephen Soderland, and Mausam Mausam. 2011. Open information extraction: The second generation. In *IJCAI*, volume 11, pages 3–10.
- Luis Galárraga, Jeremy Heitz, Kevin Murphy, and Fabian Suchanek. 2014. Canonicalizing Open Knowledge Bases. CIKM.
- Kartik Goyal, Sujay Kumar, Jauhar Huiying, Li Mrinmaya, Sachan Shashank, and Srivastava Eduard Hovy. 2013. A structured distributional semantic model: Integrating structure with semantics.
- R. A. Harshman. 1970. Foundations of the PARAFAC procedure: Models and conditions for an “explanatory” multi-modal factor analysis. *UCLA Working Papers in Phonetics*, 16(1):84.
- Marti A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *In Proceedings of the 14th International Conference on Computational Linguistics*, pages 539–545.
- Denis Krompaß, Maximilian Nickel, Xueyan Jiang, and Volker Tresp. 2013. Non-negative tensor factorization with rescal. *Tensor Methods for Machine Learning, ECML workshop*.
- Daniel D. Lee and H. Sebastian Seung. 2000. Algorithms for non-negative matrix factorization. In *In NIPS*, pages 556–562. MIT Press.
- Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60.
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.
- T. Mitchell, W. Cohen, E. Hruschka, P. Talukdar, J. Betteridge, A. Carlson, B. Dalvi, M. Gardner, B. Kisiel, J. Krishnamurthy, N. Lao, K. Mazaitis, T. Mohamed, N. Nakashole, E. Platanios, A. Ritter, M. Samadi, B. Settles, R. Wang, D. Wijaya, A. Gupta, X. Chen, A. Saparov, M. Greaves, and J. Welling. 2015. Never-ending learning. In *Proceedings of AAAI*.
- Tahir P. Mohamed, Estevam R. Hruschka, Jr., and Tom M. Mitchell. 2011. Discovering relations be-

- tween noun categories. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1447–1455, Stroudsburg, PA, USA. Association for Computational Linguistics.
- M. Mørup, L. K. Hansen, and S. M. Arnfred. 2008. Algorithms for sparse non-negative TUCKER. *Neural Computation*, 20(8):2112–2131, aug.
- Dana Movshovitz-Attias and William W. Cohen. 2015. Kb-lda: Jointly learning a knowledge base of hierarchy, relations, and facts. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.
- Brian Murphy, Partha Pratim Talukdar, and Tom M Mitchell. 2012. Learning effective and interpretable semantic models using non-negative sparse embedding. In *COLING*, pages 1933–1950.
- Atsuhiko Narita, Kohei Hayashi, Ryota Tomioka, and Hisashi Kashima. 2012. Tensor factorization using auxiliary information. *Data Mining and Knowledge Discovery*, 25(2):298–324.
- Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2011. A three-way model for collective learning on multi-relational data. In Lise Getoor and Tobias Scheffer, editors, *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, ICML '11, pages 809–816, New York, NY, USA, June. ACM.
- Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2012. Factorizing yago: Scalable machine learning for linked data. In *Proceedings of the 21st International Conference on World Wide Web, WWW '12*, pages 271–280, New York, NY, USA. ACM.
- Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M. Marlin. 2013. Relation extraction with matrix factorization and universal schemas. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*, pages 74–84.
- Sameer Singh, Tim Rocktäschel, and Sebastian Riedel. 2015. Towards Combined Matrix and Tensor Factorization for Universal Schema Relation Extraction. In *NAACL Workshop on Vector Space Modeling for NLP (VSM)*.
- Fabian M Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: a core of semantic knowledge. In *Proceedings of WWW*.
- Paola Velardi, Stefano Faralli, and Roberto Navigli. 2013. Ontolearn reloaded: A graph-based algorithm for taxonomy induction. *Computational Linguistics*, 39(3):665–707.
- Yichen Wang, Robert Chen, Joydeep Ghosh, Joshua C. Denny, Abel N. Kho, You Chen, Bradley A. Malin, and Jimeng Sun. 2015. Rubik: Knowledge guided tensor factorization and completion for health data analytics. In Longbing Cao, Chengqi Zhang, Thorsten Joachims, Geoffrey I. Webb, Dragos D. Margineantu, and Graham Williams, editors, *KDD*, pages 1265–1274. ACM.