Learning to Propagate Knowledge in Web Ontologies

Pasquale Minervini[1], Claudia d'Amato[1], Nicola Fanizzi[1], Volker Tresp[2]

[1] Department of Computer Science - University of Bari, Italy
{pasquale.minervini|claudia.damato|nicola.fanizzi}@uniba.it
[2] Siemens AG, Corporate Technology, Munich, Germany
volker.tresp@siemens.com

Abstract. The increasing availability of structured machine-processable knowledge in the WEB OF DATA calls for machine learning methods to support standard pattern matching and reasoning based services (such as query-answering and inference). Statistical regularities can be efficiently exploited to overcome the limitations of the inherently incomplete knowledge bases distributed across the Web. This paper focuses on the problem of predicting missing class-memberships and property values of individual resources in Web ontologies. We propose a transductive inference method for inferring missing properties about individuals: given a class-membership/property value prediction problem, we address the task of identifying relations encoding similarities between individuals, and efficiently propagating knowledge across their relations.

1 Introduction

Standard query answering and reasoning services for the Semantic Web [2] (SW) mainly rely on deductive inference. However, purely deductive inference suffers from several limitations [20]: reasoning tasks might be computationally complex, do not always address uncertainty and only rely on axiomatic prior knowledge. However, purely deductive reasoning with SW representations suffers from several limitations owing to its complexity and the inherent incompleteness and incoherence of distributed knowledge bases (KBs) in a Web-scale scenario modeled by formal ontologies. In such a context many complex tasks (e.g. query answering, clustering, ranking, etc.) are ultimately based on assessing the truth of ground facts. Deciding on the truth of specific facts (assertions) in SW knowledge bases requires to take into account the *open-world* form of reasoning adopted in this context: a failure on ascertaining the truth value of a given fact does not necessarily imply that such fact is false, but rather that its truth value cannot be deductively inferred from the KB (e.g. because of incomplete or uncertain knowledge). Other issues are related to the distributed nature of the data across the Web. Cases of contradictory answers or flawed inferences may be caused by distributed pieces of knowledge that may be mutually conflicting.

The prediction of the truth value of an assertion can be cast as a *classification* problem to be solved through *statistical learning*: individual resources in an ontology can be regarded as statistical units, and their properties can be statistically inferred even when they cannot be deduced from the KB. Several approaches have been proposed in the SW literature (see [20] for a recent survey). A major issue with the methods proposed so far is that the induced statistical models (as those produced by kernel methods, tensor factorization, etc.) are either difficult to interpret by experts and to integrate in logic-based SW infrastructures, or computationally impractical.

1.1 Related Work

A variety of methods have been proposed for predicting the truth value of assertions in Web ontologies, including generative models [18,21], kernel methods [4,8,16], upgrading of propositional algorithms [15], matrix and tensor factorization methods [9,17,26]. An issue with existing methods is that they either rely on a possibly expensive search process, or induce statistical models that are often not easy to interpret by human experts. Kernel methods induce models (such as separating hyperplanes) in a high-dimensional feature space implicitly defined by a kernel function. The underlying kernel function itself usually relies on purely syntactic features of the neighborhood graphs of two individual resources (such as their common subtrees [16]). In both cases, there is not necessarily a direct translation in terms of domain knowledge. Latent variable and matrix or tensor factorization methods such as [9, 17, 21, 26] try to explain the observations in terms of latent classes or attributes, which also may be non-trivial to describe using the domain's vocabulary. The approaches in [15, 18] try to overcome this limitation by making use of more complex features and knowledge representation formalisms jointly with the ontology's terminology: these methods involve either a search process in a possibly very large feature space or complex probabilistic inference tasks, which might not be feasible in practice.

1.2 Contribution

We propose a transductive inference method for predicting the truth value of assertions, which is based on the following intuition: examples (each represented by a individual in the ontology) that are *similar* in some aspects tend to be linked by specific relations. Yet it may be not straightforward to find such relations for a given learning task. Our approach aims at identifying such relations, and permits the efficient propagation of information along chains of related entities. It turns out to be especially useful with real-world *shallow* ontologies [22] (i.e. those with a relatively simple fixed terminology and populated by very large amounts of instance data such as citation or social networks), in which the characteristics of related entities tend to be correlated. These features are particularly relevant in the context of the *Linked Open Data* [10] (LOD). Unlike other approaches, the proposed method can be used to elicit which relations link examples with similar characteristics. The proposed method is efficient, since the complexity of both inference and learning grows polynomially with the number of statistical units.

As in graph-based semi-supervised learning (SSL) methods [5], we rely on a similarity graph among examples for propagating knowledge. However, SSL methods are often designed for propositional representations; our method addresses the problem of learning from real ontologies, where examples can be interlinked by heterogeneous relations. In particular, this paper makes the following contributions:

- A method to efficiently *propagating* knowledge among similar examples: it leverages a similarity graph, which plays a critical role in the propagation process.

- An approach to efficiently *learning* the similarity matrix, by leveraging a set of semantically heterogeneous relations among examples in the ontology.

To the best of our knowledge, our approach is the first to explicitly identify relations that semantically encode similarities among examples w.r.t. a given learning task.

The remainder of the paper is organized as follows. In the next section, we review the basics of semantic knowledge representation and reasoning tasks, and we introduce the concept of *transductive learning* in the context of semantic KBs. In Sect. 3, we illustrate the proposed method based on an efficient propagation of information among related entities, and address the problem of identifying the relations relevant to the learning task. In Sect. 4, we provide empirical evidence for the effectiveness of the proposed method. Finally, in Sect. 5, we summarize the proposed approach, outline its limitations and discuss possible future research directions.

2 Transductive Learning with Web Ontologies

We consider ontological KBs using *Description Logics* (DLs) as a language to describe objects and their relations [1] Basics elements are *atomic concepts* $N_C = \{C, D, \ldots\}$ interpreted as subsets of a domain of objects (e.g. `Person` or `Article`), and *atomic roles* $N_R = \{R, S, \ldots\}$ interpreted as binary relations on such a domain (e.g. `friendOf` or `authorOf`). Domain objects are represented by *individuals* $N_I = \{a, b, \ldots\}$: each represents a domain entity (such as a person in a social network, or an article in a citation network).

In RDF(S)/OWL [1], concepts, roles and individuals are referred to as *classes*, *properties* and *resources*, respectively, and are identified by their URIs. Complex concept descriptions can be built using atomic concepts and roles by means of specific constructors offered by expressive DLs.

A *Knowledge Base* (KB) $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ contains a *TBox* $\mathcal{T}$, made by terminological axioms, and an *ABox* $\mathcal{A}$, made by ground axioms, named *assertions*, involving individuals. $\mathsf{Ind}(\mathcal{A})$ denotes the set of individuals occurring in $\mathcal{A}$. As inference procedure, *Instance Checking* consists in deciding whether $\mathcal{K} \models Q(a)$ (where Q is a query concept and a is an individual) holds. Because of the *Open-World Assumption* (OWA), instance checking may provide three possible outcomes, i.e. i) $\mathcal{K} \models Q(a)$, ii) $\mathcal{K} \models \neg Q(a)$ and iii) $\mathcal{K} \not\models Q(a) \land \mathcal{K} \not\models \neg Q(a)$. This means that failing to deductively infer the membership of an individual a to a concept Q does not imply that a is a member of its complement $\neg Q$.

Given the inherent incompleteness of deductive inference under open-world reasoning, in this work we focus on *transductive inference* [27]: this consists in reasoning from observed (training) cases to a specific set of test cases, without necessarily generalizing to unseen cases. This differs from *inductive inference*, where training cases are used to infer a general model, which is then applied to test cases.

The main motivation behind the choice of transductive learning is described by the *main principle* in [27]: “If you possess a restricted amount of information for solving some problem, try to solve the problem directly and never solve a more general problem

[1] OWL 2 W3C Recommendation: `http://www.w3.org/TR/owl-overview/`

as an intermediate step. It is possible that the available information is sufficient for a direct solution but is insufficient for solving a more general intermediate problem".

On the ground of the available information, the proposed approach aims at learning a *labeling function* for a given target class that can be used for predicting whether individuals belong to a class C (positive class) or to its complement $\neg C$ (negative class) when this cannot be inferred deductively. The problem can be defined as follows:

Definition 2.1 (Transductive Class-Membership Learning). Given:

- *a* target *class C in a KB $\mathcal{K}$;*
- *a set of examples $X \subseteq \mathsf{Ind}(\mathcal{A})$ partitioned into:*
  - *a set of* positive examples*: $X_+ \triangleq \{a \in X \mid \mathcal{K} \models C(a)\}$;*
  - *a set of* negative examples*: $X_- \triangleq \{a \in X \mid \mathcal{K} \models \neg C(a)\}$;*
  - *a set of* neutral *(unlabeled)* examples*: $X_0 \triangleq \{a \in X \mid a \notin X_+ \wedge a \notin X_-\}$;*
- *a space of* labeling functions *$\mathcal{F}$ with domain X and range $\{-1, +1\}$, i.e.*

$$\mathcal{F} \triangleq \{\mathbf{f} \mid \mathbf{f} : X \to \{+1, -1\}\};$$

- *a* cost function $\mathrm{cost}(\cdot) : \mathcal{F} \mapsto \mathbb{R}$*, specifying the cost associated to each labeling functions $\mathbf{f} \in \mathcal{F}$;*

Find $\mathbf{f}^* \in \mathcal{F}$ *minimizing* $\mathrm{cost}(\cdot)$ *w.r.t. X:*

$$\mathbf{f}^* \leftarrow \arg\min_{\mathbf{f} \in \mathcal{F}} \mathit{cost}(\mathbf{f}).$$

The transductive learning task is cast as the problem of finding a *labeling function* $\mathbf{f}^*$ for a target class C, defined over a finite set of labeled and unlabeled examples X (represented by a subset of the individuals in the KB), and minimizing some arbitrary cost criterion.

Example 2.1 (Transductive Class-Membership Learning). Consider an ontology modeling an academic domain. The problem of learning whether a set of persons is affiliated to a given research group or not, provided a set of positive and negative examples of affiliates, can be cast as a *transductive class-membership learning* problem: examples (a subset of the individuals in the ontology, each representing a person), represented by the set X, can be either *positive*, *negative* or *neutral* depending on their membership to a target class `ResearchGroupAffiliate`. Examples can be either *unlabeled* (if their membership to the target class cannot be determined) or *labeled* (if positive or negative). The transductive learning problem reduces to finding the best labeling function $\mathbf{f}$ (according to a given criterion, represented by the cost function) providing membership values for examples in X.

In this work, we exploit the relations holding among examples to *propagate* knowledge (in the form of label information) through chains of related examples. Inspired by graph-based semi-supervised transductive learning, the criterion on which the cost function will be defined follows the *semi-supervised smoothness assumption* [5]: if two points in a high-density region are close, then so should be the corresponding outputs. Transductive and semi-supervised learning are closely related: in both settings, test examples are available during the learning task (in the form of unlabeled examples). In the proposed method, the learning task is reduced to finding a labeling function $\mathbf{f}$ which *varies smoothly* across the similarity graph defined over examples.

3 Knowledge Propagation

In this section, we present our method for solving the learning problem in Def. 2.1 in the context of Web ontologies. In Sect. 3.1 we show that a similarity graph between examples can be used to efficiently propagate label information among similar examples. The effectiveness of this approach strongly depends on the choice of a similarity graph (represented by its adjacency matrix $\mathbf{W}$). In Sect. 3.2, we show how the matrix $\mathbf{W}$ can be learned from examples, by leveraging their relationship within the ontology.

3.1 Transductive Inference as an Optimization Problem

We now propose a solution to the transductive learning problem in Def. 2.1. As discussed in the end of Sect. 2, we need a labeling function $\mathbf{f}^*$ defined over examples X, which is both consistent with the training labels, and *varies smoothly* among similar examples (according to the semi-supervised smoothness assumption). In the following, we assume that a similarity graph over examples in X is already provided. Such a graph is represented by its adjacency matrix $\mathbf{W}$, such that $\mathbf{W}_{ij} = \mathbf{W}_{ji} \geq 0$ if $x_i, x_j \in X$ are *similar*, and 0 otherwise (for simplicity we assume that $\mathbf{W}_{ii} = 0$). In Sect. 3.2 we propose a solution to the problem of learning $\mathbf{W}$ from examples.

Formally, each labeling function $\mathbf{f}$ can be represented by a finite-size vector, where $\mathbf{f}_i \in \{-1, +1\}$ is the label for the i-th element in the set of examples X. According to [30], labels can be enforced to vary smoothly among similar examples by considering a cost function with the following form:

$$E(\mathbf{f}) \triangleq \frac{1}{2} \sum_{i=1}^{|X|} \sum_{j=1}^{|X|} \mathbf{W}_{ij} (\mathbf{f}_i - \mathbf{f}_j)^2 + \epsilon \sum_{i=1}^{|X|} \mathbf{f}_i^2, \tag{1}$$

where the first term enforces the labeling function to vary smoothly among similar examples (i.e. those connected by an edge in the similarity graph), and the second term is a L_2 regularizer (with weight $\epsilon > 0$) over $\mathbf{f}$. A labeling for unlabeled examples X_0 is obtained by minimizing the function $E(\cdot)$ in Eq. 1, constraining the value of $\mathbf{f}_i$ to 1 (resp. -1) if $x_i \in X_+$ (resp. $x_i \in X_-$).

Let $L \triangleq X_+ \cup X_-$ and $U \triangleq X_0$ represent labeled and unlabeled examples respectively. Constraining $\mathbf{f}$ to discrete values, i.e. $\mathbf{f}_i \in \{-1, +1\}, \forall x_i \in X_0$, has two main drawbacks:

- The function $\mathbf{f}$ only provides a *hard* classification (i.e. $\mathbf{f}_U \in \{-1, +1\}^{|U|}$), any measure of confidence;
- E defines the energy function of a discrete Markov Random Field, and calculating the marginal distribution over labels $\mathbf{f}_U$ is inherently difficult [13].

To overcome these problems, in [30] authors propose a continuous relaxation of $\mathbf{f}_U$, where labels for unlabeled examples are represented by real values, $\mathbf{f}_U \in \mathbb{R}^{|U|}$, which also express a measure of the classification confidence. This allows for a simple, closed-form solution to the problem of minimizing E for a fixed $\mathbf{f}_L$, where $\mathbf{f}_L$ represents the labels of labeled examples.

Application to Class-Membership Learning We can solve the learning problem in Def. 2.1 by minimizing the cost function $E(\cdot)$ in Eq. 1. It can be rewritten as [30]:

$$E(\mathbf{f}) = \mathbf{f}^T(\mathbf{D} - \mathbf{W})\mathbf{f} + \epsilon \mathbf{f}^T = \mathbf{f}^T(\mathbf{L} + \epsilon \mathbf{I})\mathbf{f}, \tag{2}$$

where $\mathbf{D}$ is a diagonal matrix such that $\mathbf{D}_{ii} = \sum_{j=1}^{|X|} \mathbf{W}_{ij}$ and $\mathbf{L} \triangleq \mathbf{D} - \mathbf{W}$ is the *graph Laplacian* of $\mathbf{W}$. Reordering the matrix $\mathbf{W}$, the graph Laplacian $\mathbf{L}$ and the vector $\mathbf{f}$ w.r.t. their membership to L and U, they can be rewritten as:

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}_{LL} & \mathbf{W}_{LU} \\ \mathbf{W}_{UL} & \mathbf{W}_{UU} \end{bmatrix}, \quad \mathbf{L} = \begin{bmatrix} \mathbf{L}_{LL} & \mathbf{L}_{LU} \\ \mathbf{L}_{UL} & \mathbf{L}_{UU} \end{bmatrix}, \quad \mathbf{f} = \begin{bmatrix} \mathbf{f}_L \\ \mathbf{f}_U \end{bmatrix}. \tag{3}$$

The problem of finding the $\mathbf{f}_U$ minimizing the cost function E for a fixed value for $\mathbf{f}_L$ has a closed form solution:

$$\mathbf{f}_U^* = (\mathbf{L}_{UU} + \epsilon \mathbf{I})^{-1} \mathbf{W}_{UL} \mathbf{f}_L. \tag{4}$$

Complexity A solution for Eq. 4 can be computed efficiently in nearly-linear time w.r.t. $|X|$. Indeed computing $\mathbf{f}_U^*$ can be reduced to solving a linear system in the form $\mathbf{A}\mathbf{x} = \mathbf{b}$, with $\mathbf{A} = (\mathbf{L}_{UU} + \epsilon \mathbf{I})$, $\mathbf{b} = \mathbf{W}_{UL}\mathbf{f}_L$ and $\mathbf{x} = \mathbf{f}_U^*$. A linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$ with $\mathbf{A} \in \mathbb{R}^{n \times n}$ can be solved in nearly linear time w.r.t. n if the coefficient matrix $\mathbf{A}$ is *symmetric diagonally dominant*[2] (SDD), e.g. using the algorithm in [6], whose time-complexity is $\approx \mathrm{O}\big(m \log^{1/2} n\big)$, where m is the number of non-zero entries in $\mathbf{A}$ and n is the number of variables in the system of linear equations. In Eq. 4, the matrix $(\mathbf{L}_{UU} + \epsilon \mathbf{I})$ is SDD (since $\mathbf{L}_{UU}$ is a principal submatrix of $\mathbf{L}$, which is SDD [25]). An efficient parallel solver for SDD linear systems is discussed in [19].

3.2 Learning to Propagate Knowledge in Web Ontologies

As discussed in Sect. 3.1, the proposed approach to knowledge propagation relies on a similarity graph, represented by its adjacency matrix $\mathbf{W}$.

The underlying assumption of this work is that some relations among examples in the KB might encode a similarity relation w.r.t. a specific target property or class. Identifying such relations can help propagate information through similar examples.

In the literature, this effect goes under the name of *Guilt-by-Association* [14]: related examples influence each other, and some relations (e.g. *friendship* in a social network) can encode some form of similarity w.r.t. a set of properties (such as political views, hobbies, religious beliefs). However, depending on the learning task at hand, not all relations are equally effective at encoding similarity relations. For example in a social network, friends may tend to share common interests, while quiet people may tend to prefer talkative friends and vice-versa [14].

In this work, we represent each relation by means of an *adjacency matrix* $\tilde{\mathbf{W}}$, such that $\tilde{\mathbf{W}}_{ij} = \tilde{\mathbf{W}}_{ji} = 1$ iff the relation $\texttt{rel}(x_i, x_j)$ between x_i and x_j holds in the ontology; `wrel` might represent any generic relation between examples (e.g. friendship or co-authorship). For simplicity, we assume that $\tilde{\mathbf{W}}_{ii} = 0, \forall i$.

[2] A matrix $\mathbf{A}$ is SDD iff $\mathbf{A}$ is symmetric (i.e. $\mathbf{A} = \mathbf{A}^T$) and $\forall i : \mathbf{A}_{ii} \geq \sum_{i \neq j} |\mathbf{A}_{ij}|$.

Given a set of adjacency matrices $\mathcal{W} \triangleq \{\tilde{\mathbf{W}}_1, \ldots, \tilde{\mathbf{W}}_r\}$ (one for each relation type), we can define $\mathbf{W}$ as a linear combination of the matrices in $\mathcal{W}$:

$$\mathbf{W} \triangleq \sum_{i=1}^{r} \mu_i \tilde{\mathbf{W}}_i, \qquad \text{with } \mu_i \geq 0, \forall i \tag{5}$$

where μ_i, is a parameter representing the contribution of $\tilde{\mathbf{W}}_i$ to the adjacency matrix of the similarity graph $\mathbf{W}$. Non-negativity in $\boldsymbol{\mu}$ ensures that $\mathbf{W}$ has non-negative weights, and therefore the corresponding graph Laplacian $\mathbf{L}$ is positive semidefinite [25] (PSD), leading to the unique, closed form solution in Eq. 4.

Probabilistic Interpretation as a Gaussian Random Field Let us consider the relaxation of the energy function in Eq. 2, such that labels $\mathbf{f}$ are allowed to range in $\mathbb{R}^{|X|}$ ($\mathbf{f} \in \mathbb{R}^{|X|}$). It corresponds to the following probability density function over $\mathbf{f}$:

$$p(\mathbf{f}) = (2\pi)^{-\frac{1}{2}} |\boldsymbol{\Sigma}|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2} E(\mathbf{f})\right\} = \mathcal{N}\Big(\mathbf{0}, (\mathbf{L} + \epsilon \mathbf{I})^{-1}\Big). \tag{6}$$

The probability density function in Eq. 6 defines a Gaussian Markov Random Field [13] (GMRF) $\mathbf{f} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma} = \boldsymbol{\Omega}^{-1}$ and $\boldsymbol{\Omega} = (\mathbf{L} + \epsilon \mathbf{I})$ are respectively its *covariance* and *inverse covariance* (or *precision*) matrix, and $|\boldsymbol{\Sigma}|$ indicates the determinant of the covariance matrix $\boldsymbol{\Sigma}$.

The covariance matrix and its inverse fully determine the independence relations among variables in a GMRF [13]: if $\boldsymbol{\Omega}_{ij} \neq 0$, then there is an edge between $\mathbf{f}_i$ and $\mathbf{f}_j$ in the minimal I-map GMRF of p. A zero element in the inverse covariance matrix implies that two variables are conditionally independent given all the other variables.

Parameters Learning The parametric form of $\mathbf{W}$ is fully specified by the parameters $\boldsymbol{\mu}$ in Eq. 5, which may be unknown. We will estimate the parameters by means of *Leave-One-Out* (LOO) *Error minimization*: given that propagation can be performed efficiently, we are able of directly minimizing the LOO error, consisting in the summation of reconstruction errors obtained by considering each labeled example, in turn, as unlabeled, and predicting its label (as in [29]). This leads to a computationally efficient procedure for evaluating the matrix $\mathbf{W}$, and yields more flexibility as arbitrary loss functions are allowed. Let $U_i \triangleq U \cup \{x_i\}$ and $L_i \triangleq L - \{x_i\}$: the labeling vector $\mathbf{f}$ and matrices $\mathbf{W}$ and $\mathbf{L}$, for any given $x_i \in L$, can be rewritten as follows:

$$\mathbf{f} = \begin{bmatrix} \mathbf{f}_{L_i} \\ \mathbf{f}_{U_i} \end{bmatrix}, \quad \mathbf{W} = \begin{bmatrix} \mathbf{W}_{L_i L_i} & \mathbf{W}_{L_i U_i} \\ \mathbf{W}_{U_i L_i} & \mathbf{W}_{U_i U_i} \end{bmatrix}, \quad \mathbf{L} = \begin{bmatrix} \mathbf{L}_{L_i L_i} & \mathbf{L}_{L_i U_i} \\ \mathbf{L}_{U_i L_i} & \mathbf{L}_{U_i U_i} \end{bmatrix}, \tag{7}$$

where w.l.o.g. we assume that the left-out example $x_i \in L$ corresponds to the first element in U_i (in the enumeration used for the block representation in Eq. 7). Let $\ell(x, \hat{x})$ be a generic, differentiable loss function (e.g. $\ell(x, \hat{x}) = |x - \hat{x}|$ for the absolute loss, or $\ell(x, \hat{x}) = (x - \hat{x})^2/2$ for the quadratic loss). The LOO Error is defined as follows:

$$\mathcal{Q}(\boldsymbol{\Theta}) \triangleq \sum_{i=1}^{|L|} \ell(\mathbf{f}_i, \hat{\mathbf{f}}_i), \tag{8}$$

where $\mathbf{e}^T \triangleq (1, 0, \dots, 0) \in \mathbb{R}^{u+1}$ and $\hat{\mathbf{f}}_i \triangleq \mathbf{e}^T(\mathbf{L}_{U_iU_i} + \epsilon\mathbf{I})^{-1}\mathbf{W}_{U_iL_i}\mathbf{f}_{L_i}$ represents the continuous label value assigned to x_i as if such a value was not known in advance. The vector $\mathbf{e}^T$ is needed to select the first value of $\mathbf{f}^*_{U_i}$ only, i.e. the inferred continuous label associated to the left-out example $x_i \in L$. This leads to the definition of the following criterion for learning the optimal set of parameters $\boldsymbol{\Theta} \triangleq \{\boldsymbol{\mu}, \epsilon\}$:

Definition 3.1 (Minimum LOO Error Parameters). *Given a set of labeled (resp. unlabeled) examples L (resp. U) and a similarity matrix $\mathbf{W}$ defined by parameters $\boldsymbol{\Theta}$ (according to the parametric form of $\mathbf{W}$ in Eq. 5), the* minimum LOO Error Parameters *$\boldsymbol{\Theta}^*_{LOO}$ are defined as follows:*

$$\boldsymbol{\Theta}^*_{LOO} \triangleq \arg\min_{\boldsymbol{\Theta}} \mathcal{Q}(\boldsymbol{\Theta}) + \lambda||\boldsymbol{\Theta}||^2, \tag{9}$$

where the function $\mathcal{Q}$ is defined as in Eq. 8 and $\lambda > 0$ is a small positive scalar that weights a regularization term over $\boldsymbol{\Theta}$ (useful for avoiding some parameters to diverge).

The objective function in Def. 3.1 is differentiable and can be efficiently minimized by using gradient-based function minimization approaches such as L-BFGS.

Let $\mathbf{Z}_i = (\mathbf{L}_{U_iU_i} + \epsilon\mathbf{I})$. The gradient of $\mathcal{Q}$ w.r.t. a parameter $\theta \in \boldsymbol{\Theta}$ is given by:

$$\frac{\partial \mathcal{Q}(\boldsymbol{\Theta})}{\partial \theta} = \sum_{i=1}^{|L|} \frac{\partial \ell(\mathbf{f}_i, \hat{\mathbf{f}}_i)}{\partial \hat{\mathbf{f}}_i} \left[\mathbf{e}^T \mathbf{Z}_i^{-1} \left(\frac{\partial \mathbf{W}_{U_iL_i}}{\partial \theta} \mathbf{f}_{L_i} - \frac{\partial \mathbf{Z}_i}{\partial \theta} \mathbf{f}^*_{U_i} \right) \right]. \tag{10}$$

Complexity of the Gradient Calculation Let $\mathbf{z}_i = \left(\frac{\partial \mathbf{W}_{U_iL_i}}{\partial \theta} \mathbf{f}_{L_i} - \frac{\partial \mathbf{Z}_i}{\partial \theta} \mathbf{f}^*_{U_i} \right)$. Calculating $\mathbf{Z}_i^{-1}\mathbf{z}_i$ can be reduced to solving a linear system in the form $\mathbf{A}\mathbf{x} = \mathbf{b}$, with $\mathbf{A} = \mathbf{Z}_i = (\mathbf{L}_{U_iU_i} + \epsilon\mathbf{I})$ and $\mathbf{b} = \mathbf{z}_i$. As discussed in Sect. 3.1, this calculation has a nearly-linear complexity in the number of non-zero elements in $\mathbf{A}$, since $\mathbf{Z}_i$ is SDD.

4 Empirical Evaluation

The transductive inference method discussed in Sect. 3, which we will refer to as *Adaptive Knowledge Propagation* (AKP), was experimentally evaluated [3] in comparison with other approaches proposed in the literature on a variety of assertion prediction problems. In the following, we describe the setup of experiments and their outcomes.

4.1 Setup

In empirical evaluations, we used an open source DL reasoner [4]. In experiments, we considered the DBPEDIA 3.9 Ontology [3]. DBPEDIA [3] makes available structured information extracted from Wikipedia the LOD cloud providing unique identifiers for the described entities that can be dereferenced over the Web. DBPEDIA 3.9, released in September 2013, describes 4.0 million entities.

[3] Sources and datasets are available at `http://lacam.di.uniba.it/phd/pmm.html`

[4] Pellet v2.3.1 – `http://clarkparsia.com/pellet/`

Experimental Setting As discussed in Sect. 3.2, parameters $\boldsymbol{\Theta} = \{\boldsymbol{\mu}, \epsilon\}$ in AKP are estimated by minimizing the Leave-One-Out error $\mathcal{Q}$, as described in Eq. 9. We solved the problem by using *Projected Gradient Descent*, according to the gradient formulation in Eq. 10 (enforcing $\boldsymbol{\mu} \geq \mathbf{0}$ and $\epsilon > 0$), together with an intermediate line search to assess the step size. The regularization parameter λ in Eq. 9 was fixed to $\lambda = 10^{-8}$. In this work, each of the adjacency matrices $\mathcal{W} = \{\tilde{\mathbf{W}}_1, \ldots, \tilde{\mathbf{W}}_r\}$ is associated to a distinct *atomic role* in the ontology, linking at least two examples.

Before each experiment, all knowledge inherent to the target class was removed from the ontology. Following the evaluation procedures in [16,28], members of the target concepts were considered as *positive examples*, while an equal number of *negative examples* was randomly sampled from unlabeled examples. Remaining instances (i.e. neither positive nor negative) were considered as *neutral examples*.

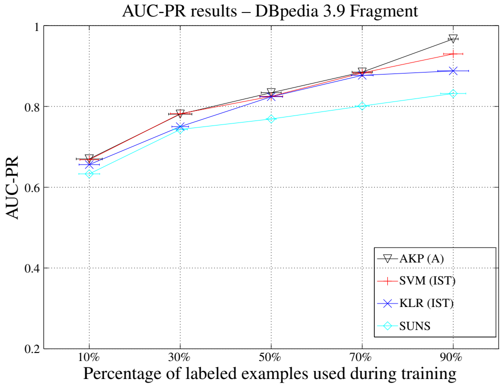
Results are reported in terms of *Area Under the Precision-Recall Curve* (AUC-PR), a measure to evaluate rankings also used in e.g. [17], and calculated using the procedure described in [7]. In each experiment, we considered the problem of predicting the membership to each of several classes; for each of such classes, we performed a 10-fold cross validation (CV), and report the average AUC-PR obtained using each of the considered methods. Since the folds used to evaluate each of the methods do not vary, we report statistical significance tests using a paired, non-parametric difference test (Wilcoxon T test). We also report diagrams showing how using a limited quantity of randomly sampled labeled training instances (i.e. $10\%, 30\%, 50\%, \ldots$, a plausible scenario for a number of real world settings with limited labeled training data), and using the remaining examples for testing, affects the results in terms of AUC-PR.

Setup of the Compared Methods We compared our method with state-of-the-art approaches proposed for learning from ontological KBs. Specifically, we selected two kernel methods: Soft-Margin SVM [23, pg. 223] (SM-SVM) and Kernel Logistic Regression (KLR), jointly with the *Intersection SubTree* [16] (IST) kernel for ontological KBs, and the SUNS [26] relational prediction model. The relational graph used by both the RDF kernel and SUNS was materialized as follows: all $\langle \mathtt{s}, \mathtt{p}, \mathtt{o} \rangle$ triples were retrieved by means of SPARQL-DL queries (where $\mathtt{p}$ was either an object or a data-type property) together with all *direct type* and *direct sub-class* relations.

As in [16], IST kernel parameters were ranging in $d \in \{1, 2, 3, 4\}$ and $\lambda_{ist} \in \{0.1, 0.3, \ldots, 0.9\}$). In order to obtain a ranking among instances (provided by soft-labels $\mathbf{f}$ in AKP), we applied the logistic function s to the decision boundary f instead of the standard sign function, commonly used in the classification context (thus obtaining $s(f(\cdot)) : \mathcal{X} \to [0, 1]$). In SM-SVM, $C \in \{0.0, 10^{-6}, 10^{-4}, \ldots, 10^{4}, 10^{6}\}$, while in KLR the weight λ_k associated to the L_2 regularizer was found considering $\lambda_k \in \{10^{-4}, 10^{-3}, \ldots, 10^{4}\}$. In SUNS, parameters were selected by means of a 10-fold CV within the training set by grid optimization, with $t \in \{2, 4, 6, \ldots, 24\}$ and $\lambda_s \in \{0, 10^{-2}, 10^{-1}, \ldots, 10^{6}\}$.

4.2 Results

Similarly to [17], we evaluated the proposed approach on two prediction tasks, namely predicting party affiliations to either the Democratic and the Republican party for US



Method	AUC-PR (mean ± var.)	(S)
AKP (A)	0.967 ± 0.003	
SM-SVM (IST)	0.930 ± 0.011	▽
KLR (IST)	0.888 ± 0.029	▼
SUNS	0.832 ± 0.019	▼

Fig. 1: DBPEDIA 3.9 Ontology – Left: AUC-PR results (mean, st.d.) estimated by 10-fold CV, obtained varying the percentage of examples used for training – Right: AUC-PR results estimated by 10-fold CV: $\blacktriangledown/\triangledown$ (resp. $\blacktriangle/\triangle$) indicates that AKP's mean is significantly higher (resp. lower) in a paired Wilcoxon T test with $p < 0.05$ / $p < 0.10$

presidents and vice-presidents. The experiment illustrated in [17] uses a small RDF fragment containing the `president` and `vicePresident` predicates only. In this experiment, we used a real-life fragment of DBPEDIA 3.9 (obtained by means of a crawling process), containing a number of irrelevant and possibly noisy entities and relations. Following the procedure in [11], the DBPEDIA 3.9 RDF graph was traversed starting from resources representing US presidents and vice-presidents: all immediate neighbors, i.e. those with a recursion depth of 1, were retrieved, together with their related schema information (direct classes and their super-classes, together with their hierarchy). All extracted knowledge was used to create an $\mathcal{ALCH}$ ontology fragment, with 78795 axioms, 16606 individuals, 132 properties and 11 classes.

In this experiment, 82 individuals representing US presidents and vice-presidents were interlinked by 25 relations represented by atomic roles. The proposed method, denoted as AKP (A), makes use of such atomic roles to identify relations holding among the examples in the ontology.

Experimental results are summarized in Fig. 1. We observe that AUC-PR values obtained with AKP (A) are significantly higher than results obtained by other methods considered in comparison ($p < 0.05$, except for three cases in which $p < 0.10$). Results show how presidents and vice-presidents linked by simple relations such as `president` and `vicePresident` tend to be affiliated to the same political party.

AKP (A) is able to identify which atomic roles are likely to link same party affiliates. As expected, it recognizes that relations represented by the `president` and `vicePresident` atomic roles should be associated to higher weights, which means that presidents and their vice-presidents tend to have similar political party affiliations. AKP (A) also recognizes that presidents (or vice-presidents) linked by the `successor` atomic role are unlikely to have similar political party affiliations.

5 Conclusions and Future Works

In this work, we proposed a semi-supervised transductive inference method for statistical learning in the context of the WEB OF DATA. Starting from the assumption that some relations among entities in a Web ontology can encode similarity information w.r.t. a given prediction task (pertaining a particular property of examples, such as a class-membership relation), we proposed a method (named *Adaptive Knowledge Propagation*, or AKP) for efficiently learning the best way to propagate knowledge among related examples (each represented by an individual) in a Web ontology.

We empirically show that the proposed method is able to identify which relations encode similarity w.r.t. a given property, and that their identification can provide an effective method for predicting unknown characteristics of individuals. We also show that the proposed method can provide competitive results, in terms of AUC-PR, in comparison with other state-of-the-art methods in literature.

We only considered relations between statistical units (i.e. training examples) encoded by atomic roles. However, those do not always suffice: for example, in the research group affiliation prediction task discussed in [16], individuals representing researchers in the AIFB PORTAL ontology are not related by any atomic role. We are currently investigating other approaches to identifying meaningful relations among individuals, for example by means of Conjunctive Query Answering [12]. Other research directions involve the study of different objective functions and optimization methods.

Acknowledgments This work fulfills the objectives of the PON 02_00563_3489339 project "PUGLIA@SERVICE - Internet-based Service Engineering enabling Smart Territory structural development" funded by the Italian Ministry of University and Research (MIUR).

References

1. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F. (eds.): The Description Logic Handbook. Cambridge University Press (2007)
2. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. Scientific American 284(5), 34–43 (May 2001)
3. Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S.: DBpedia - a crystallization point for the Web of Data. J. Web Sem. 7(3), 154–165 (2009)
4. Bloehdorn, S., Sure, Y.: Kernel methods for mining instance data in ontologies. In: Aberer, K., et al. (eds.) Proceedings of ISWC'07. LNCS, vol. 4825, pp. 58–71. Springer (2007)
5. Chapelle, O., Schölkopf, B., Zien, A. (eds.): Semi-Supervised Learning. MIT Press (2006)
6. Cohen, M.B., Kyng, R., Miller, G.L., Pachocki, J.W., Peng, R., Rao, A., Xu, S.C.: Solving SDD linear systems in nearly $m\log^{1/2}n$ time. In: Shmoys [24], pp. 343–352
7. Davis, J., Goadrich, M.: The relationship between Precision-Recall and ROC curves. In: Cohen, W., et al. (eds.) Proceedings of ICML'06. pp. 233–240. ACM (2006)
8. Fanizzi, N., d'Amato, C., Esposito, F.: Induction of robust classifiers for web ontologies through kernel machines. J. Web Sem. 11, 1–13 (2012)
9. Franz, T., Schultz, A., Sizov, S., Staab, S.: TripleRank: Ranking Semantic Web Data by Tensor Decomposition. In: Bernstein, A., et al. (eds.) International Semantic Web Conference. LNCS, vol. 5823, pp. 213–228. Springer (2009)

10. Heath, T., Bizer, C.: Linked Data: Evolving the Web into a Global Data Space. Synthesis Lectures on the Semantic Web, Morgan & Claypool Publishers (2011)
11. Hellmann, S., Lehmann, J., Auer, S.: Learning of OWL Class Descriptions on Very Large Knowledge Bases. Int. J. Semantic Web Inf. Syst. 5(2), 25–48 (2009)
12. Hitzler, P., Krötzsch, M., Rudolph, S.: Foundations of Semantic Web Technologies. Chapman & Hall/CRC (2009)
13. Koller, D., Friedman, N.: Probabilistic Graphical Models: Principles and Techniques. MIT Press (2009)
14. Koutra, D., Ke, T.Y., Kang, U., Chau, D.H., Pao, H.K.K., Faloutsos, C.: Unifying Guilt-by-Association Approaches: Theorems and Fast Algorithms. In: Gunopulos, D., et al. (eds.) Proceedings of ECML/PKDD'11. LNCS, vol. 6912, pp. 245–260. Springer (2011)
15. Lin, H.T., Koul, N., Honavar, V.: Learning Relational Bayesian Classifiers from RDF Data. In: Aroyo, L., et al. (eds.) International Semantic Web Conference (1). LNCS, vol. 7031, pp. 389–404. Springer (2011)
16. Lösch, U., Bloehdorn, S., Rettinger, A.: Graph kernels for RDF data. In: Simperl, E., et al. (eds.) Proceedings of ESWC'12. LNCS, vol. 7295, pp. 134–148. Springer (2012)
17. Nickel, M., Tresp, V., Kriegel, H.P.: A Three-Way Model for Collective Learning on Multi-Relational Data. In: Getoor, L., et al. (eds.) Proceedings of ICML'11. pp. 809–816. Omnipress (2011)
18. Ochoa-Luna, J.E., Cozman, F.G.: An Algorithm for Learning with Probabilistic Description Logics. In: Bobillo, F., et al. (eds.) Proceedings of the 5th International Workshop on Uncertainty Reasoning for the Semantic Web, URSW09. CEUR Workshop Proceedings, vol. 654, pp. 63–74. CEUR-WS.org (2009)
19. Peng, R., Spielman, D.A.: An efficient parallel solver for SDD linear systems. In: Shmoys [24], pp. 333–342
20. Rettinger, A., Lösch, U., Tresp, V., d'Amato, C., Fanizzi, N.: Mining the Semantic Web: Statistical learning for next generation knowledge bases. Data Min. Knowl. Discov. 24(3), 613–662 (2012)
21. Rettinger, A., Nickles, M., Tresp, V.: Statistical Relational Learning with Formal Ontologies. In: Buntine, W.L., et al. (eds.) Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases, ECML/PKDD'09. LNCS, vol. 5782, pp. 286–301. Springer (2009)
22. Shadbolt, N., Berners-Lee, T., Hall, W.: The Semantic Web Revisited. IEEE Intelligent Systems 21(3), 96–101 (2006)
23. Shawe-Taylor, J., Cristianini, N.: Kernel Methods for Pattern Analysis. Cambridge University Press (2004)
24. Shmoys, D.B. (ed.): Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014. ACM (2014)
25. Spielman, D.A.: Algorithms, Graph Theory, and Linear Equations in Laplacian Matrices. In: Proceedings of ICM'10. pp. 2698–2722 (2010)
26. Tresp, V., Huang, Y., Bundschus, M., Rettinger, A.: Materializing and querying learned knowledge. In: Proceedings of IRMLeS'09 (2009)
27. Vapnik, V.N.: Statistical learning theory. Wiley, 1 edn. (Sep 1998)
28. de Vries, G.K.D.: A Fast Approximation of the Weisfeiler-Lehman Graph Kernel for RDF Data. In: Blockeel, H., et al. (eds.) ECML/PKDD (1). LNCS, vol. 8188, pp. 606–621. Springer (2013)
29. Zhang, X., et al.: Hyperparameter Learning for Graph Based Semi-supervised Learning Algorithms. In: Schölkopf, B., et al. (eds.) NIPS. pp. 1585–1592. MIT Press (2006)
30. Zhu, X., Ghahramani, Z., Lafferty, J.D.: Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions. In: Fawcett, T., et al. (eds.) Proceedings of ICML'03. pp. 912–919. AAAI Press (2003)