Statistical Relational Learning with Formal Ontologies

Achim Rettinger[1], Matthias Nickles[2], and Volker Tresp[3]

[1] Technische Universität München, Germany
achim.rettinger@cs.tum.edu
[2] University of Bath, United Kingdom
M.L.Nickles@cs.bath.ac.uk
[3] Siemens AG, CT, IC, Learning Systems, Germany
volker.tresp@siemens.com

Abstract. We propose a learning approach for integrating formal knowledge into statistical inference by exploiting ontologies as a semantically rich and fully formal representation of prior knowledge. The logical constraints deduced from ontologies can be utilized to enhance and control the learning task by enforcing description logic satisfiability in a latent multi-relational graphical model. To demonstrate the feasibility of our approach we provide experiments using real world social network data in form of a $\mathcal{SHOIN}(D)$ ontology. The results illustrate two main practical advancements: First, entities and entity relationships can be analyzed via the latent model structure. Second, enforcing the ontological constraints guarantees that the learned model does not predict inconsistent relations. In our experiments, this leads to an improved predictive performance.

1 Introduction

This paper focuses on the combination of statistical machine learning with ontologies specified by formal logics. In contrast to existing approaches to the use of constraints in machine learning (ML) and data mining, we exploit a semantically rich and fully formal representation of hard constraints which govern and support the stochastic learning task. Technically, this is achieved by combining the *Infinite Hidden Relational Model* (IHRM) approach to Statistical Relational Learning (SRL) with inference guided by the constraints implied by a *Description Logic* (DL) ontology used on the Semantic Web (SW). In this way, our approach supports a tight integration of formal background knowledge resulting in an *Infinite Hidden Semantic Models* (IHSM). The term *Semantic* in IHSM stands for this integration of "meaningful", symbolic knowledge which enables the use of deductive reasoning.

Benefits of the presented approach are (1) the analysis of known entity classes of individuals by means of clustering and (2) the completion of the knowledge base (KB) with uncertain predictions about unknown relations while considering constraints as background knowledge for the machine learning process. Thus, it is guaranteed that the learned model does not violate ontological constraints and

W. Buntine et al. (Eds.): ECML PKDD 2009, Part II, LNAI 5782, pp. 286–301, 2009.

the predictive performance can be improved . While there is some research on data mining for the SW, like instance-based learning and classification of individuals, considering "hard" constraints specified in the ontology during machine learning has hardly been tried so far or only in quite restricted and semi-formal ways (see Sec. 6 for related work). Even though we use a social network OWL DL ontology and settle on SRL as an apparently natural counterpart for logical constraints, our general approach is in no way restricted to DL or SRL and could be easily adapted to other formal and learning frameworks.

To provide an intuitive understanding of the presented approach we will use a simple example throughout this paper to illustrate the application of constraints in our learning setting: Consider a social network where, amongst others, the age of persons and the schools they are attending is partially known. In addition, an ontology designer specified that persons under the age of 5 are not allowed to attend a school. All this prior knowledge is provided in a formal ontology and the ultimate task is to predict unknown elements of this network.

The remainder of this paper is structured as follows: In Sec. 2 we specify an ontology in OWL DL that defines the taxonomy, relational structure and constraints. Next, we show how to infer a relational model from the ontology and transfer the relational model into an IHSM (Sec. 3). Then, we learn the parameters of this infinite model in an unsupervised manner while taking the constraints into account (Sec. 4). In Sec. 5 the IHSM is evaluated empirically using a complex dataset from the Semantic Web. Finally, we discuss related work in Sec. 6 and conclude in Sec. 7.

2 Formal Framework

Our approach requires the specification of formal background knowledge and formal constraints for the learning process. We do so by letting the user of the proposed machine learning algorithm specify a formal ontology or use an existing ontology e.g. from the SW. In computer science, an ontology is the formal representation of the concepts of a certain domain and their relations. In the context of the (Semantic) Web and thus also in our approach, such an ontology is typically given as a so-called *TBox* and a *ABox*, each of which consists of a number of description logic formulas. The TBox comprises conceptual knowledge (i.e., knowledge about classes and their relations), whereas the ABox comprises knowledge about the instances of these classes. In our context, the given ontology and also all knowledge which is a logical consequence from it constitutes the "hard" knowledge for the learning process (i.e., knowledge which cannot be overwritten during learning), as described later.

However, our approach is not restricted to ontologies, but works in principle with all sorts of formal knowledge bases. We are using ontologies mainly because there is an obvious relatedness of clustering and ontological classification, and because formal languages, reasoners, editors and other tools and frameworks for ontologies on the SW are standardized and widely available. Consequently, we use a description logic for our examples. This is not only because ontologies

and other formal knowledge on the Web (which is our application here) are usually represented using DLs, but also because the standard DL which we use is a decidable fragment of first-order logic (FOL) for which highly optimized reasoners exist.

We settle on the $\mathcal{SHOIN}(D)$ [1] description logic, because entailment in the current Semantic Web standard ontology language OWL DL can be reduced to $\mathcal{SHOIN}(D)$ knowledge base satisfiability. We could likewise work with OWL DL syntax directly, but that wouldn't have any technical advantages and would just reduce the readability of our examples. Our approach requires that the satisfiability or consistency of ontologies can be checked, which is a standard operation of most automated reasoning software for the SW. Allowing to check the satisfiability means that the reasoner is able to check whether a given KB (ontology) has a model. On the syntactic level, satisfiability corresponds to consistency, i.e., there are no sentences in the given ontology which contradict each other. The following specifies the syntax of $\mathcal{SHOIN}(D)$. Due to lack of space, please refer to [1] for a detailed account of this language.

$$\begin{aligned} C \rightarrow A|\neg C|C_1 \sqcap C_2|C_1 \sqcup C_2|\exists R.C|\forall R.C \quad & | \\ \geq nS|\leq nS|\{a_1,...,a_n\}|\geq nT|\leq nT| & \\ \exists T_1,...,T_n.D|\forall T_1,...,T_n.D|D \rightarrow d|\{c_1,...,c_n\} & \end{aligned}$$

Here, C denote *concepts*, A denote *atomic concepts*, R denote *abstract roles* or *inverse roles* of abstract roles (R^-), S denote abstract *simple roles*, the T_i denote *concrete roles*, d denotes a concrete *domain predicate*, and the a_i / c_i denote abstract / concrete *individuals*.

A $\mathcal{SHOIN}(D)$ *ontology* or *knowledge base* is then a non-empty, finite set of TBox axioms and ABox assertions $C_1 \sqsubseteq C_2$ (inclusion of concepts), $Trans(R)$ (transitivity), $R_1 \sqsubseteq R_2$, $T_1 \sqsubseteq T_2$ (role inclusion for abstract respectively concrete roles), $C(a)$ (concept assertion), $R(a,b)$ (role assertion), $a = b$ (equality of individuals), and $a \neq b$ (inequality of individuals). Concept equality is denoted as $C_1 \equiv C_2$ which is just an abbreviation for mutual inclusion, i.e., $C_1 \sqsubseteq C_2, C_2 \sqsubseteq C_1$. Defining a semantics of $\mathcal{SHOIN}(D)$ is not required within the scope of this work, the canonical semantics which we assume in this work can be found, e.g., in [1].

2.1 Constraints

Constraints in the sense of this work are actually just formal statements. Our approach is expected to work with all kinds of logical frameworks which allow for satisfiability (or consistency) checks over some given set of logical sentences, for example an ontology. This set of given statements is denoted as the KB in the further course of this paper. Formally, we define a set of constraints C to be the deductive closure $\Theta(KB)$ of a given knowledge base KB, with $\Theta(KB) = \{c|KB \models c\}$. The deductive closure contains not only explicitly given knowledge (the knowledge base KB), but also all logical sentences which can be derived from

the KB via deductive reasoning. E.g., if the KB would contain the sentences $\neg a$ and $\neg a \rightarrow b$, the deductive closure would also contain b.

The application-specific constraint set which we use as an OWL DL ontology is similar to the well-known *Friend-Of-A-Friend* (FOAF) social network schema, together with additional constraints which will be introduced later. The following ontology SN comprises only a fragment of the full FOAF-like ontology we have used (with DOB meaning "date of birth" and $hasBD$ meaning "has birthday".).

$$\begin{array}{r|c|l} Person \sqsubseteq Agent & knows^{-} \sqsubseteq knows & \exists knows.\top \sqsubseteq Person \\ \top \sqsubseteq \forall knows.Person & \exists hasBD.\top \sqsubseteq Person & \top \sqsubseteq \forall hasBD.DOB \\ \top \sqsubseteq \leq 1\ hasBD & \top \sqsubseteq \geq 1\ hasBD & \exists yearValue.\top \sqsubseteq DOB \\ \top \sqsubseteq \forall yearValue.gYear & \top \sqsubseteq \leq 1\ yearValue & \top \sqsubseteq \forall attends.School \end{array}$$

These axioms mainly express certain properties of binary relations (so-called *roles*) between classes. For example, $\top \sqsubseteq \forall attends.School$ specifies that in our example ontology the range (target set) of role $attends$ is $School$.

In addition to these, we provide the machine learning algorithm with an ABox which models an incomplete social network. The later machine learning task consists essentially in a (uncertain) completion of this given network fragment. An example for such additional individuals-governing constraints A: $tim : Person, tina : Person, tom : Person$; $(tina, tim) : knows, (tina, tom) : knows$.

Note, that these relationships among persons cannot be weakened or overwritten by the learning process, even if they contradict observed data. They need to be provided manually by the KB engineer. As further constraints, we assume some specific properties G of the analyzed social network. The following set of axioms expresses that no one who is younger than six years goes to school. At this, $UnderSixYearsOld$ is the class which contains persons with an age less than six years (calculated from the given dates of birth):

$$Pupil \sqsubseteq Person \mid Pupil \sqsubseteq \neg UnderSixOld \mid Pupil \sqsubseteq \exists\ attendsSchool$$

The complete set of given formal and definite knowledge for our running example is then $C = \Theta(SN \sqcup A \sqcup G)$.

Example Data: The set of data used as examples for the learning tasks takes the form of ABox assertions. But in contrast to the ABox knowledge in set A above, an example here might turn out to be wrong. We also do not demand that examples are mutually consistent, or consistent with the ontology. In order to maintain compatibility with the expected input format for relational learning, we restrict the syntax of examples to the following two description logic formula patterns:

$$instance : category$$
$$(instance_a, instance_b) : role$$

At this, roles correspond to binary relations. The set of all example data given as logical formulas is denoted as D.

3 Infinite Hidden Semantic Models

The proposed *Infinite Hidden Semantic Model* (IHSM) is a machine learning algorithm from the area of SRL [2]. The novelty is its additional ability to exploit formal ontologies as prior knowledge given as a set of logical formulas. In our case, the constraints are provided as a $\mathcal{SHOIN}(D)$ ontology with a TBox and an ABox as just described in the previous section. In traditional ML, prior knowledge is just specified by the likelihood model and the prior distributions, parameters of the learning algorithm or selection of features.

In this section, we first show how the ontology from Sec. 2 defines a *Relational Model* (RM) which is the basis for an *Infinite Hidden Relational Model* (IHRM). Then, the IHSM is generated by constraining the IHRM appropriately.

3.1 Relational Models

First an abstract RM of concepts and roles defined in our social network ontology is created. Based on the TBox axioms given by the ontology we can create a simple sociogram as depicted in Fig. 1. A sociogram consists of three different elements: concept individuals (individuals that are instances of a concept (e.g. $tim : Person$)), attribute instances (relations between a concept and a literal (e.g. $tina : hasImage$)), role instances (relations between concepts (e.g. $(tina, tim) : knows$)). Please note that many TBox elements first need to be deduced from the ontology, so that all individuals can be assigned to its most specific concepts. This process is known as *realization* in DL reasoning. Fig. 2 shows the full RM we use for experiments in Sec. 5.

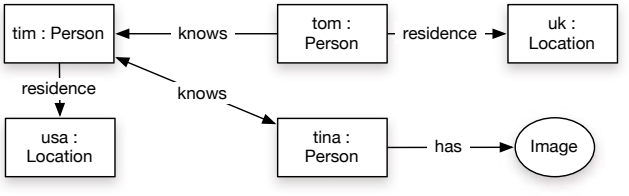

Fig. 1. Partial sociogram of the LJ-FOAF-domain

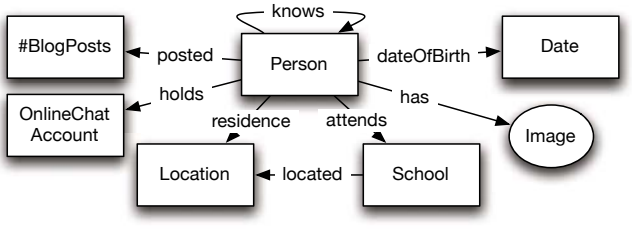

Fig. 2. Relational Model of the LJ-FOAF-domain

3.2 Infinite Hidden Relational Models

Following [3] and [4] we extend the RM to a Hidden Relational Model (HRM) by assigning a hidden variable denoted as Z_i^c to each individual i of concept c with current state k. Given that the hidden variables have discrete probability distributions they can be intuitively interpreted as clusters Z where similar individuals of the same concept c (in our case similar persons, locations, schools,...) are grouped in one specific component k. These assignments of latent states specify the component one individual is assigned to.

The resulting HRM of the sociogram shown in Fig. 1 is depicted in Fig. 3. Following the idea of hidden variables in *Hidden Markov Models* (HMMs) or *Markov Random Fields*, those additional variables can be thought of as unknown properties (roles or attributes) of the attached concept. We assume that all attributes of a concept only depend on its hidden variable and roles depend on two hidden variables of the two concepts involved. This implies that if the hidden variables were known, attributes and roles can be predicted independently. In addition, the hidden variables in the IHSM incorporate restrictions in the form of constraints imposed by the ontology (see Sec. 3.3).

Considering the HRM model shown in Fig. 3, information can now propagate via those interconnected hidden variables Z. E.g. if we want to predict whether *tom* with hidden state Z_3^1 might know *tina* (Z_2^1) we need to consider a new relationship $R_{3,2}$. Intuitively, the probability is computed based on (i) the attributes A_3^1 and A_1^1 of the latent states of immediately related persons Z_3^1 and Z_2^1; (ii) the known relations associated with the persons of interest, namely the role *knows* and *residence* $R_{2,1}$, $R_{3,1}$ and $R_{3,2}$; (iii) higher-order information indirectly transferred via hidden variables Z_3^1 and Z_2^1. In summary, by introducing hidden variables, information can globally distribute in the HRM. This reduces the need for extensive structural learning, which is known to be difficult.

Critical parameters in the HRM are the number of states in the various latent variables, which might have to be tuned as part of a complex optimization routine. A solution here offers the IHRM, that was introduced by [4] and [3]. In the IHRM, a hidden variable has a potentially infinite number of states and an estimate of the optimal number of states is determined as part of the inference process.

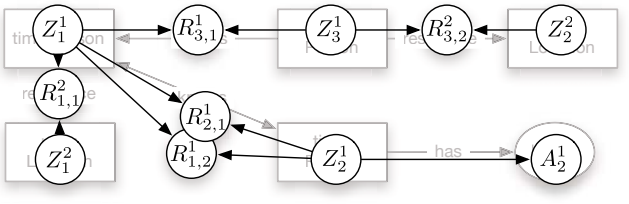

Fig. 3. Hidden relational model of the sociogram defined in Fig. 1

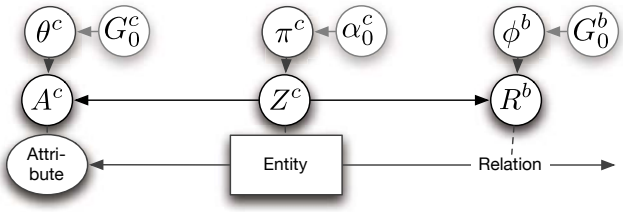

Fig. 4. Parameters of an IHRM

Finally, we need to define the remaining variables, their probability distributions and model parameters[1]. The most important parameters in our case are shown in Fig. 4. The state k of Z_i^c specifies the cluster assignment of the concept (aka entity class) c. K denotes the number of clusters in Z. Z is sampled from a multinomial distribution with parameter vector $\pi = (\pi_1, \ldots, \pi_K)$, which specifies the probability of a concept belonging to a component, i.e. $P(Z_i = k) = \pi_k$. π is referred to as mixing weights, and is drawn according to a truncated stick breaking construction with a hyperparameter α_0. α_0 is referred to as a *concentration parameter* in Dirichlet Process (DP) mixture modeling and acts as a tuning parameter that influences K. K is also limited by a truncation parameter that specifies the maximum number of components per cluster for each entity class.

Attributes A^c are generated from a Bernoulli distribution with parameters θ_k. For each component, there is an infinite number of mixture components θ_k. Each person in the component k inherits the mixture component, thus we have: $P(G_i = s|Z_i = k, \Theta) = \theta_{k,s}$. These mixture components are independently drawn from a prior G_0. The base distributions G_0^c and G_0^b are conjugated priors with hyperparameters β^c and β^b.

The truth values for the role $R_{i,j}$ involving two persons (i and j) are sampled from a binomial distribution with parameter $\phi_{k,\ell}$, where k and ℓ denote cluster assignments of the person i and the person j, respectively. $\phi_{k,\ell}^b$ is the correlation mixture component indexed by potentially infinite hidden states k for c_i and ℓ for c_j, where c_i and c_j are indexes of the individuals involved in the relationship class b. Again, G_0^b is the Dirichlet Process base distribution of a role b. If an individual i is assigned to a component k, i.e. $Z_i = k$, the person inherits not only θ_k, but also $\phi_{k,\ell}, \ell = \{1, \ldots, K\}$.

3.3 Infinite Hidden Semantic Models

The IHSM is based on the idea that formal constraints can be imposed on the correlation mixture component $\phi_{k,\ell}$ and thus restrict possible truth values for the roles $R_{i,j}$. This, amongst others, imposes constraints on the structure of the underlying ground network or, more specifically in our application, the structure

[1] Please note that we cannot focus on the technical details of an lHRM and need to refere the reader to [4] and [3] for a more detailed introduction.

of the sociogram. Recap the simple example from Sec. 1: According to this a person i known to be younger than 5 years old should not be attending any school j. The IHSM will extract this information from the ontology and set the correlation mixture component $\phi_{k,\ell}$ at entries representing according relations from *Person* component k to *School* component ℓ to 0. Here, k and ℓ denote the components the person i and the school j are assigned to. This eliminates inconsistent structural connection from the underlying ground network. More generally, all connections $R_{i,j}$ between two components k and ℓ where inconsistent individuals i and j are (partial) member of are considered void.

However, this redirection of relations by the latent variables allows IHSM not only to restrict possible connections in the ground network but makes this restriction influence the likelihood model itself. By restricting ϕ, π is affected as well. Ultimately, cluster assignments Z are influenced and information can globally propagate through the network and influence all ϕ, π and θ (see Sec. 3.2).

While this section (3.3) focused on a conceptual description of IHSM the algorithm will be specify in detail in the next section (4) before Sec. 5 presents experimental results.

4 Learning, Constraining and Predictions

The key inferential problem in the IHSM is to compute the joint posterior distribution of unobservable variables given the data. In addition, we need to avoid inconsistent correlation mixture components ϕ during learning. As computation of the joint posterior is analytically intractable, approximate inference methods need to be considered to solve the problem. We use the blocked Gibbs sampling (GS) with truncated stick breaking representation [5] a Markov chain Monte Carlo method to approximate the posterior.

Let D be the set of all available observations (observed example data, each represented as a logical formula as defined in 2.1), and let $Agents = Agent^I$ be the set of all instances of category *Agent* under interpretation I - that is informally, all persons which contribute to the social network. At each iteration, we first update the hidden variables conditioned on the parameters sampled in the last iteration, and then update the parameters conditioned on the hidden variables. So, for each entity class

1. Update hidden variable Z_i^c for each e_i^c: Assign to component with probability proportional to:

$$\pi_k^{c(t)} P(A_i^c | Z_i^{c(t+1)} = k, \theta^{c(t)}) \times \prod_{b'} \prod_{j'} \quad P(R_{i,j'}^{b'} | Z_i^{c(t+1)} = k, Z_{j'}^{c_{j'}(t)}, \ \phi^{b'(t)})$$

2. Update $\pi^{c(t+1)}$ as follows:

   (a) Sample $v_k^{c(t+1)}$ from

Beta($\lambda_{k,1}^{c(t+1)}, \lambda_{k,2}^{c(t+1)}$) for $k = \{1, \ldots, K^c - 1\}$ with

$$\lambda_{k,1}^{c(t+1)} = 1 + \sum_{i=1}^{N^c} \delta_k(Z_i^{c(t+1)}),$$
$$\lambda_{k,2}^{c(t+1)} = \alpha_0^c + \sum_{k'=k+1}^{K^c} \sum_{i=1}^{N^c} \delta_{k'}(Z_i^{c(t+1)}),$$

and set $v_{K^c}^{c(t+1)} = 1$. $\delta_k(Z_i^{c(t+1)})$ equals to 1 if $Z_i^{c(t+1)} = k$ and 0 otherwise.

(b) Compute $\pi^{c(t+1)}$ as: $\pi_1^{c(t+1)} = v_1^{c(t+1)}$ and

$$\pi_k^{c(t+1)} = v_k^{c(t+1)} \prod_{k'=1}^{k-1} (1 - v_{k'}^{c(t+1)}), \quad k > 1.$$

3. Update θ:

$$\theta_k^{c(t+1)} \sim P(\cdot | A^c, Z^{c(t+1)}, G_0^c)$$

4. Constrain ϕ to satisfiable relations:
For entity cluster k, let $F_{ext}^k = F^k \cap \{(\mathbf{e_m}, \mathbf{e_n}) : \mathbf{r} | e_m, e_n \in Agents, r \in R, m \neq n\}$ be the set of those logical formulas in the example data set which represent some relation ("role") r between two different individuals (persons) e_m and e_n where person e_m is assigned to component k already and e_n is assigned to a component ℓ. To keep the notation compact, we spell out role instances $(\mathbf{e_1}, \mathbf{e_2}) : \mathbf{r}$ only asymmetrically (i.e., we omit $(\mathbf{e_2}, \mathbf{e_1}) : \mathbf{r}$ if we have covered the case $(\mathbf{e_1}, \mathbf{e_2}) : \mathbf{r}$). Let $F_k \subseteq D$ be the set of *all* example formulas which have already been used to learn component k so far, that is, the subset of the data D which has been used for forming that cluster until now. Let furthermore $\vartheta(e, D)$ be the set of all sampled formulas in D where the person e appears, i.e., $f \in \vartheta(e, D)$ iff $f \in D \wedge (f \equiv \mathbf{e} : \mathbf{c} \vee f \equiv (\mathbf{e}, \mathbf{e_x}) : \mathbf{r}$ for some $c \in C$, $e_x \in Agents$ and $r \in R$. We use $\rho(e, j)$ to express that a certain entity e has already been assigned to a certain component j. The following steps are now used in order to check whether component k is usable w.r.t. the given set of logical constraints C:

(a) Identify the largest subset F_{clean}^k of formulas within F_{ext}^k which is consistent with C and the set of example data about person e_i^c:

$$F_{clean}^k \subseteq 2^{F_{ext}^k}, \exists\, \mathcal{I}, \mathcal{I} \models F_{clean}^k \cup \vartheta(e_i^c, D) \cup C,$$
$$\forall F \subseteq 2^{F_{ext}^k}, \exists\, \mathcal{I}, \mathcal{I} \models F \cup \vartheta(e_i^c, D) \cup C : F \subseteq F_{clean}^k$$

($\mathcal{I} \models X$ expresses that the set of logical sentences X is satisfiable, $\mathcal{I}$ being an interpretation).

(b) Verify whether F^k_{clean}, the formulas which have been used to learn "related" other clusters, $\vartheta(e^c_i, D)$ and the constraints are consistent in sum if we replace in F^k_{clean} the names of all persons which are assigned to components other than k with the name of person e^c_i.

Let $F^k_{upd} = \{(\mathbf{e^c_i}, \mathbf{e_m}) : \mathbf{r} | (\mathbf{e_n}, \mathbf{e_m}) : \mathbf{r} \in F^k_{ext}\}$ be the latter set of formulas. Furthermore, let $F^k_{rel} = \bigcup_{j \neq k, \rho(e_m,k), (\mathbf{e_m}, \mathbf{e_n}):\mathbf{r} \in F^j} F^j$ be the set of all formulas in all other components than k which "relate" to component k using role formulas. The overall consistency check for component k yields a positive result *iff*

$$\exists\, \mathcal{I}, \mathcal{I} \models \vartheta(e^c_i, D) \cup F^k_{upd} \cup F^k_{rel} \cup C \wedge F^k_{clean} \neq \emptyset$$

Where the consistency check described above yielded a positive result:

$$\phi^{b(t+1)}_{k,\ell} \sim P(\cdot | R^b, Z^{(t+1)}, G^b_0).$$

After the GS procedure reaches stationarity the role of interest is approximated by looking at the sampled values. Here, we only mention the simple case where the predictive distribution of the existence of a relation $R_{i,j}$ between to known individuals i, j is approximated by $\phi^b_{i',j'}$ where i' and j' denote the cluster assignments of the objects i and j, respectively.

5 Experiments

The increasing popularity of social networking services like MySpace and Facebook has fostered research on social network analysis in the last years. The immense number of user profiles demands for automated and intelligent data management capabilities, e.g. formal ontologies. While data mining techniques can handle large amounts of simple facts, little effort has been made to exploit the semantic information inherent in social networks and user profiles. There is almost no work on statistical relational learning with formal ontologies in general and with SW data in particular. The lack of experiments on large and complex real world ontologies is not only due to the absence of algorithms but also due to missing suitable datasets. In this section we will present both, a large and complex SW dataset and the methodology of how to apply IHSM in practice. Ultimately, we evaluate our approach by presenting results of an empirical comparison of IHSM and IHRM in this domain.

5.1 Data and Methodology

As mentioned before our core ontology is based on Friend of a Friend (FOAF) data. The purpose of the FOAF project is to create a web of machine-readable pages describing people, the links between them and the things they create and do. The FOAF ontology is defined using OWL DL/RDF(S) and formally specified in the FOAF Vocabulary Specification 0.91[2]. In addition, we make use of

[2] http://xmlns.com/foaf/spec/

Table 1. No. of individuals, no. of instantiated roles and final number of components

Concept	#Indivi.	Role	#Inst.	#C. IHRM	#C. IHSM
Location	200	*residence*	514	18	17
School	747	*attends*	963	**36**	**48**
OnlineChatAccount	5	*holdsAccount*	427	4	4
Person	638	*knows*	8069	**38**	**45**
		hasImage	574		
Date	4	*dateOfBirth*	194	**4**	**2**
#BlogPosts	5	*posted*	629	4	4

further concepts and roles which are available in the data (see Sec. 2.1). We gathered our FOAF dataset from user profiles of the community website LiveJournal.com[3] (This specific ontology will be called LJ-FOAF from now on).

All extracted concepts and roles are shown in Fig. 2. Tab. 1 lists the number of different individuals (left column) and their known instantiated roles (middle column). Please note that *Date* and *#BlogPosts* are reduced to a small number of discrete states. As expected for a social networks *knows* is the primary source of information. This real world data set offers both, a sufficiently large set of individuals for inductive learning and a formal ontology specified in RDFS and OWL. However, while LJ-FOAF offers a taxonomy there are no complex constraints given. Thus, to demonstrate the full potential of IHSM, we additionally added constraints that are not given in the original ontology (see Sec. 2.1).

To implement all features of IHSM we made use of additional open source software packages: The Semantic Web framework Jena[4] is used to load, store and query the ontology and Pellet[5] provides the OWL DL reasoning capabilities. This outlines the workflow: First, the TBox axioms are designed and loaded into Jena. Next, all ABox assertions are added and loaded into Jena. Then, by using the taxonomy information from the ontology and the ABox assertions we extract the RM as described in Sec. 3.1. This RM is transferred into a IHSM by adding hidden variables and parameters, accordingly. Finally, the parameters are learned from the data, while constraints are constantly checked as shown in Sec. 4.

In our experiments the standard setting for the truncation parameter were $\#Individuals/10$ for entity classes with over 100 instances and $\#Individuals$ for entity classes with less individuals. The standard iterations of the Gibbs sampler are 100. We did not engage in extensive parameter tuning because the purpose of this evaluation is to examine the influence of the constraints and not optimal predictive performance. Thus, we fixed $\alpha_0 = 5$ for every entity class and $\beta_0 = 20$ for every relationship class.

5.2 Results

We will now report our results on learning and constraining with the LJ-FOAF data set.

[3] http://www.livejournal.com/bots/
[4] http://jena.sourceforge.net/
[5] http://pellet.owldl.com/

Computational Complexity: The additional consistency check for every individual per iteration made training slower by approximately a factor of 6 if performed with Jena and Pellet. After implementing a non-generic constraining module optimized for the simple example introduced in Sec. 1 we could reduce the additional computation considerably. A comparison between IHSM and IHRM for different truncation parameter settings is given in Fig. 5. Obviously, there is almost no computational overhead in the latter case.

Evaluating the convergence of the cluster sizes is another interesting aspect in the comparison of IHSM and IHRM. Fig. 6 shows the number of individuals for the two largest components of the entity cluster Z^{Person} plotted over Gibbs sampler iterations for one exemplary training run. Apparently, the constraining does not affect the convergence speed which is desirable.

Cluster Analysis: An interesting outcome of the comparison of IHRM and IHSM is the number of components per hidden variable after convergence (see Table 1 right column). In both cases, if compared to the initialization, Gibbs sampling converged to a much smaller number of components. Most of the individuals were assigned to a few distinct components leaving most of the remaining components almost empty. There is a noticeable difference between IHRM and IHSM concerning the concepts *School* and *Person* which needed more components after training with IHSM (see bold numbers in Table 1). A closer analysis of the components revealed that IHSM generated additional components for inconsistent individuals, because both concepts are affected by constraints. However, the last concept affected by the constraints (*Date*) has fewer components. Here, IHSM divided more generally into age groups "too young" and "old enough" which also reflects the constraints. This demonstrates that the restriction of roles does influence the states of the latent variables.

Fig.7 compares the learned parameter ϕ^{attend} of IHRM to the one learned by IHSM. A brighter cell indicates stronger relations between two components. Although hard to generalize, a cell with 50% gray might indicate that no

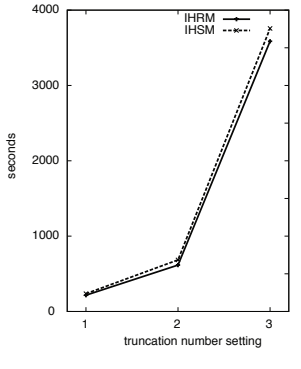

Fig. 5. Running time

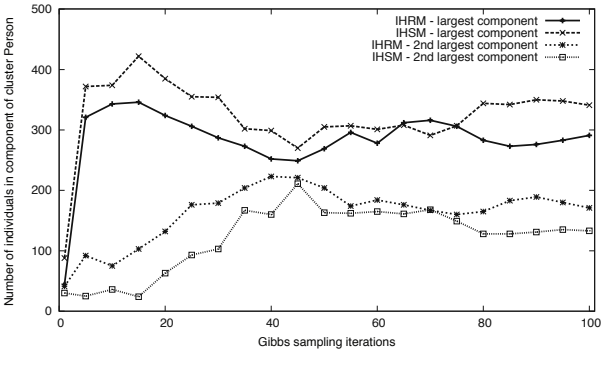

Fig. 6. Convergence of the two largest components

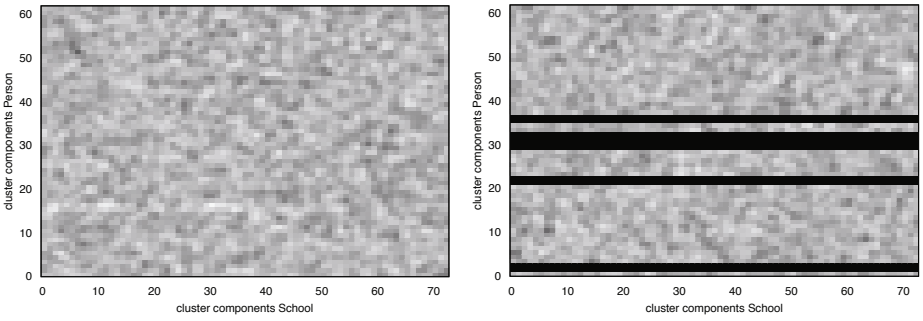

Fig. 7. Correlation mixture component ϕ^{attend} for each combination of components Z^{Person} and Z^{School}. Left: without constraining (IHRM). Right: with constraining (IHSM).

significant probabilistic dependencies for individuals in this component are found in the data. The most obvious results are the rows with black cells only which represent *Person* components that have no relation to any school. In fact, all of those cells contained at least one persons that conflicted with the ontology by having specified an age under 5. This proves that one of the main goals of IHSM is achieved, namely the exploitation of constraints provided by the ontology.

Note that the learned clusters can also be used to extract symbolic and uncertain knowledge and feed it back to the ontology. This is a promising direction of future research.

Predictive Performance: Given LJ-FOAF data for social network analysis one could for instance want to predict "who knows who" in case either this information is unknown or the systems wants to recommend new friendships. Other relations that could be interesting to predict in case they are unknown are the school someone attends/attended or the place he lives/lived. Furthermore one could want to predict unspecified attributes of certain persons, like their age. The purpose of this section is not to show superior predictive performance of IHRM compared to other multi-relational learning algorithms. This has been evaluated before, e.g. in [3]. Here, we want to show the influence of constraining on the predictive performance for IHSM compared to IHRM.

We ran a 5-fold cross validation to evaluate the predictions of different relationship classes. In specific, the non-zero entries of the relationship matrix to be predicted were randomly split in 5 parts. Each part was once used for testing while the remaining parts were used for training. The entries of each testing part were set to zero (unknown) for training and to their actual value of 1 for testing. Each fold was trained with 100 iterations of the Gibbs sampler, where 50 iterations are discarded as the burn-in period. After this, the learned parameters are recorded every fifth iteration. In the end we use the 10 recorded parameter sets to predict the unknown relationship values, average over them and calculate

Table 2. Predictive performance for different LJ-FOAF roles: AUC and 95% confidence intervals

Role	attends	dateOfBirth	knows
IHRM	0.577 (±0.013)	0.548 (±0.018)	0.813 (±0.005)
IHSM	**0.608** (±0.017)	**0.561** (±0.011)	**0.824** (±0.002)

the area under the ROC curve (AUC) as our evaluation measure[6]. Finally we average over the 5 folds and calculate the 95% confidence interval.

The obvious roles to evaluate are *attends* and $dateOfBirth$. Both are constrained by the ontology, so IHSM should have an advantage over IHRM because it cannot predict any false positives. The results in Table 2 confirm this observation. In both cases IHSM did outperform IHRM. A less obvious outcome can be examined from the influence of the constraining on a relationship that is not directly constrained by the ontology like *knows*. Still, in our experiments IHSM showed a slight advantage over IHRM. Thus, there seems to be a positive influence of the background knowledge, although a lot of users specify an incorrect age. However, there is the potential that the opposite may occur likewise. If the given constraints are conflicting with the empirical evidence there could even be a decrease in predictive performance. It is the ontology designers choice to decide whether to enforce a constraint that conflicts with the observed evidence.

Considering the numerous ongoing efforts concerning ontology learning for the Semantic Web more data sets with complex ontologies should become available in the near future. Thus, we expect to achieve more definite results of IHSM in those domains.

6 Related Work

Very generally speaking, our proposed method aims at combining machine learning with formal logic. So far, machine learning has been mainly approached either with statistical methods, or with approaches which aim at the inductive learning of formal knowledge from examples which are also provided using formal logic. The most important direction in this respect is *Inductive Logic Programming* (ILP) [6]. *Probabilistic-* and *Stochastic Logic Programming* (e.g., [7]) (SLP) are a family of ILP-based approaches which are capable of learning stochastically weighted logical formulas (the weights of formulas, respectively). In contrast to that, our approach learns probability distributions with the help of a given, formal theory which acts as a set of hard constraints. To the best of our knowledge, this direction is new. What (S)ILP and our approach have in common is that our method also uses examples formalized in a logic language as data. There

[6] Please note that SW data has no negative samples, because zero entries do not represent negative relations but unknown ones (open world assumption). Still, the AUC is appropriate because it has been shown to be a useful measure for probabilistic predictions of binary classification on imbalanced data sets.

are also some approaches which build upon other types of "uncertain logic", for example [8]. Although (S)ILP and SRL are conceptually very closely related and often subsumed under the general term *relational learning*, SRL still is rarely integrated with formal logic or ontologies as prior knowledge. [9] use ontologies in an similar model but only use taxonomic information as additional "soft" knowledge (i.e., knowledge which can be overwritten during the learning process) in the form of features for learning. They do not restrict their results using formal hard constraints. One exception are *Markov Logic Networks* [10] which combine First Order Logic and *Markov Networks* and learn weights of formulas.

Surprisingly there are also hardly any applications of (pure) SRL algorithms to (SW) ontologies. The few examples, e.g. [11], [12], do not consider formal constraints. There are various approaches to the learning of categories in formal ontologies from given instance data and/or similar categories (e.g., [13]). However, these approaches do not allow for the statistical learning of relations in the sense of SRL and their aims are all in all more related to those of ILP than to our learning goals. Although there are applications of SRL to social networks, such as [14], none of those approaches uses a formal ontology or any other kind of formal knowledge. Furthermore, the social networks examined in this work are mostly significantly less complex in regard of the underlying relation model.

The use of *hard constraints* for clustering tasks in purely statistical approaches to learning, as opposed to the ubiquitous use of "soft" prior knowledge, has been approached in, e.g., [15]. A common characteristic of these approaches is that they work with a relatively narrow, semi-formal notion of constraints and do not relate constraints to relational learning. In contrast to these efforts, our approach allows for rich constraints which take the form of a OWL DL knowledge base (with much higher expressivity). The notion of forbidden pairings of data points (*cannot-link* constraints [15]) is replaced with the more general notion of logical (un-)satisfiability w.r.t. formal background knowledge.

7 Conclusions and Future Work

In the presented approach, we explored the integration of formal ontological prior knowledge into machine learning tasks. We introduced IHSM and provided empirical evidence that hard constraints cannot only improve predictive performance of unknown roles, which are directly affected by the constraints, but also unconstraint roles via IHSMs latent variables.

In general we are hope to see more work on inductive learning with SW ontologies and on the other hand complex Semantic Web ontologies that can be supplemented with uncertain evidence. For the IHSM in particular, future work will concern a detailed theoretical analysis of the effect of constraining on clusters. Refining the ontology by extracting formal knowledge from the latent model structure is another promising research direction. As mentioned before we intend to obtain additional experimental evidence concerning computational complexity and predictive performance as soon as more suitable ontologies become available. We expect that the increased research on semantic technologies

will soon result in those suitable formal ontologies that contain both, complex consistency reasoning tasks and large sets of instances.

References

1. Horrocks, I., Patel-Schneider, P.F.: Reducing owl entailment to description logic satisfiability. Journal of Web Semantics, 17–29 (2003)
2. Getoor, L., Taskar, B. (eds.): Introduction to Statistical Relational Learning. The MIT Press, Cambridge (2007)
3. Xu, Z., Tresp, V., Yu, K., Kriegel, H.P.: Infinite hidden relational models. In: Proceedings of the 22nd International Conference on Uncertainity in Artificial Intelligence, UAI 2006 (2006)
4. Kemp, C., Tenenbaum, J.B., Griffiths, T.L., Yamada, T., Ueda, N.: Learning systems of concepts with an infinite relational model. In: Proc. 21st Conference on Artificial Intelligence (2006)
5. Ishwaran, H., James, L.: Gibbs sampling methods for stick breaking priors. Journal of the American Statistical Association 96(453), 161–173 (2001)
6. Lisi, F.A., Esposito, F.: On Ontologies as Prior Conceptual Knowledge in Inductive Logic Programming. In: ECML PKDD 2008 Workshop: Prior Conceptual Knowledge in Machine Learning and Knowledge Discovery PriCKL 2007 (2007)
7. Raedt, L.D., Kersting, K.: Probabilistic logic learning. SIGKDD Explor. Newsl. 5(1), 31–48 (2003)
8. Carbonetto, P., Kisynski, J., de Freitas, N., Poole, D.: Nonparametric bayesian logic. In: Proc. 21st UAI (2005)
9. Reckow, S., Tresp, V.: Integrating Ontological Prior Knowledge into Relational Learning. In: NIPS 2008 Workshop: Structured Input - Structured Output (2008)
10. Richardson, M., Domingos, P.: Markov logic networks. Journal of Machine Learning Research 62, 107–136 (2006)
11. Kiefer, C., Bernstein, A., Locher, A.: Adding Data Mining Support to SPARQL via Statistical Relational Learning Methods. In: Bechhofer, S., Hauswirth, M., Hoffmann, J., Koubarakis, M. (eds.) ESWC 2008. LNCS, vol. 5021, pp. 478–492. Springer, Heidelberg (2008)
12. Fanizzi, N., d'Amato, C., Esposito, F.: Induction of classifiers through non-parametric methods for approximate classification and retrieval with ontologies. International Journal of Semantic Computing 2(3), 403–423 (2008)
13. Fanizzi, N., D'Amato, C., Esposito, F.: A multi-relational hierarchical clustering method for datalog knowledge bases. In: An, A., Matwin, S., Raś, Z.W., Ślęzak, D. (eds.) Foundations of Intelligent Systems. LNCS (LNAI), vol. 4994, pp. 137–142. Springer, Heidelberg (2008)
14. Xu, Z., Tresp, V., Rettinger, A., Kersting, K.: Social network mining with non-parametric relational models. In: Advances in Social Network Mining and Analysis - the Second SNA-KDD Workshop at KDD 2008 (2008)
15. Davidson, I., Ravi, S.S.: The complexity of non-hierarchical clustering with instance and cluster level constraints. Data Min. Knowl. Discov. 14(1), 25–61 (2007)