An Integrated Probabilistic and Logic Approach to Encyclopedia Relation Extraction with Multiple Features *

Xiaofeng YU **Wai LAM**

Information Systems Laboratory
Department of Systems Engineering & Engineering Management
The Chinese University of Hong Kong
Shatin, N.T., Hong Kong
{xfyu,wlam}@se.cuhk.edu.hk

Abstract

We propose a new integrated approach based on Markov logic networks (MLNs), an effective combination of probabilistic graphical models and first-order logic for statistical relational learning, to extracting relations between entities in encyclopedic articles from Wikipedia. The MLNs model entity relations in a unified undirected graph collectively using multiple features, including contextual, morphological, syntactic, semantic as well as Wikipedia characteristic features which can capture the essential characteristics of relation extraction task. This model makes simultaneous statistical judgments about the relations for a set of related entities. More importantly, implicit relations can also be identified easily. Our experimental results showed that, this integrated probabilistic and logic model significantly outperforms the current state-of-the-art probabilistic model, Conditional Random Fields (CRFs), for relation extraction from encyclopedic articles.

1 Introduction

Relation extraction is a growing area of research that discovers various predefined semantic relations (e.g., visited, associate, and executive) between entity pairs in text. As a subtask in Information Extraction (IE), this problem has generated much interest and has been formulated as part of Message Understanding Conferences (MUC) and Automatic Content Extraction (ACE) Evaluation.

Reliably extracting relations between entities in natural-language documents is still a difficult, unsolved problem. A large number of engineered systems were developed for identifying relations of interest. Recent approaches to this problem include statistical parsing (Miller *et al.*, 2000), logistic regression (Kambhatla, 2004), feature-based methods (Zhou *et al.*, 2005; Toru *et al.*, 2007), and kernel methods (Zelenko *et al.*, 2003; Culotta and Sorensen, 2004; Bunescu and Mooney, 2005, 2006).

In text, this usually amounts to examining pairs of entities in a document and determining whether a relation exists between them. In general, the above approaches to relation extraction suffer from the following three difficulties: (1) enumerating all pairs of entities, even when restricted to pairs within a sentence, results in a low density of positive relation examples, (2) these approaches assume that relations only exist within document, and classify them independently without considering dependencies between entities. However, this assumption does not hold in practice, and ignoring dependencies between entities may lead to reduced performance, and (3) implicit relations can hardly be discovered in these models since they generally exist in cross document and they are only implied by the text. And these are the sorts of relations on which current extraction models perform most poorly.

In this paper we propose a new integrated approach based on Markov logic networks (MLNs) to extracting relations between entities in English encyclopedic articles from Wikipedia. We predict only relations between the *principal entity* and each mentioned *secondary entity* in Wikipedia articles. By anchoring one argument of relations to be the principal entity, we alleviate the difficulty of enumerating all pairs of entities in a document. This approach can incorporate rich dependencies between entities by modeling entity relations in a coherent undirected graph in a *collective* manner, and make simultaneous statistical judgments about the relations for a set of related entities. It can also exploit relational *autocorrelation*, a widely observed characteristic of relational data

The work described in this paper is substantially supported by grants from the Research Grant Council of the Hong Kong Special Administrative Region, China (Project Nos: CUHK4193/04E and CUHK4128/07) and the Direct Grant of the Faculty of Engineering, CUHK (Project Codes: 2050363 and 2050391). This work is also affiliated with the Microsoft-CUHK Joint Laboratory for Human-centric Computing and Interface Technologies.

Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), pages 1065–1072
Manchester, August 2008

in which the value of a variable for one instance is highly correlated with the value of the same variable on another instance. We show how a variety of well-engineered features can be easily and concisely formulated as first-order logic and incorporated into MLNs, and we show how implicit relations can be easily discovered in this modeling. We apply Gibbs sampling, a widely used Markov chain Monte Carlo (MCMC) algorithm, to perform collective inference in MLNs. Experimental results showed that this model yields substantially better results on encyclopedia relation extraction over the current state-of-the-art probabilistic relation extraction model, such as Conditional Random Fields (CRFs).

2 Wikipedia

Wikipedia[1] is the world's largest free online encyclopedia, representing the outcome of a continuous collaborative effort of a large number of volunteer contributors. Virtually any Internet user can create or edit a Wikipedia webpage, and this "freedom of contribution" has a positive impact on both the quantity (fast-growing number of articles) and the quality (potential mistakes are quickly corrected within the collaborative environment) of this online resource. Currently Wikipedia has approximately 9.25 million articles in more than 200 languages.

We investigate the task of discovering semantic relations between entity pairs from Wikipedia's English encyclopedic articles. The basic entry in Wikipedia is an *article*, which mainly defines and describes an entity (also known as *principal entity*) or an event, and consists of a hypertext document with hyperlinks to other pages within or outside Wikipedia. This document mentions some other entities as *secondary entities* related to the *principal entity* (Culotta *et al.*, 2006). All the entities are hyper-linked within the text, and the topic of an article usually defines the principal entity. Moreover, Wikipedia has the category hierarchy structure which is used to classify articles according to their content. All these characteristics make Wikipedia an appropriate resource for the task of relation extraction. In this paper, we predict only relations between the principal entity and each mentioned secondary entity.

An illustrative example of Wikipedia article is shown in Figure 2, where the principal entity *Albert Einstein* is boxed and in italic font, and secondary entities are in italic font. Our goal is to predict what relation, if any, each secondary entity has to the principal entity. For example, there is a *job_title* relation between *theoretical physicist* and *Albert Einstein* and a *father* relation between *Hermann Einstein* and *Albert Einstein*, but no relation between *salesman* and *Albert Einstein*.

Albert Einstein
Albert Einstein (*March 14, 1879 - April 18, 1955*) was a *theoretical physicist*. He was born in *Germany*. His father was *Hermann Einstein*, a *salesman* and *engineer*, and his mother was *Pauline Einstein*. In 1880, the family moved to *Munich*. Albert attended a *Catholic elementary school* and finally he was enrolled in the mathematics program at *ETH Zurich*. Einstein received the *Nobel Prize in Physics* for his services to Theoretical Physics in 1921.

Figure 1: An example of Wikipedia article for relation extraction. The principal entity is boxed and in italic font, and secondary entities are in italic font.

3 Relation Extraction as Sequence Labeling: A Baseline Approach

Note that our goal is to extract relations between the principal entity and each mentioned secondary entity in Wikipedia's English encyclopedic articles. This formulation allows us to view relation extraction as a sequence labeling task such as part-of-speech tagging. Motivated by this observation, we therefore apply Conditional Random Fields (CRFs) (Lafferty *et al.*, 2001), a probabilistic graphical model that has been successfully employed on sequence labeling tasks with state-of-the-art performance. By using the CRF model, each secondary entity's label is its relation to the principal entity, and we can capture the dependency between adjacent labels. For example, in the dataset it is common to see phrases such as "*Albert Einstein* (*1879 - 1955*) was born in *Germany*" for which the labels *birth_year*, *death_year*, and *birth_place* occur consecutively. Sequence models are specifically designed to handle these kinds of dependencies. The modeling flexibility of CRFs permits the feature functions to be complex, arbitrary, non-independent, and overlapping features of the input without requiring additional assumptions, allowing the multiple features described in Section 5 to be directly exploited. To avoid overfitting, we penalized the log-likelihood by the commonly used zero-mean Gaussian prior over the parameters. This gives us a competitive baseline CRF model for relation extraction.

[1] http://www.wikipedia.org/

4 Markov Logic Networks for Collective Relation Extraction

Markov logic networks (MLNs) conduct *statistical relational learning* (SRL) by incorporating the expressiveness of first-order logic into the flexibility of probabilistic graphical models under a single coherent framework (Richardson and Domingos, 2006). An MLN consists of a set of weighted formulae and provides a way of softening first-order logic by making situations, in which not all formulae are satisfied, less likely but not impossible. More formally, the probability distribution of a particular truth assignment x to X specified by the ground Markov network $M_{L,C}$[2] is given by

$$P(X = x) = \frac{1}{Z}\exp\left(\sum w_i n_i(x)\right) = \frac{1}{Z}\prod \phi_i\left(x_{\{i\}}\right)^{n_i(x)} \quad (1)$$

where X is the set of all propositions describing a world x (i.e. all gliterals formed by grounding the predicates with the constants in the domain), $\mathcal{F}$ is the set of all clauses in the MLN, w_i is the weight associated with clause $F_i \in \mathcal{F}$, $n_i(x)$ is the number of true groundings of F_i in x, $x_{\{i\}}$ is the true value of the atoms appearing in F_i, Z is the normalizing partition function, ϕ_i is a real-valued potential function and $\phi_i\left(x_{\{i\}}\right) = e^{w_i}$.

MLNs model the relation extraction task in a *collective* manner and take into account the relation types of related entities. Note that this is different from other relation extraction methods that predict relations independently without considering the relationship between entities. Attributes can be represented in MLNs as predicates of the form $A(x, v)$, where A is an attribute, x is an entity, and v is the value of A in x. The relation is a designated attribute C, representable by $C(x, v)$, where v is x's relation. The relations of different entities depend on each other. Classification is now simply the problem of inferring the truth value of $C(x, v)$ for all x and v of interest given all known $A(x, v)$. In this collective modeling, the Markov blanket of $C(x_i, v)$ includes other $C(x_j, v)$, even after conditioning on the known $A(x, v)$. Relations between entities are represented by predicates of the form $R(x_i, x_j)$.

[2]The graphical structure of $M_{L,C}$ is that: there is an edge between two nodes of $M_{L,C}$ iff the corresponding ground atoms appear together in at least one grounding of one first-order formula.

4.1 Weight Learning

Given a relational database and a set of first-order logic, the weight of each clause can in principle be learned very efficiently by maximizing the pseudo-log-likelihood of this database on the closed world assumption using the limited-memory BFGS algorithm (Liu and Nocedal, 1989). These weights reflect how often the clauses are actually observed in the training data.

To estimate the weights, we maximize the logarithm of the conditional likelihood of the training data

$$\sum_{(x_h, x_o)\in T} \log\Big(p(X_h = x_h | X_o = x_o)\Big) \quad (2)$$

where X_h is a list of possible variables and x_h are the corresponding values in the observation. X_h contains all variables referring to possible ground atoms of entity relations. X_o is the set of variables corresponding to all possible instantiations of the predicates. T is the set of training observations (x_h, x_o). For relation extraction, Equation 2 can be rewritten as

$$p(X_h = x_h | X_o = x_o) = \prod_{Entity\_pairs(p,q)} p\Big(X_{e(p,q)} = x_{e(p,q)} | X_{g(p,q)} = x_{g(p,q)}\Big) \quad (3)$$

where $X_{e(p,q)}$ corresponds to the ground atoms, and $X_{g(p,q)}$ is a list of all variables corresponding to predicates.

With Equation 3, the conditional likelihood in Equation 2 simplifies to

$$\sum_{(x_h, x_o)\in T} \sum_{Entity\_pairs(p,q)} \log\Big(p\big(x_{e(p,q)} | x_{g(p,q)}\big)\Big). \quad (4)$$

where $p(x|y)$ is the abbreviation for $p(X = x|Y = y)$. To calculate the conditional likelihood, we have

$$p\big(x_{e(p,q)} | x_{g(p,q)}\big) = \frac{p\big(x_{e(p,q)}, x_{g(p,q)}\big)}{p\big(1, x_g(p,q)\big) + p\big(0, x_g(p,q)\big)} \quad (5)$$

During MLN weight learning, each first-order formula is converted to Conjunctive Normal Form (CNF). The probabilities of all formulae collectively determine all weights, if we view them as empirical probabilities and learn the maximum likelihood weights. Conversely, the weights in a learned MLN can be viewed as collectively encoding the empirical formula probabilities.

4.2 Inference

In order to perform inference over a given MLN, one needs to ground it into its corresponding Markov network (Pearl, 1988). A large number of efficient inference techniques are applicable and the most widely used approximate solution to probabilistic inference in MLNs is Markov chain Monte Carlo (MCMC) (Gilks *et al.*, 1996). One such algorithm to perform collective inference is called Gibbs sampling. Gibbs sampling starts by assigning a truth value to each query gliteral (a ground literal, i.e. one that contains only ground terms). It then proceeds in rounds to re-sample a value for gliteral X, given the truth values of its Markov blanket MB_X (i.e. the nodes with which it participates in ground clauses).

5 Feature Set

We describe the features used in our model. These features have been shown to be very effective for relation extraction.

Contextual features: Bag-of-words consisting of 4 words to the left and right of the target entity.

Part-of-Speech: Part-of-speech tags are obtained using the Stanford POS Tagger[3], which used rich knowledge sources and features in a log-linear model. POS tags with a window size of 4 around the target entity are used.

Morphological features: Such as whether the entity is capitalized or contains digits or punctuation, whether the entity ends in some suffixes such as *-eer* and *-ician*, etc.

Syntactic features: Syntactic information can lead to significant improvements in extraction accuracy (e.g., Culotta and Sorensen (2004), Bunescu and Mooney (2005)). The POS-tagged corpus is submitted to the Stanford Lexicalized Dependency Parser[4] which generates a dependency parse tree for each sentence and assigns word positions to each word. This parser can also output grammatical relations (typed dependency). The grammatical relations are of the form $relation(rel_i, w_i, w_j)$, where rel_i is one of the fixed set of relations assigned by the parser, and w_i and w_j are two words. The dependency paths, which contain the relevant terms describing the relations between the entity pairs, can be easily extracted. We design a set of first-order formulae that captures some of the most important syntactic phenomena for relation extraction.

Table 1: Representative relation types and corresponding keywords.

Relation	Keywords
job_title	secretary, writer, novelist, captain, cartoonist, actor, actress, physicist, mathematician, singer, naturalist, architect, musician, physician, professor, journalist, banker, businessman, producer, philosopher, worker
visited	from, to, in, at, near, along, visited
associate	work for, along with, together with, perform with, work with, colleague, struck with
member_of	member of, serve in, serve at, serve with, select to, campaign for, election to, involve in, captain with, play for, fellow of, enter
opus	sitcom, picture, film, teleplay, novel, essay, comedy, autobiography, show, movie, plot, drama, painting, book, cartoon, song, music
education	university, academy, school, college, institute
executive	lead, head, leader, president, chairman, committee, executive, officer, mayor, prince, chair, governor
birth_place	born in, born at, birth
death_place	bury in, died in, died at, pass away, inter
nationality	American, English, Irish, French, Italian, Australian, Canadian, Jewish, Russian
award	award, medal, fellowship, prize, pennant, scholarship
participant	during, through

Entity features: Important entities are hyperlinked within the text, but they are not classified by type. Entity type is very helpful for relation extraction. For instance, the relation between a *person* and a *location* should be *visited*, *birth_place*, *death_place*, etc., but cannot be *executive*, *founder*, etc. We identify named entities (person, location and organization) by applying the Stanford Named Entity Recognizer[5], a CRF based sequence labeling model coupled with well-engineered features including additional distributional similarity features. The model is trained on data from CoNLL, MUC-6, MUC-7, and ACE, making it fairly robust in practice. Types of other entities (e.g., date, year, and month) can be well classified by rule-based approach due to their relatively fixed forms.

Keyword features: Some keywords provide crucial clues for relationships between entity pairs. Consider the following sentence:

Bill Gates is the founder of the *Microsoft* Corporation.

If *Bill Gates* is the principal entity and *Microsoft* is the secondary entity, the keyword *founder* implies that there is a *founder* relation between them. Similarly, the *executive* relation may be implied by

[3] http://nlp.stanford.edu/software/tagger.shtml

[4] http://nlp.stanford.edu/software/lex-parser.shtml

[5] http://nlp.stanford.edu/software/CRF-NER.shtml

keywords *lead*, *head*, *leader*, *president*, *chairman*, *executive officer*, *director*, and *administrator*. Moreover, it is particularly interesting that some entities indicate their relation types to corresponding principal entities. Entities containing keywords such as *secretary*, *writer*, *novelist* or *actor* show a *job_title* relation to their principal entities. We exploit tf-idf approach to co-occurrence (collocation) analysis for keyword extraction. Tf-idf is used to measure the relevance of words with a window size of 8 to each relation between entity pairs. And then we rank the relevance scores with respect to each relation and choose keywords with scores higher than the user-defined threshold.

Semantic features: Due to data sparseness, tf-idf model might be unsatisfactory to extract sufficient keywords. We employ WordNet (Fellbaum, 1998), an online lexical database, to extend and enrich each keyword candidate to its synonyms (synsets). For example, the keyword *university* for relation *education* is extended to the set {*university*, *academy*, *college*, *institute*}. Table 1 shows some representative relation types and keywords using tf-idf method and semantic extension.

Wikipedia characteristic features: Relations only exist between principal entities and secondary entities. There is no relation between any two principal entities p, q or two secondary entities x, y.

6 First-Order Logic Representation

All the features described in Section 5 can be easily and concisely represented by first-order formulae, which are used during the MLN learning. First-order formulae are recursively constructed from atomic formulae using logical connectives and quantifiers. Atomic formulae are constructed using constants, variables, functions, and predicates. We give a couple of examples here.

For **contextual features**, it is common to see two secondary entities x and y occur consecutively, accompanied by conjunctions such as "and" or punctuation such as ",", then probably the two entities may have the same relation to the principal entity p. This can be written in first-order logic form as `occur_conse(x,y)` $\Rightarrow$ `same_relation(x,y)`. For **morphological features**, suffixes such as *-eer* and *-ician* may probably show a *job_title* relation to the principal entity p. We therefore can easily write down the logic `person(p)` $\wedge$ `job_suffix(x)` $\Rightarrow$ `job_title(x,p)` to capture this information.

Entity features can be represented using some first-order formulae such as:
`person(p)`$\wedge$`location(x)` $\Rightarrow$ `visited(x,p)` $\vee$ `birth_place(x,p)` $\vee$ `death_place(x,p)`
`person(p)`$\wedge$`location(x)` $\Rightarrow$ `!executive(x,p)`$\wedge$`!founder(x,p)`. The formula `founder_key(x,p)` $\Rightarrow$ `founder_relation(x,p)` can be used for **keyword features**. And **Wikipedia characteristic features** can be well and easily expressed by the logic `principal(p)` $\wedge$ `principal(q)` $\Rightarrow$ `no_relation(p,q)` and `secondary(x)` $\wedge$ `secondary(y)` $\Rightarrow$ `no_relation(x,y)`.

It is worth noticing that some features can be combined in first-order logic formulation. For example, `person(p)` $\wedge$ `organization(x)` $\wedge$ `founder_key(x,p)` $\Rightarrow$ `founder_relation(x,p)` means if there is a *founder* keyword between a person and an organization, probably there is a *founder* relation between them.

7 Implicit Relation Extraction

Implicit relations are those that do not have direct contextual evidence. Implicit relations generally exist in different paragraphs, or even across documents. They require additional knowledge to be detected. Notably, these are the sorts of relations that are likely to have significant impact on performance. A system that can accurately discover knowledge that is implied by the text will effectively provide access to the implications of a corpus. Unfortunately, extracting implicit relations is challenging even for current state-of-the-art relation extraction models.

We show that MLNs can enable this technology. By employing the first-order logic formalism, the implicit relations can be easily discovered from text. Since these formulae will not always hold, we would like to handle them probabilistically by estimating the confidence of each formula. One

Table 2: Examples of first-order logic for implicit relation extraction.

`wife(x,y)` $\Rightarrow$ `husband(y,x)`
`father(x,y)` $\Rightarrow$ `son(y,x)` $\vee$ `daughter(y,x)`
`brother(x,y)` $\Rightarrow$ `brother(y,x)` $\vee$ `sister(y,x)`
`husband(x,y)` $\wedge$ `daughter(z,x)` $\Rightarrow$ `mother(y,z)`
`father(x,y)` $\wedge$ `father(y,z)` $\Rightarrow$ `grandfather(x,z)`
`founder(x,y)` $\wedge$ `superior(x,z)` $\Rightarrow$ `employer(z,y)`
`associate(x,y)` $\wedge$ `member_of(x,z)` $\Rightarrow$ `member_of(y,z)`
`executive(x,y)` $\wedge$ `member_of(z,y)` $\Rightarrow$ `superior(x,z)`

of the benefits of the MLN probabilistic extraction model is that confidence estimates can be straightforwardly obtained.

Consider the following 2 sentences in Wikipedia articles:

1. On November 4, 1842 *Abraham Lincoln* married *Mary Todd*.

2. *Abraham Lincoln* had a son named *Robert Todd Lincoln* and he was born in Springfield, Illinois on 1 August 1843.

State-of-the-art extraction models may be able to detect the *wife* relation between *Mary Todd* and *Abraham Lincoln*, and the *son* relation between *Robert Todd Lincoln* and *Abraham Lincoln* successfully from local contextual clues. However, in the descriptive article of *Robert Todd Lincoln* in Wikipedia, *Robert Todd Lincoln* becomes the principal entity, and the *mother* relation between *Mary Todd* and *Robert Todd Lincoln* is only implied by the text and it is an implicit relation. First-order formalism allows the representation of deep and relational knowledge. Using the logic `wife(x,y)` $\wedge$ `son(z,y)` $\Rightarrow$ `mother(x,z)`, the relational knowledge in the above example can be easily captured to infer the implicit relation. These formulae are generally simple, and capture important knowledge for implicit relation extraction. Examples of first-order logic to infer implicit relations are listed in Table 2.

8 Experiments

8.1 Data

We use the same dataset as in (Culotta *et al.*, 2006) to conduct our experiments. This dataset consists of 1127 paragraphs from 441 pages from the online encyclopedia Wikipedia with 4701 relation instances and 53 relation types labeled. Table 3 shows the relation types and corresponding frequencies of this dataset.

This dataset was split into training and testing sets (70%-30% split), attempting to separate the entities into connected components. There are still occasional paths connecting entities in the training set to those in the testing set, and we believe this methodology reflects a typical real-world scenario.

8.2 Results and Discussion

We design 38 first-order logic formulae (15 formulae are used for implicit relation extraction) to construct the structure of MLNs. Using the features described in Section 5, we train MLNs using a Gaussian prior with zero mean and unit variance on each weight to penalize the pseudo-likelihood, and with the weights initialized at the mode of the prior (zero). The features specify a ground Markov network (e.g., ground atoms) containing one feature for each possible grounding of a first-order formula. Inference is performed for answering the query predicates, given the evidence predicates and other relations that can be deterministically derived. We apply Gibbs sampling to predict relations of entity pairs simultaneously.

Table 3: Statistics of relation types and corresponding frequencies.

Relation	Frequency	Relation	Frequency
job_title	379	daughter	35
visited	368	husband	33
birth_place	340	religion	32
associate	326	influence	31
birth_year	287	underling	27
member_of	283	sister	20
birth_day	283	grandfather	20
opus	267	ancestor	19
death_year	210	grandson	18
death_day	199	inventor	15
education	185	cousin	13
nationality	148	descendant	11
executive	127	role	10
employer	111	nephew	9
death_place	93	uncle	6
award	86	supported_person	6
father	84	granddaughter	6
participant	81	owns	4
brother	71	great_grandson	4
son	68	aunt	4
associate_competition	58	supported_idea	3
wife	57	great_grandfather	3
superior	54	gpe_competition	3
mother	50	brother_in_law	2
political_affiliation	44	grandmother	1
friend	43	discovered	1
founder	43	Overall	4701

Table 4 presents the performance of our relation extraction system based on MLNs compared to CRFs for different types of relations. We use the same set of features for both MLNs and CRFs. For MLNs, all the features are represented using first-order logic. It shows that the MLN system performing collective relation prediction and integrating implicit relation extraction yields substantially better results, leading to an improvement of up to 1.84% on the overall F-measure over the current state-of-the-art CRF model. The improvement is statistically significant ($p < 0.05$ with a 95% confidence interval) according to McNemar's paired tests.

As shown in Table 4, the performance varies greatly from different relation types. Both of the two systems perform quite well on 4 relations: *death_day*, *death_year*, *birth_day*, and *birth_year*.

Table 4: Comparative relation extraction performance. Both CRFs and MLNs are tested on the same set of features in Section 5.

	CRFs			MLNs		
Relation	Precision	Recall	$F_{\beta=1}$	Precision	Recall	$F_{\beta=1}$
death_day	100.00%	94.74%	97.30	98.85%	96.00%	**97.40**
death_year	98.21%	94.83%	96.49	98.14%	95.18%	**96.64**
birth_year	95.12%	95.12%	95.12	94.59%	95.68%	**95.13**
birth_day	93.90%	95.06%	94.48	93.20%	95.80%	94.48
nationality	88.37%	95.00%	**91.57**	88.10%	95.02%	91.43
birth_place	86.81%	92.94%	89.77	87.78%	93.32%	**90.47**
job_title	87.07%	91.82%	89.38	87.63%	91.55%	**89.55**
death_place	89.47%	80.95%	85.00	91.66%	82.99%	**87.11**
education	72.41%	89.36%	80.00	75.11%	90.22%	**81.97**
father	70.97%	88.00%	78.57	71.88%	89.82%	**79.85**
wife	72.22%	81.25%	76.47	72.30%	81.75%	**76.74**
award	94.12%	61.54%	**74.42**	80.88%	66.49%	72.98
mother	81.82%	64.29%	72.00	80.89%	69.33%	**74.67**
political_affiliation	100.00%	53.33%	**69.57**	85.66%	57.12%	68.54
husband	66.67%	60.00%	63.16	67.39%	62.48%	**64.84**
visited	66.29%	55.14%	60.20	66.70%	55.83%	**60.78**
daughter	66.67%	54.55%	60.00	63.67%	59.00%	**61.25**
founder	81.82%	47.37%	60.00	77.39%	52.63%	**62.65**
member_of	59.32%	49.30%	53.85	60.91%	51.66%	**55.90**
executive	64.00%	44.44%	52.46	60.20%	48.48%	**53.71**
superior	66.67%	42.11%	**51.61**	60.55%	44.23%	51.12
brother	50.00%	46.67%	48.28	48.80%	48.57%	**48.68**
opus	68.00%	33.33%	44.74	50.55%	44.75%	**47.47**
son	50.00%	39.13%	43.90	49.30%	41.55%	**45.09**
associate	42.28%	45.22%	43.70	40.77%	47.89%	**44.04**
participant	41.67%	23.81%	**30.30**	31.98%	26.05%	28.71
employer	46.67%	21.21%	29.17	47.78%	27.33%	**34.77**
associate_competition	23.08%	20.00%	21.43	24.38%	20.42%	**22.22**
religion	100.00%	8.33%	**15.38**	15.55%	10.23%	12.34
friend	0	0	0	50.38%	42.33%	**46.01**
sister	0	0	0	34.66%	20.55%	**25.80**
grandfather	0	0	0	23.74%	16.56%	**19.51**
grandson	0	0	0	20.01%	13.39%	**16.04**
cousin	0	0	0	22.00%	7.13%	**10.77**
other types	0	0	0	0	0	0
Overall	73.57%	64.20%	68.57	74.70%	66.58%	**70.41**

Since these relations can be easily identified using the distinct contextual evidence. However, some relations (e.g., *role*, *owns*, etc.) can hardly be extracted. One possible reason is the lack of training data (these relations occur rarely in the dataset). Among all the 53 relation types in the dataset, MLNs successfully extract 34 relations, while CRFs can only detect 29. For all the 34 relations listed in Table 4, MLNs outperform CRFs on 27 types of them. It is particularly interesting that MLNs can successfully predict relations *friend*, *sister*, *grandfather*, *grandson*, and *cousin*, whereas CRFs cannot. CRFs perform relation extraction sequentially without considering connections between entities. This may lead to the label inconsistency problem. For example, CRF sometimes fails to label the *father* relation between *George H. W. Bush* and *George W. Bush*. Implicit relations can hardly be investigated in this sequence labeling model. These disadvantages limit the ability of CRFs for relation extraction to a large extent.

9 Related Work

Only a few research work has attempted relation extraction from Wikipedia. Culotta *et al.* (2006) proposed a probabilistic model based on CRFs to integrate extraction and data mining tasks performed on biographical Wikipedia articles. Relation extraction was treated as a sequence labeling problem and relational patterns were discovered to boost the performance. However, this model extracts relations without considering dependencies between entities, and the best reported F-measure is 67.91, which is significantly (by 2.5%) lower than our MLN system when evaluated on the same training and testing sets. Nguyen *et al.* (2007b,a) proposed a subtree mining approach to extracting relations from Wikipedia by incorporating information

from the Wikipedia structure and by the analysis of Wikipedia text. In this approach, a syntactic tree that reflects the relation between a given entity pair was built, and a tree-mining algorithm was used to identify the basic elements of syntactic structure of sentences for relations. This approach mainly relies on syntactic structures to extract relations. Syntactic structures are important for relation extraction, but insufficient to extract relations accurately. The obtained F-measure was only 37.76, which shows that there is a large room for improving. To the best of our knowledge, our approach is the first attempt at using MLNs for relation extraction from Wikipedia which achieves state-of-the-art performance.

We mention some other related work. Bunescu and Mooney (2007) presented an approach to extract relations from the Web using minimal supervision. Rosenfeld and Feldman (2007) presented a method for improving semi-supervised relation extraction from the Web using corpus statistics on entities. Our work is different from these research work. We investigate supervised relation extraction from Wikipedia based on probabilistic and logic integrated graphical models.

10 Conclusion

We summarize the contribution of this paper. First, we propose a new integrated model based on MLNs, which provide a natural and systematic way by modeling entity relations in a coherent undirected graph collectively and integrating implicit relation extraction easily, to extract relations in encyclopedic articles from Wikipedia. Second, we design multiple features which can be concisely formulated by first-order logic and exploit the collective inference algorithm (Gibbs sampling) to predict relations between entity pairs simultaneously. Third, our system achieved significantly better results compared to the current state-of-the-art probabilistic model for relation extraction from encyclopedic articles.

Having established this relation extraction model, our next step will be to evaluate it on larger datasets, where we expect collective relation extraction and implicit relation discovery to be even more interesting.

References

Razvan C. Bunescu and Raymond J. Mooney. A shortest path dependency kernel for relation extraction. In *Proceedings of HLT-EMNLP 2005*, pages 724–731, Vancouver, British Columbia, Canada, 2005.

Razvan C. Bunescu and Raymond J. Mooney. Subsequence kernels for relation extraction. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems 18*, pages 171–178. MIT Press, Cambridge, MA, 2006.

Razvan C. Bunescu and Raymond J. Mooney. Learning to extract relations from the Web using minimal supervision. In *Proceedings of ACL-07*, pages 576–583, Prague, Czech Republic, June 2007.

Aron Culotta and Jeffrey Sorensen. Dependency tree kernels for relation extraction. In *Proceedings of ACL-04*, Barcelona, Spain, 2004.

Aron Culotta, Andrew McCallum, and Jonathan Betz. Integrating probabilistic extraction models and data mining to discover relations and patterns in text. In *Proceedings of HLT-NAACL 2006*, pages 296–303, New York, 2006.

Christiane Fellbaum, editor. *WordNet: An Electronic Lexical Database*. The MIT Press, 1998.

W.R. Gilks, S. Richardson, and D.J. Spiegelhalter. *Markov chain Monte Carlo in practice*. Chapman and Hall, London, UK, 1996.

Nanda Kambhatla. Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations. In *Proceedings of ACL-04*, Barcelona, Spain, 2004.

John Lafferty, Andrew McCallum, and Fernando Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of ICML-01*, pages 282–289. Morgan Kaufmann, San Francisco, CA, 2001.

Dong C. Liu and Jorge Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45:503–528, 1989.

Scott Miller, Heidi Fox, Lance Ramshaw, and Ralph Weischedel. A novel use of statistical parsing to extract information from text. In *Proceedings of NAACL-2000*, pages 226–233, Seattle, Washington, 2000.

Dat P. T. Nguyen, Yutaka Matsuo, and Mitsuru Ishizuka. Relation extraction from Wikipedia using subtree mining. In *Proceedings of AAAI-07*, pages 1414–1420, Vancouver, British Columbia, Canada, 2007.

Dat P. T. Nguyen, Yutaka Matsuo, and Mitsuru Ishizuka. Subtree mining for relation extraction from Wikipedia. In *Proceedings of HLT-NAACL 2007*, pages 125–128, Rochester, New York, 2007.

Judea Pearl. *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. Morgan Kaufmann Publishers Inc., San Francisco, CA, 1988.

Matthew Richardson and Pedro Domingos. Markov logic networks. *Machine Learning*, 62(1-2):107–136, 2006.

Benjamin Rosenfeld and Ronen Feldman. Using corpus statistics on entities to improve semi-supervised relation extraction from the Web. In *Proceedings of ACL-07*, pages 600–607, Prague, Czech Republic, June 2007.

Hirano Toru, Matsuo Yoshihiro, and Kikui Genichiro. Detecting semantic relations between named entities in text using contextual features. In *Proceedings of ACL-07*, pages 157–160, Prague, Czech Republic, June 2007.

Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella. Kernel methods for relation extraction. *Journal of Machine Learning Research*, 3:1083–1106, 2003.

Guodong Zhou, Jian Su, Jie Zhang, and Min Zhang. Exploring various knowledge in relation extraction. In *Proceedings of ACL-05*, pages 427–434, Ann Arbor, Michigan, 2005.