



## Markov logic for statistical relation extraction



M. D. S. Seneviratne<sup>1,\*</sup>, K. S. D. Fernando<sup>2</sup>, D. D. Karunaratne<sup>1</sup>

<sup>1</sup>University of Colombo School of Computing, Colombo, Sri Lanka

<sup>2</sup>Faculty of Information Technology, University of Moratuwa, Moratuwa, Sri Lanka

### ARTICLE INFO

#### Article history:

Received 15 February 2020

Received in revised form

7 June 2020

Accepted 8 June 2020

#### Keywords:

Entity

Relation

Relation extraction

Language dependencies

Markov logic network

### ABSTRACT

In today's world, the Internet has become a fast and efficient information provider, although the relevancy or accuracy of the information found is not guaranteed. The web itself presents numerous problems in finding a required piece of information, mainly due to its heterogeneous nature. Therefore, extracting information from the web is still a challenging task despite the fact that numerous work has been cited in the literature. Extracting information in the form of entities and relations has been addressed by various techniques such as machine learning, natural language processing, and statistical methods, etc. In this paper, we present a rule-based method which is hybridized by machine learning and statistical techniques for accurate performance in domain-specific relation extraction. The rules are modeled in Markov Logic Network to enable statistical performance. Our results on two test domains show overall high values in precision.

© 2020 The Authors. Published by IASE. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

### 1. Introduction

The web being the latest and fastest information provider, people prefer accessing it for their information needs. However, finding a specific piece of information from a massive collection of web sources is a tedious, time-consuming task for a human being. Information extraction, concept definition from various web sources, and text mining are required processes for identifying categories of entities and relationships. Various tasks such as question answering, populating knowledge bases, generating and populating domain ontology essentially require automatic extraction of information from the web resources in order to avoid the time-consuming tedious process of information searching. We focus on extracting relations for domain ontology construction and population.

A considerable amount of work has been carried out in the area of information extraction at a preliminary stage. Many researchers have exploited machine learning (Aitken, 2002; Ciravenga and Wills, 2003; Craven et al., 2000), pattern matching

(Cimiano and Völker, 2005; Barrière, 2016; Wang et al., 2006), shallow natural language processing (Ciravenga and Wills, 2003; Celjuska and Vargas-Vera, 2004) and statistical methods (Craven et al., 2000; Eom and Zhang, 2004). However, further improvements are required to increase precision while maintaining a good recall. Statistical machine learning methods such as Support Vector Machines (Hong, 2005), Hidden Markov Model (Zhang, 2001), etc. as well as rule-based learning (Aitken, 2002; Ciravenga and Wills, 2003; Celjuska and Vargas-Vera, 2004; Cunningham et al., 2002; Yildiz and Miksch, 2011; Pawar et al., 2016) have also been exploited extensively in research work.

Despite the effort made by the researchers, extracting relations from unstructured text still remains a complicated task that requires new and refined techniques. Since rule-based systems are declarative and easy to comprehend, maintain, and incorporate domain knowledge (Chiticariu et al., 2013), these systems are widely used in information extraction. Some systems use hand-coded rules (Drumond and Girardi, 2009), and some systems use machine learning algorithms to induce rules from training data (Aitken, 2002; Celjuska and Vargas-Vera, 2004). The rules need to be weighed to reflect their strength, which contributes to finding the probability of an extracted relation instance. Therefore, finding the weight of a rule is very necessary for determining the accuracy of the extracted information. However, most of the previous rule-based information extraction systems

\* Corresponding Author.

Email Address: [mdeepika65@gmail.com](mailto:mdeepika65@gmail.com) (M. D. S. Seneviratne)

<https://doi.org/10.21833/ijaas.2020.10.001>

Corresponding author's ORCID profile:

<https://orcid.org/0000-0003-3362-8550>

2313-626X/© 2020 The Authors. Published by IASE.

This is an open access article under the CC BY-NC-ND license

(<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

either lack a weight learning process (Aitken, 2002; Ciravenga and Wills, 2003; Yildiz and Miksch, 2011; Lima et al., 2019) or employ a poor weight learning method (Celjuska and Vargas-Vera, 2004; Mooney and Nahm, 2002; Drumond and Girardi, 2009). This implies the necessity of a proper weight learning process in rule-based systems. Markov Logic that accomplishes weight learning for first-order formulas can be investigated for the possibility of weight learning in rule-based information extraction systems and hence for statistical relation extraction.

Current research focuses on information extraction that has turned into distant supervision based techniques that use entities and relations in a large knowledge base such as freebase to annotate the sentences in a text corpus. The problem with this kind of annotation is that some sentences which contain the two entities do not give the expected relation. They might indicate a different relation or might not give a relation at all. Pairwise convolutional neural network (PCNN) is used in most distant supervised based research work (Ji et al., 2017; Zeng et al., 2018) to represent sentences in order to select them for training a text corpus to make the full use of supervision information.

Although statistical machine learning has become the choice of many recent academic researchers in information extraction, rule-based methods find higher applicability in the practical environment and dominate the commercial world (Chiticariu et al., 2013). Therefore, we create a hybridized rule-based system with statistical machine learning in a semi-supervised manner to extract relations from unstructured text. Our system uses Inductive logic programming to generate rules from language dependencies and Markov Logic Network (MLN) to learn the weights for the rules during the statistical relation extraction process. The dependencies between the constituents in a sentence can be used effectively to identify the relationships existing within the sentence. Here we use these language characteristics effectively in relation-extraction-rules to achieve high precision, and each rule is evaluated by the certainty of the extracted relation instances.

## 2. Related works

Relation extraction has been performed by statistical machine learning, rule-based techniques, and natural language processing over the years. Many supervised learning systems induce extraction rules based on identified patterns in the training data set. In identifying patterns, the learning algorithms use the features of labeled data and their neighborhood words. Therefore, the successful application of extraction rules depends on the identification of appropriate neighborhood and features of the language tokens.

PubMiner (Eom and Zhang, 2004), which has been developed to extract entities and relationships from massive biological literature, extracts verbs from a sentence as events and finds the binary

relation between two name entities identified in the sentence. Although PubMiner is capable of extracting both entities and relationships, it also extracts many false positives in the domain. In finding relations, PubMiner heavily depends on external resources such as public medical databases and treasures.

Wang et al. (2006) have addressed hierarchical relation extraction using SVM based approach. They have experimented on Doddington et al. (2004) training data, which defines a hierarchy of relations with 7 top types and 22 subtypes. These relation types include the most commonly used general relationships. Pawar et al. (2016) have also used rules involving three maximum entropy classifiers based on entity features to predict entity types and relation types. Their work is also based on Doddington et al. (2004) data set, and therefore the entity types and relation types are restricted to Doddington et al. (2004) entity and relation types except for NONE and NULL types. Weight learning methods were not used, although rules are modeled in the Markov logic network. Weights of the rules are determined by weight assignment strategies based on odd log ratio and constant multiplier. Drumond and Girardi (2009) also modeled a set of rules in Markov Logic Network and rules wrap extracted terms based on tf-idf measure in first-order logic. Three hand-coded rules, including one rule using language dependencies, infer a term into a concept. They use dependencies to identify a concept based on the fact that two terms bearing the same syntactic dependencies with the same term denote the same concept. This method can cluster textural surface forms that have similar meaning using latent relation models, but clustering does not provide reliable implications in generalization. Yao et al. (2011) used features from the dependency path between entity mentions in a generative probabilistic model in their unsupervised approach for relation extraction.

Recent neural network-based clinical relation extraction systems (Li et al., 2019; Ningthoujam et al., 2019) exploit shortest path dependency (SPD) between target entities. Li et al. (2019) used SPD information features concatenated with word embedding of sentence sequences and demonstrated better performance with SPD, whereas Ningthoujam et al. (2019) system relied only on the SDP. In our approach, we process the dependencies of a sentence to eliminate the unwanted dependencies with respect to the relationship between the target entities. Then in the rule generation, process rules are created from the shortest dependency path between the target entities.

Open Information Extraction (IE) systems are employed to capture all types of relations present in text documents. Open IE systems extract relational tuples from text without requiring predefined vocabulary by identifying relation phrases and associated arguments in arbitrary sentences (Schmitz et al., 2012). REVERB (Fader et al., 2011; Etzioni et al., 2011) an WOE (Wu and Weld, 2010) are such state-of-the-art open IE systems. OLLIE (Schmitz et al., 2012) is an improved open IE system

developed to address the weaknesses of REVERB and WOE. One of the weaknesses pointed by OLLIE is the extraction of relations, which are not asserted as factual in the sentence. It is addressed by our system, as explained in section 3.2, although there are limitations. Another weakness of the state-of-the-art open IE system is the extraction of relations mediated only by verbs. Although we also focus on identifying the related verb in the sentence, some relations such as *has\_characteristics*, which is usable in a wide range of domains, are grasped by the adjectives in our system.

Carlson et al. (2010) employed a semi-supervised approach by using seed instances and patterns on a large corpus to extract more instances and patterns for concepts and relations. They try to reduce the effect of semantic drift by ranking the extracted instances and newly found patterns to select information for further extraction. We also use a semi-supervised method where the training corpus is updated with selected relation instances extracted in the process.

Novel approach in Mintz et al. (2009) work is the distance supervision method. In their method, they use relations in freebase to extract lexical and syntactic features for feature vectors of the multiclass logistic classifier. Aljamel et al. (2015) used distant supervision machine learning techniques for domain-specific relation extraction. They use GATE to extract entities and identify features to use in machine learning for relation extraction. The distant supervision methods have limitations mentioned above in section 1, and they can lead to false positives in automatically annotating a training corpus. Naturally, the predictions of distant supervised methods are subjective, depending on the availability of domain-specific information in the knowledge base.

### 3. Statistical relation extraction

We use inductive logic programming (ILP) to generate relation-extraction-rules from dependency clauses (Seneviratne and Ranasinghe, 2011). Therefore, language dependencies are preprocessed in order to reduce them by filtering irrelevant clauses out and collapsing nouns to form entities (Seneviratne and Ranasinghe, 2014). In extracting relations from sentences, we use conditional statements present in sentences for correct identification of relation instances. We use the Markov Logic Network (Domingos and Richardson, 2006) to model the relation-extraction-rules to find rule weights for statistical relation extraction.

#### 3.1. Generation of relation extraction rules

A verb is the powerful lexical term which binds two adjacent syntactic categories, and a relation can be defined as a predicate expression of two nouns, i.e., subject and object wrapped in syntactic categories as follows:

*Verb (Subject, Object)*

or,

*Verb\_Prep (Subject, Object)*

*Verb\_prep* is the form of the verb combined with a preposition. Therefore, a relation can be identified by verb constituent of the sentence. When there is more than one verb present in a sentence, the most suitable verb should be identified as a related verb.

The process of rule generation employs the inductive logic programming (ILP) technique (Dzeroski and Lavrac, 1994) in the learning algorithm to derive a set of rules from the dependencies based on the text annotated with the entities. The text processing tool GATE (Cunningham et al., 2002) is used to identify entities and sentences annotated with entities that are parsed by Stanford parser producing dependencies. The rule learning process uses the output of the Stanford parser to learn rules to extract relation instances for a known relation such as *located\_in*, *part\_of*, *feed\_on*, etc., and some of which are domain-specific relations.

For example, the sentence *s(1)*:

- *The diet of **Ostrich** mainly consists of **plant matter**, though it eats **insects*** *s(1)*

gives the relation instances:

*feed\_on (Ostrich, plant\_matter)*  
*feed\_on (Ostrich, insects)*

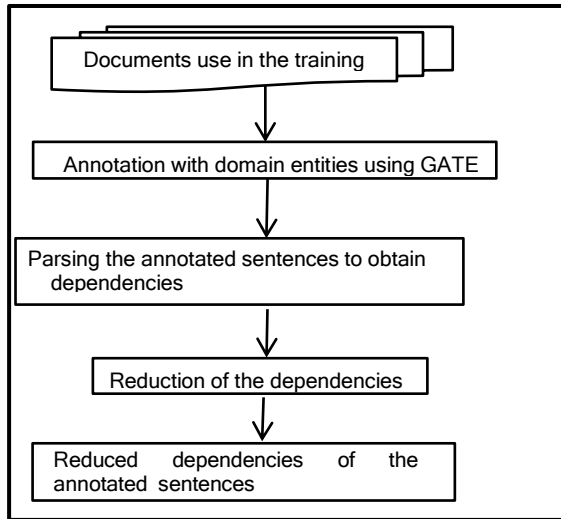
The example sentence gives the following Stanford collapsed Type dependencies:

*det (diet-2, The-1)*  
*nsubj (consists-6, diet-2)*  
*prep\_of (diet-2, **Ostrich**-4)*  
*advmod (consists-6, mainly-5)*  
*nn (matter-9, **plant**-8)*  
*prep\_of (consists-6, matter-9)*  
*mark (eats-13, though-11)*  
*nsubj (eats-13, it-12)*  
*advcl (consists-6, eats-13)*  
*dojb (eats-13, **insects**-14)*

The reduced dependencies are:

*nsubj (consists-6, diet\_of\_**Ostrich**-4)*  
*prep\_of (consists-6, **plant\_matter**-9)*  
*nsubj (eats-13, it-12)*  
*dojb (eats-13, **insects**-14)*

The reduced dependencies of the sentences which contain positive or negative instances for relation and syntactic tags provide background knowledge to learn relation-extraction-rules. At the end of the process, we have a set of rules for each relation considered. Initially, we consider a set of predefined relations, but during the extraction process, new relations can also be identified. Fig. 1 shows the preprocessing sequence of the training documents, and Fig. 2 shows the input and output of both rule learning and relation extraction processes.



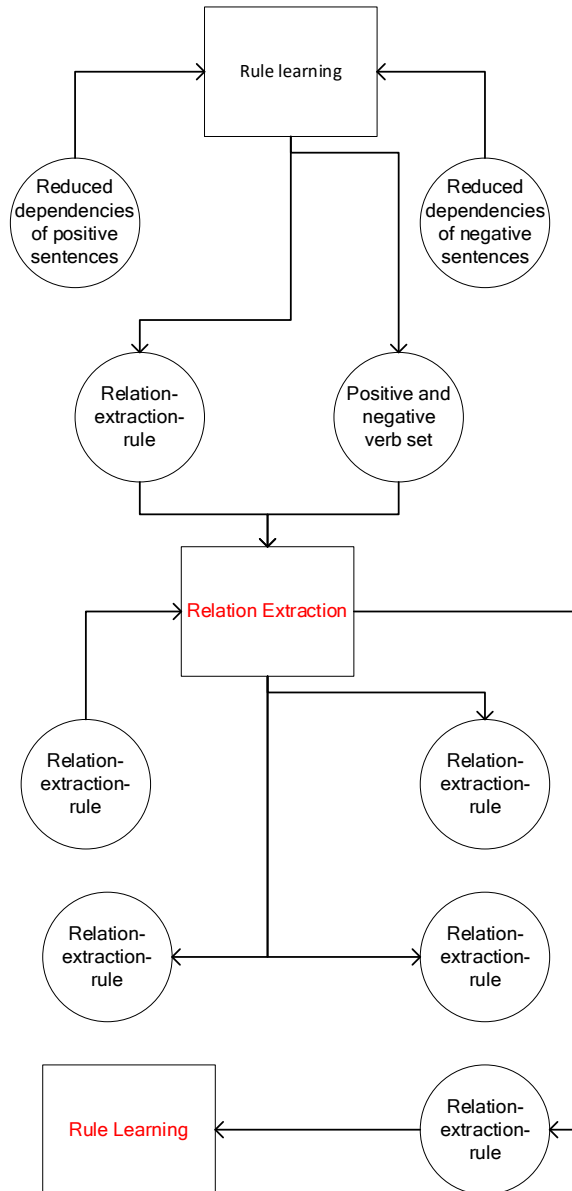
**Fig. 1:** Preprocessing sequence of training documents

In the training phase, a set of verbs are identified as equivalent verbs, including synonyms for each relation. The negative verb set includes antonyms and verbs relevant to any other relation existing between the same entities. The sets of positive and negative verbs for relation are obtained during the rule learning process. In the end, rules are presented in first-order predicate form, as shown in the following example:

In the domain Bird a relation-extraction-rule generated for the relations *feed\_on()* is,

$$\forall x \forall y (nsubj(VB, x) \wedge dobj(VB, y) \wedge \neg prep\_except(NN, y) \wedge \neg negative(VB) \wedge \neg neg(VB, not) \rightarrow feed\_on(x, y))$$

where,  $x \in \{\text{Bird}\}$ ,  $y \in \{\text{Bird\_food}\}$ ,  $VB \in \{\text{Verb}\}$  and  $NN \in \{\text{Noun}\}$ . The negative clauses  $\neg prep\_except((NN, y))$  and  $\neg neg(VB, not)$  are added to the rule from negative sentences.



**Fig. 2:** Input and output of the two processes rule learning and relation extraction



### 3.2. Identifying relative clauses and conjunctions

Extracting information accurately, from the compound sentences and sentences which consist of relative clauses or conjunctions should be addressed further, paying attention to connecting words used in each of those categories. In compound sentences, two independent clauses are connected by compound words. Therefore, sentence constituents connected by compound words can be processed separately, and all the constituents in the sentence can be considered as true statements in the normal way as in simple sentences. For example, from the sentence s(1) shown in section 3.1, two instances of the relation *feed-on()* can be extracted independently. Since relative clauses describe nouns and are directly addressed by the dependencies, further processing is not needed in generating relation-extraction-rules. Sentence s(2) is an example of a sentence with a relative clause:

- *The West Indian whistling duck is a whistling duck that breeds in the Caribbean* s(2)

But with conjunctions, an extra effort is needed to retain the accuracy of the information given by the sentence because the truth value of the sentence depends on some conjunctions. Some conjunctions are strictly conditional and have a direct effect on the truth value of the information given by a sentence depending on the truth value of a constituent coming with conjunction. Sentences can be made more informative with some conjunctions, and the truth value of the sentence does not depend on the truth value of the conjunctive part. Therefore, unconditional conjunctions can be handled in the same way as compound sentences. For example, unconditional conjunction “and” in the sentence s(3) have no effect on the truth value of either part.

- *A standard fistball is hollow, filled with air, and is made of leather.* s(3)

We show commonly used conditional and unconditional conjunctions in Table 1.

**Table 1:** Commonly used conjunctions

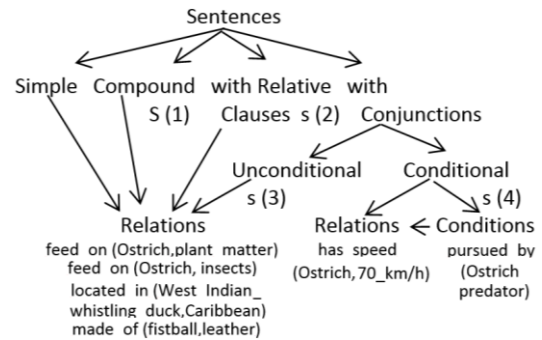
Conditional Conjunctions	Unconditional Conjunctions
If	While
When	Though
As long as	Although
Before	Whether
After	And
Provided	As well as
Until	

Conditions that come with the conjunctions can be considered as a restriction for the relationship to be true. Information wrapped in the sentence fragment with the conditional conjunction can be captured separately and presented as a condition/restriction for information given by the other fragment of the sentence. For example, the sentence s(4) gives the information that ostriches

can have a minimum speed of 70 km/h under the condition “pursued by a predator.”

- *When being pursued by a predator, ostriches have been known to reach speeds in excess of 70 km/h* s(4)

Then the relation instance *has\_speed(Ostrich, 70\_km/h)* will be true if the condition *pursued\_by(Ostrich, predator)* holds. Fig. 3 shows the way the sentences are categorized with respect to relation extraction, accommodating four example sentences.



**Fig. 3:** Main sentence categories with respect to information extraction

In extracting the condition, the dependencies are searched to find the verb constituent in the conjunctive part. Then that verb is taken as a relation predicate, and it is attributed to the subjective noun or an entity where applicable and the closest noun to the conjunctive verb.

There are limited numbers of conjunctions, as shown in Table 1 in their categorized form. Then conjunctions can be identified by the lexical term itself

### 3.3. Modelling relation-extraction-rules on Markov logic network

MLN combines first-order logic with the probabilistic model and requires grounding all the first-order clauses by substituting constants for all the variables in them (Domingos and Richardson, 2006). The probability distribution over possible worlds  $x$  specified by the ground Markov network is given by Eq. 1 and Eq. 2:

$$P(X = x) = \frac{1}{Z} \exp(w_i n_i(x)) \quad (1)$$

where,

$$Z = \sum_{x \in X} \exp(\sum_j w_j n_j(x)) \quad (2)$$

where,  $w_i$  is the weight of the  $i^{\text{th}}$  clause, and  $n_i$  is the number of true occurrences of the  $i^{\text{th}}$  clause.

Verbs and entity instances in the training data corpus are used to ground the relation-extraction-rules. Since the number of grounding is intractable with a large number of substitutions, reducing the number of clauses in the condition of the rules is

vital for efficient implementation before MLN is used on them. Sets of negative and positive verbs are obtained during the implementation of the ILP method for rule generation. Negative verb clauses ( $\neg\text{negative}(VB)$ ) can be omitted from the rules because all the verbs used in MLN are positive verbs. Although the atom  $\neg\text{neg}(VB, \text{not})$  does not contain any entity mentions, it is relevant to a particular pair of entity instances. In the example rule, it is relevant to *Bird* and *Bird\_food* instances. But the atom itself does not contain *Bird* or *Bird\_food* variables because the rules normally applied to the reduced dependencies of a sentence. The negative literal  $\neg\text{prep\_except}((NN, y))$  is not necessarily relevant to one rule and can be added to all the rules to avoid the extraction of false positives. Therefore omitting  $\neg\text{neg}(VB, \text{not})$  and  $\neg\text{prep\_except}((NN, y))$  does not have a significant impact on the rule weight. Then the above-mentioned example rule for the relation *feed\_on*(*Bird*, *Bird\_food*) in section 3.1 is reduced to a rule with only three clauses as follows for the MLN weight learning process.

$$\forall x \forall y (n\text{subj}(VB, x) \wedge d\text{obj}(VB, y) \rightarrow \text{feed\_on}(x, y))$$

However, in the relation extraction process, the complete rule with all the clauses is applied.

Identified entities and verbs from the dependencies are used in grounding the relation-extraction-rules when modeling them in MLN. In addition, the knowledge base consists of evidence that is considered as known as atoms from the training corpus.

MLN requires counting the number of true groundings of formula at a given world state. The probabilistic state-space created by a large database is intractable to do these counting. The higher the number of objects in the MLN, the more difficult the computations become. In this situation, the state space can be reduced by removing the known true literals from the MLN. The negative verbs are not used in grounding the rules, but negative relation instances are used as evidence. Furthermore, the main relation verb can replace all the equivalent verbs generated during the ILP process. In this way number of atoms in the initial MLN can approximately be reduced to a number which fluctuates around half of the initial number of atoms, depending on the number of evidence atoms available.

We use discriminative learning to find weights for relation-extraction-rules. In discriminative learning, the conditional likelihood of query atoms is used (Singla and Domingos, 2005). The conditional likelihood of query atoms  $y$  given evidence atoms  $x$  is shown by Eq. 3:

$$P(y|x) = \frac{1}{Z_x} \exp(\sum_{i \in F_y} w_i n_i(x, y)) \quad (3)$$

where,  $F_y$  is the set of all MLN clauses with at least one grounding involving a query atom, and  $n_i(x, y)$  is

the number of true groundings of the  $i^{\text{th}}$  clause involving query atoms.

The gradient of the Conditional log-likelihood is given by

$$\begin{aligned} \frac{\partial}{\partial w_i} (\log P_w(y|x)) &= n_i(x, y) - E_w[n_i(x, y)] \\ \frac{\partial}{\partial w_i} (\log P_w(y|x)) &= n_i(x, y) - \sum_{y'} P_w(y'|x) n_i(x, y') \end{aligned} \quad (4)$$

Although the number of grounded atoms can be reduced as explained above, computing expected counts  $E_w$  is intractable. Closed World Assumption cannot be used with the dependency literals because the domain is infinite, although a limited number of training data is used in the experiment. Therefore  $E_w$  can be approximated by the counts  $n_i(x, y_w^*)$  in the MAP(Maximum A Posteriori) state. In the problem domain given under experimental results, finding a single MAP state is not guaranteed because the same conditional probability value exists for the number of states. Therefore, Contrastive Divergence (CD) (Lowd and Domingos, 2007) is used in gradient calculations instead of using MAP state. CD approximates the expectations from a small number of Monte Carlo Markov Chain (MCMC) samples. Gibbs sampling is chosen with CD in order to create samples of states. Each Gibbs step consists of sampling a ground atom when its Markov blanket is given. Gibbs sampling requires weights of rules in its sampling process. The weight of a rule is calculated basically for Gibbs sampling by the log odds between a world where the rule is true and a world where the rule is false when other things are equal.

Algorithm for the construction of all the groundings with respect to relation-extraction-rules:

*F*-a set of extraction rules

*E1*-a set of instances of entity1

*E2*-a set of instances of entity2

*VB*-a set of verbs

*NN*-a set of nouns

*PVB*-a set of positive verbs

*NVB*-a set of negative verbs

$G_F$ -set of ground atoms

$G_F = \emptyset$

For each rule in *F*

Convert the rule into its Clausal form CNF (*F*)

If  $NVB \neq \emptyset$

Remove the negative verbs from the *VB*

$(PVB \leftarrow VB \setminus NVB)$

$F \leftarrow \text{CNF}(F)$

For each clause  $F_i \in F$

$G_i = \{F_i\}$ .

For each variable  $x$  in  $F_i$

For each clause  $F_j(x)$

If the type of  $x$  is entity1

Obtain the ground clauses substituting all the values from *E1*

If the type of  $x$  is entity2

Obtain the ground clauses substituting all the values from E2  
 If the type of x is VB  
 Obtain the ground clauses substituting all the values from PVB  
 If the type of x is NN  
 Obtain the ground clauses substituting all the values from NN

$$G_i \leftarrow (G_i \setminus F_i(x)) \cup \{F_i(c_1), F_i(c_2), F_i(c_3), \dots\}$$

where  $c_1, c_2, c_3$ , represent the members of E1, E2, PVB, NN)

$$G_f \leftarrow (G_f \cup G).$$

The probability of a ground atom  $X_i$  with respect to a Markov Blanket  $B_i$  is given by Eq. 5:

$$P(X_i = x_i | B_i = b_i) = \frac{\exp(\sum_{f_i \in F_i} w_{f_i}(X_i = x_i, B_i = b_i))}{\exp(\sum_{f_i \in F_i} w_{f_i}(X_i = 0, B_i = b_i)) + \exp(\sum_{f_i \in F_i} w_{f_i}(X_i = 1, B_i = b_i))} \quad (5)$$

Eq. 4 poses a multivariate weight optimization problem. Gradient Descent, Diagonal Newton, and Conjugate Gradient (Lowd and Domingos, 2007) are available multivariate optimization techniques for efficient weight learning for MLN. Gradient Descent is comparatively slow, and Diagonal Newton has limitations in uncorrelated clauses. Therefore, we prefer the Conjugate Gradient method for weight optimization. In the Conjugate Gradient method, search directions are constructed by conjugation of residuals, and the Polak-Ribiere method (Shewchuk, 1994) is used to find conjugate direction though there are several equivalent expressions for this. The Polak-Ribiere method often converges much more quickly. We use JAVA to generate relation-extraction-rules and MATLAB in the Windows environment in the implementation of relation extraction and weight learning. An overview of the weight learning process is shown in Fig. 4.

In the weight learning process, the strength of a rule is determined by the number of false occurrences that can be covered by the preliminary rule as well as the number of true occurrences for the rule in the training corpus.

A set of rules for the relation *feed\_on()* is given below along with the learned weights,

$$\begin{aligned} \forall x \forall y (nsubj(VB, x) \wedge dobj(VB, y) \wedge \neg prep\_except((NN \vee y) \wedge \neg negative(VB)) \rightarrow feed\_on(x, y) \quad & 1.97 \\ \forall x \forall y (nsubj(VB, x) \wedge con\_and(y, y) \wedge \neg negative(VB) \wedge \neg neg(VB, not) \rightarrow feed\_on(x, y) \quad & 0.67 \\ \forall x \forall y (nsubj(VB, x) \wedge prep\_on(VB, y) \wedge \neg negative(VB) \wedge \neg neg(VB, not) \rightarrow feed\_on(x, y) \quad & 1.21 \\ \forall x \forall y (nsubj(VB, y) \wedge prep\_of(VB, y) \wedge \neg negative(VB) \wedge \neg neg(VB, not) \rightarrow feed\_on(x, y) \quad & 0.7 \end{aligned}$$

### 3.4. Extraction of relations

The Stanford dependencies of sentences of the known entities in the document are searched to find the compatibility of an extraction rule with the dependencies. Entity instances in sentences covered by the rules of a particular relation are extracted as

the attribute values of that relation. When a sentence cannot be covered by extraction rules, the positive and negative verb sets are searched in order to find out whether the main verb constituent is equivalent to any of the verbs in the two sets. Sentences of entities not extracted as a relation by existing relation-extraction-rules can be processed in order to find whether the entities form a negative relation or a new relation.

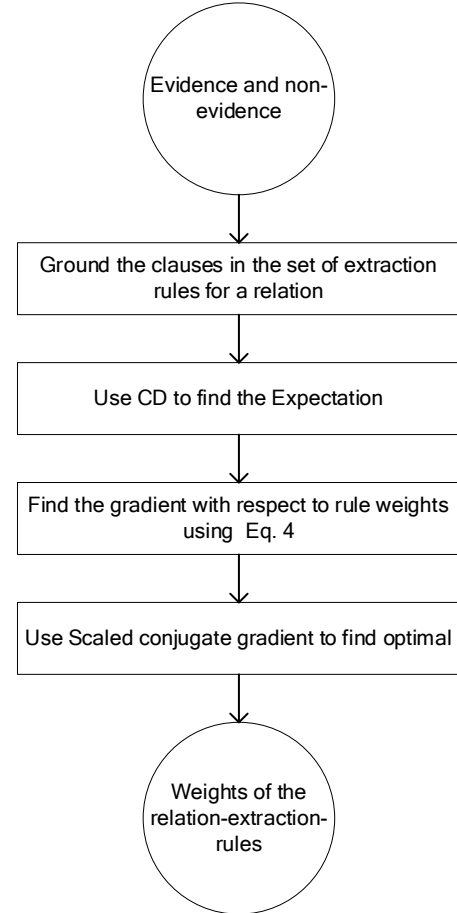


Fig. 4: Overview of the weight learning process

An ambiguous sentence with respect to extraction rules can be categorized into one of the following situations:

- Verb unknown, but extraction rules cover the dependencies, Eg. *The Cape Barren goose is a large goose resident in southern Australia*
- Verb known, but extraction rules cannot cover the dependencies, Eg. *There are subspecies of Barn Swallow which breed across the Northern Hemisphere.*
- Verb unknown and extraction rules cannot cover the dependencies.

Sentences in category (iii) are assumed to be formed a completely new relationship and they are used to formulate the new relation. The relation is labeled by the main verb of the sentence, i.e., the verb constituent contained in the atomic formula "nsubj" when there is only one verb in the dependencies.

We calculate the probability of a rule  $F_1$  in extracting relation instances with the given evidence  $F_2$  on MLN by Eq. 6.

$$P(F_1|F_2, L, C) = \frac{\sum_{X \in X_{F_1} \cap X_{F_2}} P(X=x|ML, C)}{\sum_{X \in X_{F_2}} P(X=x|ML, C)} \quad (6)$$

where,  $L$  is the MLN,  $C$  is the set of constants, and  $X_{F_1}$  is the set of states that  $F_1$  holds and  $X_{F_2}$  is the set of states that  $F_2$  holds.

#### 4. Experimental results

The proposed relation extraction system is used on the domains *Bird* and *Sport* to extract relation instances existing between annotated entities. Since the training data set is continuously updated by the system, a rather small number of Wikipedia pages (100) are used initially. The rule generation process is continued with the updated corpus to learn new relation-extraction-rules, which are added to the existing rule base. The pages are used as they are in the Wikipedia for entity extraction. Once the entities are annotated by the system, only language dependencies of the sentences annotated with entities are used to learn the rules for relation extraction. Then the reduced dependencies, sets of entities present, the relation verb, sets of other verbs and nouns, and adjectives/adverbs in the sentence are the data sources for the rule generation process. The statistics of the relation types in both domains are given in Fig. 5 and Fig. 6. The probability values shown are the values with respect to the best rule in each relation type. Table 2 shows the evaluation measures for the relations considered in the domains *Bird* and *Sport*.

Relation-extraction-rules are applied to test documents from Wikipedia, and relation instances are identified. Nonambiguous sentences with respect to relation-extraction-rules give the instances for the relations with higher probabilities.

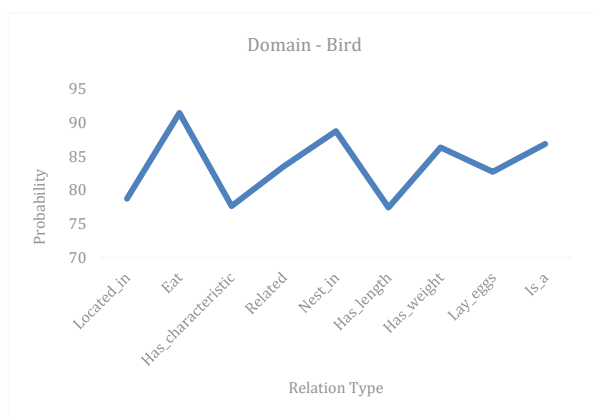


Fig. 5: Probability of relation extraction in the domain bird

The certainty of the extracted relation instances is measured by probability calculations done according to the Eq. 6. The probability calculations here are based on the dependencies of individual sentences, not on the entire knowledge base, which is used for weight calculations. Each relation-extraction-rule is invoked independently, and the

knowledge base has no impact on the relation extraction.

Playing method is identified as an entity, and "Played" is a common term for all the playing method relations such as striking, shooting, passing, etc. Since relation extraction is performed based on the entities identified in the entity extraction phase, both extraction processes are mutually exclusive events. If the entity identification is inaccurate, the relation identification between incorrect entities is bound to be false. However, relation extraction is evaluated independently of the accuracy of entity extraction because techniques have been used and developed for both extraction processes independently. Therefore, when evaluating a relation extraction, a 100% accuracy is assumed for extracted entities. The same number of Wikipedia documents have been used in both entity and relation extraction.

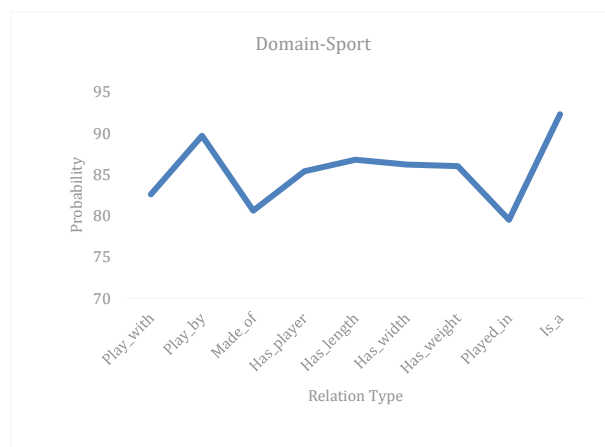


Fig. 6: Probability of relation extraction in the domain sport

When analyzing the results, few points can be readily identified. In the case of measurement, relations *has\_length*, *has\_weight* etc., the main reason for low precision is the presence of the measurements with comparative adjectives such as more than, less than. For an example the sentence "The ball weights approximately 100 grams more than the volleyball one" contains the entity types *Tool* and *Weight*; but does not give the relation *has\_weight()* correctly. Similarly, incorrect identification of equivalent verbs for a relation when the verb is unknown obviously has an impact on the precision. In the domain *Sport* there are sentences annotated with the entity type *sport* more than once in many test documents. These sentences can give the relations *is\_similar\_to(Sport, Sport)* or *is\_version\_of(Sport, Sport)* which have not been considered in the initial relation extraction task.

Since the availability of appropriate results for relation extraction similar to relations considered in our experimental domains is scarce in the literature, we first select two approaches to compare the overall performance of relation-extraction-rules. Both the approaches (Wang et al., 2006; Pawar et al., 2016) use Doddington et al. (2004) data set for their experimental evaluations and have considered a



comparatively higher number of predefined relations. They also represent two time periods along the research line of information extraction. Furthermore, Pawar et al. (2016) also used MLN to model the rules generated in their system. Table 3

shows the overall comparative performance of relation-extraction rules. Evaluation measures used for Relation-extraction-rules in Table 3 are averages of all the individual measures.

**Table 2:** Evaluation Measures for Relations in Two Domains Domain

	Relation	Precision %	Recall %	F-Measure
Bird	located_in(Bird, Location)	83.70	91.21	87.29
	eat(Bird, Diet)	88.22	68.01	76.81
	has_characteristic(Bird, Bird_Part)	80.26	83.67	81.93
	has_characteristic(Bird_Part, Feature)	80.82	76.43	78.56
	related(Bird, Bird)	72.48	89.30	77.15
	nest_in(Bird, Nest)	77.52	65.55	71.03
	has_length(Bird, length)	70.56	91.60	79.72
	has_weight(Bird, weight)	71.80	90.48	80.06
	lay_eggs(Bird, Egg_number)	91.38	90.65	91.01
Sport	is_a(Bird, Super_bird)	92.01	89.29	90.63
	play_with(Sport, Equipment)	76.75	86.74	81.44
	played(Method, Equipment)	62.15	74.06	67.58
	made_of(Equipment, Material)	80.01	82.17	81.08
	has_player(Sport, Player_number)	67.50	72.40	69.86
	has_length(Tool, Length)	72.07	80.25	75.94
	has_width(Tool, Width)	73.98	80.01	76.88
	has_weight(Tool, Weight)	68.25	83.88	75.26
	played_in(Sport, Location)	84.08	92.50	88.09
	is_a(Sport, Super_sport)	92.63	94.06	93.34

**Table 3:** Overall comparative performance of relation-extraction-rules

Approach	Precision %	Recall %	F-Measure
Wang et al. (2006)	73.87	69.5	71.59
Pawar et al. (2016)	57.9	45.6	51.0
Relation-extraction-rules	79.95	92.6	85.8

Secondly, we again select two different approaches (Carlson et al., 2010; Yao et al., 2011) to make a comparison of individual relations from each domain. In the domain Bird, the relation *located\_in()* can be compared with the relation *liveIn* from Yao et al.'s (2011) approach though arguments of *Located\_in* relation are *Bird* and *Location* whereas arguments of the *liveIn* relation are *Person* and *Location*. In the domain Sport, the relation *play\_with()* can be compared with the relation *SportUsesSportsEquipment* from Carlson et al.'s (2010) approach. It uses constraints to a couple of semi-supervised learning. Three coupling algorithms CPL, CSEAL, and MBL have been developed in their approach to information extraction. The results are shown in Table 4.

**Table 4:** Comparison of the Precision of Two Relations

Relation	Yao et al	Carlson et al			Relation-extraction-rules
		CPL	CSEAL	MBL	
located_in()	56				83.7
play_with()		33	100	33	76.75

The CSEAL algorithm achieves 100% precision, but it is claimed in their publication that the MBL gives the overall best performance, and CSEAL incurs some loss in a recall.

Although a smaller number of training examples are used to initiate the system, it will not affect the performance of the system because any situation that cannot be covered by the extraction rules is considered as an instance for a new relation and a new extraction rule is generated for the relation

accordingly. In addition to that, the training set is continuously expanded by the information relevant to extracted relation instances. Therefore, the use of a smaller training data set becomes an advantage here and has no adverse effect on the performance of the entire system. With the expanded training corpus, the performance of the system is expected to be improved further.

## 5. Conclusion

We have presented a method to use language dependency clauses successfully in a rule-based system for statistical relation extraction. Statistical relation traction is enabled by modeling the rules in the Markov logic environment for weight learning. We also have discussed the extraction of relation instances from compound sentences, considering the conditions embedded in the sentence. The applicability of relation-extraction-rules is demonstrated in two different domains with two different sets of relations, although some relations are usable over a wide range of domains with different entities. We initially start the training with a rather small corpus of selected relation instances which cover a range of sentence structures. The training corpus is expanded with extracted instances, selected by a simple statistical method that can be enhanced further. We have also shown some limitations of the system, and it can be extended to address the issues. Current relation extraction relies on entity extraction done by a different method, but entity extraction can also be done by the same method inducing rules for entity identification. This method can be used as a verification of entity extraction, and then errors propagating from incorrect entity extraction can be minimized.

## Compliance with ethical standards

## Conflict of interest

The authors declare that they have no conflict of interest.

## References

- Aitken JS (2002). Learning information extraction rules: An inductive logic programming approach. In the 15<sup>th</sup> European Conference on Artificial Intelligence (ECAI), IOS Press, Lyon France: 355-359.
- Aljamel A, Osman T, and Acampora G (2015). Domain-specific relation extraction: Using distant supervision machine learning. In the 7<sup>th</sup> International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, IEEE, Lisbon, Portugal, 1: 92-103. <https://doi.org/10.5220/0005615100920103>
- Barrière C (2016). Pattern-based relation extraction. In: Barrière C (Ed.), Natural language understanding in a semantic web context: 205-229. Springer, Cham, Switzerland. [https://doi.org/10.1007/978-3-319-41337-2\\_11](https://doi.org/10.1007/978-3-319-41337-2_11)
- Carlson A, Betteridge J, Kisiel B, Settles B, Hruschka ER, and Mitchell TM (2010). Toward an architecture for never-ending language learning. In the 24<sup>th</sup> AAAI Conference on Artificial Intelligence, AAAI Press, Atlanta, USA: 1306-1313.
- Celjuska D and Vargas-Vera M (2004). Ontosophie: A semi-automatic system for ontology population from text. In the International Conference on Natural Language Processing, Hainan Island, China.
- Chiticariu L, Li Y, and Reiss F (2013). Rule-based information extraction is dead! Long live rule-based information extraction systems! In the 2013 Conference on Empirical Methods in Natural Language Processing, Seattle, USA: 827-832.
- Cimiano P and Völker J (2005). Text2Onto. In the International Conference on Application of Natural Language to Information Systems, Springer, Alicante, Spain: 227-238. [https://doi.org/10.1007/11428817\\_21](https://doi.org/10.1007/11428817_21)
- Ciravenga F and Wills Y (2003). Designing adaptive information extraction for the semantic web in Amilcare; annotation for the semantic web. In: Handschuh S and Staab S (Eds.), Frontiers in artificial intelligence and applications: 112-127. IOS Press, Amsterdam, Netherlands.
- Craven M, DiPasquo D, Freitag D, McCallum A, Mitchell T, Nigam K, and Slattery S (2000). Learning to construct knowledge bases from the World Wide Web. Artificial Intelligence, 118(1-2): 69-113. [https://doi.org/10.1016/S0004-3702\(00\)00004-7](https://doi.org/10.1016/S0004-3702(00)00004-7)
- Cunningham H, Maynard D, Bontcheva K, Maynard HD, and Tablan V (2002). GATE: A framework and graphical development environment for robust NLP tools and applications. In the 40<sup>th</sup> Anniversary Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Philadelphia, USA: 168-175.
- Doddington GR, Mitchell A, Przybocki MA, Ramshaw LA, Strassel SM, and Weischedel RM (2004). The automatic content extraction (ace) program-tasks, data, and evaluation. In the Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 2004), 2(1): 837-840.
- Domingos P and Richardson M (2006). Markov logic networks. Machine Learning, 62(1-2): 107-136. <https://doi.org/10.1007/s10994-006-5833-1>
- Drumond L and Girardi R (2009). An experiment using Markov logic networks to extract ontology concepts from text. In the Proceedings of the 2010 ACM Symposium on Applied Computing (ACM, 2010): 1354-1358.
- Dzeroski S and Lavrac N (1994). Inductive logic programming: Techniques and applications. Ellis Horwood, New York, USA.
- Eom JH and Zhang BT (2004). PubMiner: Machine learning-based text mining system for biomedical information mining. In the International Conference on Artificial Intelligence: Methodology, Systems, and Applications, Springer, Varna, Bulgaria: 216-225. [https://doi.org/10.1007/978-3-540-30106-6\\_22](https://doi.org/10.1007/978-3-540-30106-6_22)
- Etzioni O, Fader A, Christensen J, and Soderland S (2011). Open information extraction: The second generation. In the 22<sup>nd</sup> International Joint Conference on Artificial Intelligence, AAAI Press, Barcelona, Spain: 3-10.
- Fader A, Soderland S, and Etzioni O (2011). Identifying relations for open information extraction. In the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Edinburgh, UK: 1535-1545.
- Hong G (2005). Relation extraction using support vector machine. In the International Conference on Natural Language Processing, Springer, Jeju Island, Korea: 366-377. [https://doi.org/10.1007/11562214\\_33](https://doi.org/10.1007/11562214_33)
- Ji G, Liu K, He S, and Zhao J (2017). Distant supervision for relation extraction with sentence-level attention and entity descriptions. In the 31<sup>st</sup> AAAI Conference on Artificial Intelligence, AAAI Press, San Francisco, USA: 3060-3066.
- Li Z, Yang Z, Shen C, Xu J, Zhang Y, and Xu H (2019). Integrating shortest dependency path and sentence sequence into a deep learning framework for relation extraction in clinical text. BMC Medical Informatics and Decision Making, 19: 22. <https://doi.org/10.1186/s12911-019-0736-9> PMID:30700301 PMCID:PMC6354333
- Lima R, Espinasse B, and Freitas F (2019). A logic-based relational learning approach to relation extraction: The OntoILPER system. Engineering Applications of Artificial Intelligence, 78: 142-157. <https://doi.org/10.1016/j.engappai.2018.11.001>
- Lowd D and Domingos P (2007). Efficient weight learning for Markov logic networks. In the European Conference on Principles of Data Mining and Knowledge Discovery, Springer, Warsaw, Poland: 200-211. [https://doi.org/10.1007/978-3-540-74976-9\\_21](https://doi.org/10.1007/978-3-540-74976-9_21)
- Mintz M, Bills S, Snow R, and Jurafsky D (2009). Distant supervision for relation extraction without labeled data. In the Joint Conference of the 47<sup>th</sup> Annual Meeting of the ACL and the 4<sup>th</sup> International Joint Conference on Natural Language Processing of the AFNLP, Association for Computational Linguistics, Suntec, Singapore, 2: 1003-1011. <https://doi.org/10.3115/1690219.1690287>
- Mooney RJ and Nahm UY (2002). Text mining with information extraction. In the AAAI 2002 Spring Symposium on Mining Answers from Texts and Knowledge Bases, AAAI Press, Palo Alto, USA: 60-67.
- Ningthoujam D, Yadav S, Bhattacharyya P, and Ekbal A (2019). Relation extraction between the clinical entities based on the shortest dependency path based LSTM. Available online at: <https://bit.ly/2V40kzi>
- Pawar S, Bhattacharya P, and Palshikar GK (2016). End-to-end relation extraction using Markov logic networks. In the International Conference on Intelligent Text Processing and Computational Linguistics, Springer, Konya, Turkey: 535-551. [https://doi.org/10.1007/978-3-319-75487-1\\_41](https://doi.org/10.1007/978-3-319-75487-1_41)
- Schmitz M, Bart R, Soderland S, and Etzioni O (2012). Open language learning for information extraction. In the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, Association for Computational Linguistics, Jeju Island, Korea: 523-534.
- Seneviratne MDS and Ranasinghe DN (2011). Inductive logic programming in an agent system for ontological relation

- extraction. *International Journal of Machine Learning and Computing*, 1: 344-352.  
<https://doi.org/10.7763/IJMLC.2011.V1.51>
- Seneviratne MDS and Ranasinghe DN (2014). Natural language dependencies for ontological relation extraction. In the 14<sup>th</sup> International Conference on Advances in ICT for Emerging Regions, IEEE, Colombo, Sri Lanka: 142-148.  
<https://doi.org/10.1109/ICTER.2014.7083893>
- Shewchuk JR (1994). An introduction to the conjugate gradient method without the agonizing pain. School of Computer Science, Carnegie Mellon University, Pittsburgh, USA.
- Singla P and Domingos P (2005). Discriminative training of Markov logic networks. *American Association for Artificial Intelligence*, 5: 868-873.
- Wang T, Li Y, Bontcheva K, Cunningham H, and Wang J (2006). Automatic extraction of hierarchical relations from text. In the European Semantic Web Conference, Springer, Budva, Montenegro: 215-229.  
[https://doi.org/10.1007/11762256\\_18](https://doi.org/10.1007/11762256_18)
- Wu F and Weld DS (2010). Open information extraction using Wikipedia. In the 48<sup>th</sup> Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Uppsala, Sweden: 118-127.
- Yao L, Haghighi A, Riedel S, and McCallum A (2011). Structured relation discovery using generative models. In the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Edinburgh, UK: 1456-1466.
- Yildiz B and Miksch S (2011). Motivating ontology-driven information extraction. In: Munshi UM and Chaudhuri BB (Eds.), *Multimedia information extraction and digital heritage preservation*: 1-19. Volume 10, World Scientific, Singapore, Singapore.  
[https://doi.org/10.1142/9789814307260\\_0001](https://doi.org/10.1142/9789814307260_0001)
- Zeng X, He S, Liu K, and Zhao J (2018). Large scaled relation extraction with reinforcement learning. In the 32<sup>nd</sup> AAAI Conference on Artificial Intelligence, AAAI Press, New Orleans, USA: 5658-5665.
- Zhang NR (2001). Hidden Markov models for information extraction. Technical Report, Stanford Natural Language Processing Group, Stanford, USA.