Learning Relational Bayesian Classifiers on the Semantic Web

Doina Caragea[1], **Jie Bao**[2] **and Vasant Honavar**[2]
[1] Data Mining and Bioinformatics Laboratory, Department of Computing and Information Sciences
Kansas State University, Manhattan, KS 66506, USA
[2]Artificial Intelligence Research Laboratory, Department of Computer Science
Iowa State University, Ames, Iowa 50011, USA

dcaragea@ksu.edu,{baojie,honavar}@cs.iastate.edu

Abstract

With the advent of the Semantic Web, there is an increased availability of meta data (ontologies) that make explicit the semantic commitments associated with data and an urgent need for machine learning algorithms for building predictive models from such data. Usually, there is no unique global interpretation of data from semantically disparate, autonomous sources. Furthermore, it is neither feasible nor desirable to integrate data from sources on the Semantic Web in a centralized data warehouse. In this paper, we formulate the problem of learning classifiers from a set of related, semantically heterogeneous data sources on the Semantic Web from a user's point of view. We describe a general strategy for transforming algorithms for learning classifiers from data into algorithms for learning classifiers from a set of semantically heterogeneous distributed data sources. We apply this strategy to the task of learning relational Bayesian classifiers from a collection of such data sources. The proposed approach can be generalized to other relational learning algorithms. Our results provide some of the essential elements of approaches for acquiring useful knowledge from information sources that are becoming available on the Semantic Web.

1 Introduction

Recent advances in sensors, digital storage, computing and communications technologies (world-wide web) have led to a proliferation of autonomously operated, geographically distributed data repositories in virtually every area of human endeavor including e-business and e-commerce (targeted marketing), e-science (bioinformatics, environmental informatics), e-government, and security informatics. Effective use of such data in practice (e.g., building useful predictive models of consumer behavior, discovery of factors that contribute to large climatic changes, analysis of demographic factors that contribute to global poverty, analysis of social networks, or even finding out what makes a book a bestseller) requires accessing and analyzing data from multiple sources. The Semantic Web enterprise [Berners-Lee *et al.*, 2001] is aimed at making the contents of the Web machine interpretable. Data and resources on the Web are annotated and linked by associating meta data that make *explicit*, the ontological commitments of the data source providers or in some cases, the shared ontological commitments of a small community of users. Given the autonomous nature of the data sources on the Web, and the diverse purposes for which the data are gathered, in the absence of a universal ontology, it is inevitable that there is no unique global interpretation of the data that serves the needs of all users under all scenarios. Many groups have attempted to develop, with varying degrees of success, tools for flexible integration and querying of data from semantically disparate sources [Levy, 2000; Noy, 2004; Doan and Halevy, 2005; Calvanese *et al.*, 2005], as well as techniques for discovering semantic correspondences between ontologies to assist in the process [Kalfoglou and Schorlemmer, 2005; Noy and Stuckenschmidt, 2005]. These and related advances in Semantic Web technologies present unprecedented opportunities for exploiting multiple, related data sources, each annotated with its own meta data, in discovering useful knowledge in many application domains.

While there has been significant work on applying machine learning to ontology construction, information extraction from text, and discovery of mappings between ontologies [Kushmerick *et al.*, 2005], there has been relatively little work on machine learning approaches to knowledge acquisition from data sources, each annotated with meta data that exposes the structure (schema) and semantics (in reference to a particular ontology), on the Semantic Web. However, there is a large body of literature on distributed learning (see [Kargupta and Chan, 2000] for a survey). Furthermore, recent work e.g., [Zhang *et al.*, 2005; Hotho *et al.*, 2003] has shown that use of meta data, in the form of ontologies (class hierarchies, attribute value hierarchies), in addition to data, can improve the quality (accuracy, interpretability) of the learned predictive models.

Against this background, we note that a large class of data sources on the Semantic Web can be viewed (at a certain level of abstraction) as a collection of semantically disparate relational data sources that are semantically related, from a user's point of view, in the context of a specific knowledge acquisition task. The problem of learning classifiers from a semantically homogeneous relational database has received much attention in the recent machine learning literature [Getoor *et al.*, 2001; Neville *et al.*, 2003]. In this paper, we extend such

approaches to learn classifiers from multiple semantically disparate, geographically distributed, relational data sources on the Semantic Web.

The rest of the paper is organized as follows: In Section 2, we present the problem formulation, and describe a general strategy for transforming algorithms for learning classifiers from relational data into algorithms for learning classifiers from semantically disparate, relational data sources, using ontologies and mappings between ontologies, in a setting where it is neither feasible nor desirable to integrate all the data available into a single relational data warehouse. We show that the resulting classifiers can be guaranteed (under fairly general assumptions) to be identical to those obtained from a centralized, integrated relational data warehouse constructed from a specified collection of distributed relational data sources and associated ontologies and mappings. In Section 3, we illustrate this strategy in the case of Relational Bayesian Classifiers [Neville *et al.*, 2003]. We conclude with a summary and a brief discussion of related work and some future research directions.

2 Learning Classifiers from a Set of Semantically Heterogeneous Relational Data Sources

2.1 Ontology-Extended Data Sources and User Views

We define an *ontology-extended relational data source* (OERDS) as a tuple $\mathcal{D} = \{D, S, O\}$, where D is the actual data set in the data source, S represents the data source schema and O represents the data source ontology [Bonatti *et al.*, 2003; Caragea *et al.*, 2005].

In the relational model, each data source consists of a set of *concepts* $X_1, \cdots, X_n$ and a set of *properties* of these concepts $P_1, \cdots, P_m$. Each concept has associated with it, a set of *attributes* denoted by $\mathcal{A}(X_i)$ and a set of k-ary *relations* ($k > 1$) denoted by $\mathcal{R}(X_i)$. Each attribute A_i takes values in a set $\mathcal{V}(A_i)$. The concepts and the properties of the concepts (attributes and relations) define the *schema* of a relational data source. A *data set* D is an instantiation $\mathcal{I}(S)$ of a schema S [Getoor *et al.*, 2001].

The *ontology* O of an OERDS $\mathcal{D}$ consists of two parts: *structure ontology*, O_S, that defines the semantics of the data source schema (concepts and properties of the concepts that appear in data source schema S); and *content ontology*, O_I, that defines the semantics of the content of data (values and relationships between values that the attributes can take in instantiations of schema S). *Isa* relationships induce *schema concept hierarchies* (SCHs) over subsets of concepts in a schema and *attribute value hierarchies* (AVHs) over values of attributes (AVHs can be seen as defining a *type hierarchy* over the corresponding attributes). Thus, an ontology O can be decomposed into a set of schema concept hierarchies $\{C_1, \cdots, C_r\}$ and a set of attribute value hierarchies $\{T_1, \cdots, T_l\}$, with respect to the *isa* relationship. A *cut* (or *level of abstraction*) through an SCH or AVH induces a partition of the set of leaves in that hierarchy. A *global cut* through an ontology consists of a set of cuts, one for each constituent hierarchy.

On the Semantic Web, it is unrealistic to assume the existence of a single global ontology that corresponds to a universally agreed upon set of ontological commitments for all users. Instead, it is much more realistic to allow each user or a community of users to choose the ontological commitments that they deem useful in a specific context. A *user ontology* O_U, together with a set of *interoperation constraints* IC, and the associated set of *mappings* $\{\psi_i | i = 1, p\}$ from the user ontology O_U to the data source ontologies $O_1 \cdots O_p$ define a *user view* [Caragea *et al.*, 2005]. In the relational setting considered in this paper, the interoperation constraints can be equality constraints or inclusion constraints and can be defined at the concept level (between related concepts), property level (between related attributes or relations) and at the attribute value level (between related attribute values).

2.2 Ontology-Extended Bibliographic Data Sources

We will use an example from the bibliography domain to illustrate the main notions introduced above. Consider the problem of classifying computer science research papers into categories from a topic hierarchy (e.g., Artificial Intelligence, Networking, Data Mining, Relational Data Mining, etc.) [McCallum *et al.*, 2000]. A user interested in a document classification task, might consider using several data sources, such as MIT Libraries (http://libraries.mit.edu/index.html), INRIA Reference Database (http://ontoweb.org/), etc., for learning classifiers. The Ontology Alignment Evaluation Initiative (OAEI) has made available a Test Library (http://oaei.inrialpes.fr/2005/benchmarks/) that contains representative ontologies for the data sources above. In this case, the structure ontologies define the relevant concepts, such as `Reference`, `Book`, `Article`, `Journal`, `Conference`, etc.) and properties of the concepts such as `Article` *author* `Author`; `Article` *cites* `Article`; `Author` *position*; `Article` *journal* `Journal`, etc. The properties in these ontologies include both attributes (e.g., *position*) and binary relations (e.g., *author*).

Figure 1 shows a small fragment of the schema ontology corresponding to a user view of a reference data source, using standard entity-relationship (ER) notation. Figure 2 identifies fragments of the SCHs associated with related subsets of concepts in INRIA, MIT and user schema ontologies. Figure 3 shows concept level interoperation constraints (equality $=$ and inclusion $<$) between the user SCH and the MIT and INRIA SCHs: $x = y$ means that x and y are *equivalent*, $x < y$ means that y *subsumes* x, i.e., y is *more general* than x.

Assuming that a concept `Author` has an attribute called *position*, this attribute can be described using an AVH as shown in Figure 4. The set {*faculty, research staff, engineer, student*} represents a cut Γ through this hierarchy. The set {*tenured, assistant professor, research staff, engineer, student*} is a refinement of the cut Γ.

2.3 Problem Formulation

We assume the existence of

(1) A collection of several related OERDSs $\mathcal{D}_1$=$\{D_1, S_1, O_1\}$,$\cdots$,$\mathcal{D}_p$=$\{D_p, S_p, O_p\}$ for which:

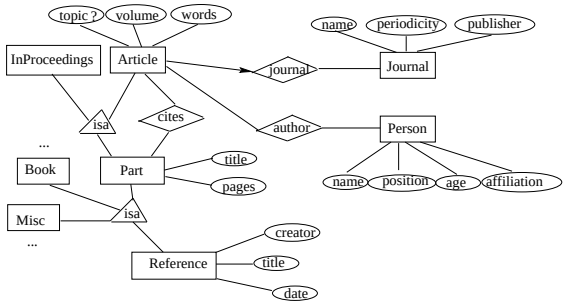


Figure 1: Small fragment of the schema ontology corresponding to a user view, using standard ER notation (rectangles represent concepts; circles represent attributes; triangles and diamonds represent *isa* or arbitrary relationships among concepts, respectively).

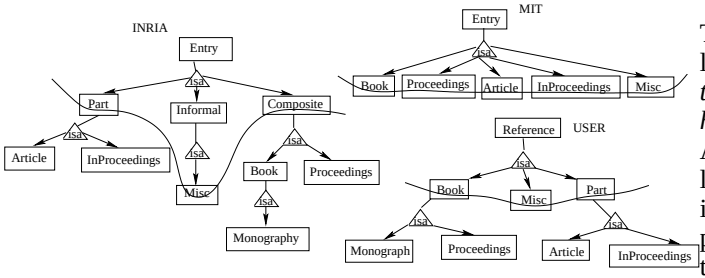


Figure 2: Small fragments of the SCHs corresponding to INRIA and MIT data sources and to a user, using standard ER notation (with attributes being omitted to avoid cluttering the figure). The set {`Book, Misc, Part`} determines a cut in the user hierarchy. Similar cuts are showed for MIT and INRIA SCHs.

USER->MIT	USER->INRIA
Reference=Entry Book=Book Monograph<Book Proceedings=Proceedings Misc=Misc Part<Entry Article=Article InProceedings=InProceedings	Reference=Entry Book=Book Monograph=Monography Proceedings=Proceedings Misc=Misc Part=Part Article=Article InProceedings=InProceedings

Figure 3: Interoperation constraints from the user to the MIT and INRIA SCHs.

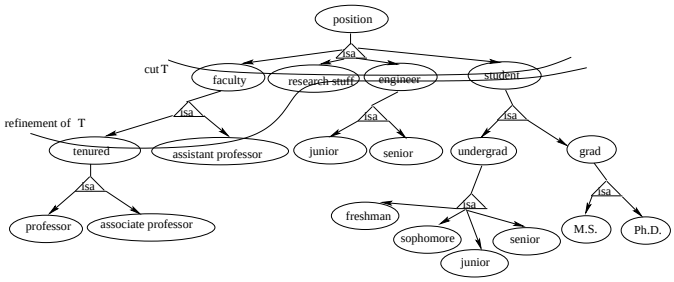


Figure 4: AVH associated with the attribute *position* of the concept *Author*. The set {*faculty, research staff, engineer, student*} represents a cut Γ through this hierarchy. The set {*tenured, assistant professor, research staff, engineer, student*} is a refinement of the cut Γ.

the schemas and the ontologies are made *explicit*; the instances in the data sources are labeled according to some criterion of interest to a user (e.g., topic categories).

(2) A user view, consisting of a user ontology O_U and a set of mappings ψ_k that relate the user ontology to the data source ontologies $O_1, \cdots, O_p$. The user view implicitly specifies a user level of abstraction, corresponding to the leaf nodes of the hierarchies in O_U. The mappings ψ_k can be specified manually by a user or semi-automatically derived.

(3) A hypothesis class H (e.g., Bayesian classifiers) defined over an *instance space* (implicitly specified by the concepts, their properties, and the associated ontologies in the domain of interest) and a performance criterion P (e.g., accuracy on a classification task).

The problem of learning classifiers from a collection of related OERDSs can be simply formulated as follows: *under the assumptions (1)-(3), the task of a learner L is to output a hypothesis $h \in H$ that optimizes P, via the mappings $\{\psi_k\}$.* As in [Caragea *et al.*, 2005], we say that an algorithm $\mathcal{L}_s$ for learning from OERDSs $\mathcal{D}_1, \cdots, \mathcal{D}_p$, via the mappings $\{\psi_k\}$, is *exact* relative to its centralized counterpart $\mathcal{L}_c$, if the hypothesis produced by $\mathcal{L}_s$ (federated approach) is identical to that obtained by $\mathcal{L}_c$ from the data warehouse $\mathcal{D}$ constructed by integrating the data sources $\mathcal{D}_1, \cdots, \mathcal{D}_p$, according to the user view, via the same mappings ψ_i (data warehouse approach).

The *exactness* criterion defined above assumes that it is possible, in principle, to create an integrated data warehouse in the centralized setting. However, in practice, the data sources $\mathcal{D}_1, \cdots, \mathcal{D}_p$ might impose access constraints Z on a user U. For example, data source constraints might prohibit retrieval of raw data from some data sources (e.g., due to query form access limitations, memory or bandwidth limitations, privacy concerns) while allowing retrieval of answers to statistical queries (e.g., count frequency queries).

2.4 Partially Specified Data

Because different data sources might specify data at different levels of abstraction (relative to a user's view), integration of OERDSs via mappings can result in data that is only partially specified. This can take the form of *partially specified schemas* (when schema concepts are partially specified) and *partially specified attributes* (when attribute values are partially specified).

The concept `Book` in the MIT hierarchy is under-specified (higher level of abstraction) with respect to (wrt) the concept `Monograph` in the user hierarchy, since a `Book` may be a `Monograph` or a `Proceedings`. On the other hand, a `Monography` in the INRIA hierarchy is fully specified (same level of abstraction) wrt a `Monograph` in the user hierarchy. Furthermore, an `Article` in the INRIA hierarchy is over-specified (lower level of abstraction) wrt a `Part` in the user hierarchy, as any `Article` is a `Part` (of a journal). We say that: a schema concept X_i in an SCH C is *partially specified* (or *under-specified*) wrt a schema concept X_j in an

equivalent SCH C' if $X_i > X_j$; X_i is *over-specified* wrt X_j if $X_i < X_j$; X_i is *fully specified* wrt X_j if $X_i = X_j$.

The attribute *grad* is under-specified wrt *Ph.D.*, since a *grad* may be a *Ph.D.* or a *M.S.*, but over-specified wrt *student* as every *grad* is a *student*. Furthermore, *freshman* is fully specified wrt *1st year*. We say that: an attribute value $v_i \in \mathcal{V}(A)$ is *partially specified* (or *under-specified*) wrt an attribute value $v_j \in \mathcal{V}(A')$ if $v_i > v_j$; v_i is *over-specified* wrt v_j if $v_i > v_j$; v_i is *fully-specified* wrt v_j if $v_i = v_j$.

Note that the problem of partially specified data (when attributes are partially specified) can be seen as a generalization of the problem of missing attribute values [Zhang *et al.*, 2005], and hence it is possible to adapt statistical approaches for dealing with missing data [Little and Rubin, 2002] to deal with partially specified data *under appropriate assumptions*, (e.g., that the distribution of an under-specified attribute value is similar to that in a data source where the corresponding attribute is fully specified). Partially specified concepts pose additional challenges. Some approaches to handling partially specified concepts are: ignore a concept that becomes under-specified in a schema; or alternatively, use only the attributes that a concept inherits from its parents, while the rest (e.g., attributes specific to that concept that are not inherited from the parent) are treated as missing in all instances of the concept in that data source.

2.5 Sufficient Statistics Based Solution

Our approach to the problem of learning classifiers from OERDSs is a natural extension of a general strategy for transforming algorithms for learning classifiers from data in the form of a single flat table (as is customary in the case of a vast majority of standard machine learning algorithms) into algorithms for learning classifiers from a collection of *horizontal* or *vertical* fragments of the data, corresponding to partitions of rows or columns of the flat table, wherein each fragment corresponds to an ontology extended data source. This strategy, inspired by [Kearns, 1998] involves a decomposition of a learning task into two parts: a *statistics gathering* component, which retrieves the statistics needed by the learner from the distributed data sources, and a *hypothesis refinement* component, which uses the statistics to refine a partially constructed hypothesis (starting with an empty hypothesis) [Caragea *et al.*, 2005].

In the case of learning classifiers from semantically disparate OERDSs, the statistics gathering component has to specify the statistics needed for learning as a *query* against the user view and assemble the answer to this query from OERDSs. This entails: decomposition of a posed query into sub-queries that the individual data sources can answer; translation of the sub-queries to the data source ontologies, via user-specific mappings; query answering from (possibly) partially specified data sources; composition of the partial answers into a final answer to the initial query (Figure 5).

3 Illustration of the Proposed Approach

The strategy outlined in the previous section can be used to design algorithms that are exact relative to their centralized (integrated data warehouse) counterparts for learning naive

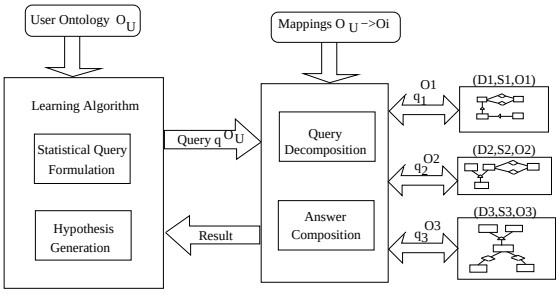
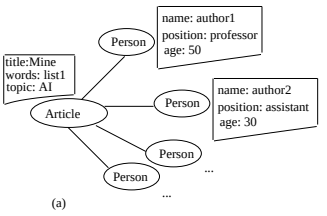


Figure 5: Learning classifiers from OERDSs

Bayes classifiers, decision trees, Bayesian networks, etc., from OERDSs (note that the sufficient statistics for such algorithms are frequency counts). For simplicity, we will use Relational Bayesian Classifiers [Neville *et al.*, 2003] to illustrate this strategy.

3.1 Relational Bayesian Classifiers (RBCs)

In contrast to the single flat table structure assumed by the vast majority of machine learning algorithms, data sources on the Semantic Web are better modeled by relational data sources, where each instance can have a different number of related objects, and therefore, a different number of *features* [Neville *et al.*, 2003]. Figure 6 (a) shows a sample relational graph corresponding to the concept `Article` in the relational reference domain. Note that each `Article` can have a variable number of `Authors`. A relational graph can be transformed into a table with a fix number of attributes, where each attribute A_i will have a *multiset* of values V_i, like in Figure 6 (b). Under the naive Bayes assumption that the attributes are independent given the class, methods for estimating probabilities involving multisets (such as $p(V_i|c_j)$) and ways to use them for inference are needed.



topic	words	title	authorName	authorPos	authorAge
AI	list1	Mine	{author1, author2,...}	{professor, assistant,..}	{50,30,...}
DB	list2	Data	{person1, person2,...}	{assistant, assistant,..}	{35,32,...}
...	...	...	...	...	...

(b)

Figure 6: (a) Graph corresponding to a relational schema (b) Relational data decomposed by (multiset) attributes.

Two different approaches for estimating such probabilities and three different approaches to inference are described in [Neville *et al.*, 2003]:

- **Average Value** (AVGVAL): To estimate probabilities, data is flattened by averaging, i.e., each multiset is replaced with the average value (continuous attributes) or mode value (discrete attributes). In this case, the most probable class is: $c_{MAP}(x) = \underset{c_j \in C}{\text{argmax}}\, p(c_j) \prod_i p(\text{mode}(V_i)|c_j)$.

- **Independent Value** (INDVAL): To estimate probabilities, data is flattened by assuming that each value in a multiset is independent of the others. An *instance* is created for each value in the multiset. In this case, the most probable class is: $c_{MAP}(x) = \underset{c_j \in C}{\text{argmax}}\, p(c_j) \prod_i \prod_{v_k \in V_i} p(v_k|c_j)$.
- **Average Probability** (AVGPROB): As in the case of INDVAL, data is flattened by assuming that each value in a multiset is independent of the others. However, in the case of AVGPROB, the most probable class is: $c_{MAP}(x) = \underset{c_j \in C}{\text{argmax}}\, p(c_j) \prod_i \frac{\sum_{v_k \in V_i} p(v_k|c_j)}{|V_i|}$.

3.2 Learning RBC from OERDS

We observe that the task of any of the RBCs above reduces to estimating the probabilities $p(c_j)$ and $p(v_i|c_j)$, for all class labels $c_j \in \mathbf{C}$ and for all attribute values $v_i \in \mathcal{V}(A_i)$. These probabilities can be estimated from data using standard methods [Mitchell, 1997]. The resulting estimates constitute sufficient statistics for the parameters that specify an RBC.

Now, we return to the task of constructing an RBC from a set of OERDSs $\mathcal{D}_1, \cdots, \mathcal{D}_p$, from a user's view under the assumptions (1)-(3) in the Problem Formulation section. We decompose the learning task into sufficient statistics gathering and hypothesis refinement components, thereby reducing the problem of learning RBCs from OERDSs to the problem of answering queries for the relevant statistics (e.g., frequency counts) against the user's view of the set of available OERDSs.

We denote by $\sigma(v_i|c_j)$ the frequency count of the value v_i of the attribute A_i given the class label c_j, and by $\sigma(c_j)$ the frequency count of the class label c_j, in the user view. The algorithm for learning an RBC from a set of related OERDSs works as follows:

- Select a global user cut Γ through the user ontology (both SCHs and AVHs). In particular, the user cut can correspond to the set of primitive values (i.e., leaves in the hierarchies), as in the case of the traditional RBCs.
- Apply the mappings $\{\psi_k\}$ to find a cut Γ_k, corresponding to the user cut Γ, in each data source $\mathcal{D}_k$.
- Formulate statistical queries asking for relative frequency counts $\sigma(v_i|c_j)$ and $\sigma(c_j)$, using terms in the user cut Γ.
- Translate these queries to queries expressed in the ontology of each data source $\mathcal{D}_k$, using terms in the data source cut Γ_k, and compute the local counts $\sigma^k(v_i|c_j)$ and $\sigma^k(c_j)$ from each OERDS $\mathcal{D}_k$.
- Send the local counts to the user and add them up to compute the global frequency counts $\sigma(v_i|c_j)$ and $\sigma(c_j)$.
- Generate the RBC h_Γ corresponding to the cut Γ based on the global frequency counts.

Note that if the cut Γ corresponds to the primitive concepts and values in the user hierarchies, the resulting RBC is *exact* with respect to the traditional RBC obtained, in principle, by integrating all the OERDSs $\mathcal{D}_1, \cdots, \mathcal{D}_p$ into a central data warehouse $\mathcal{D}$ (using the same set of mappings $\{\psi_k\}$ and the same assumptions for dealing with partially specified concepts and attribute values). This is true because $\sigma(v_i|c_j) = \sum_{i=1}^{k} \sigma^k(v_i|c_j) = \sigma^{\mathcal{D}}(v_i|c_j)$ when there is no overlap between the distributed data sources. However, construction of such an integrated centralized data warehouse might require violation of data source access constraints (Z), and hence a learning strategy relying on a centralized data warehouse may be unimplementable in practice. In contrast, the approach presented in this paper makes it possible to obtain the same classifier, as obtainable from an integrated centralized data warehouse, while circumventing the need for such a warehouse.

4 Discussion and Future Work

In this paper, we have precisely formulated the problem of learning classifiers from a collection of several related OERDSs, which make *explicit* (the typically implicit) ontologies associated with the data sources of interest. User-specific mappings between the user ontology and data source ontologies are used to answer statistical queries that provide the sufficient statistics needed for learning classifiers from OERDSs.

These mappings can be specified by the user or obtained semi-automatically [Kalfoglou and Schorlemmer, 2005] using the user's preferred method for finding mappings between the user ontology O_U and data source ontologies $O_1, \cdots, O_p$. The quality of the classifier in our setting, very likely, depends on the quality of the mappings, just as the quality of a classical classifier depends on the quality of the data (i.e., noisy data or imprecise mappings may result in very poor classifiers). In many application domains (e.g., bioinformatics), community-driven efforts are underway to develop carefully curated mappings between ontologies of interest. The cost of such efforts may be justified in some application domains, whereas automatically derived mappings may be adequate in other domains. It should be noted that even manually derived mappings are often application, user, or context specific. Thus, users may have different views of the domain and, hence, may want to use different mappings.

The proposed algorithms for learning from OERDSs are *provably exact* relative to their centralized counterparts, for a family of learning classifiers for which the sufficient statistics take the form of counts of instances satisfying certain constraints on the values of the attributes. We have illustrated the proposed approach in the case of learning RBCs from several related OERDSs. The proposed approach generalizes to other learning algorithms (e.g., decision trees, Relational Bayesian Networks).

The algorithm presented here assumes a prespecified level of abstraction defined by the user-supplied global cut through the user ontology. This algorithm could be improved further, if we consider a top down, iterative approach to refining the user cut (see Figure 4), starting with the most abstract cuts through each hierarchy in the user ontology until an "optimal cut" and, thus, an optimal level of abstraction is identified for the learning task at hand. This strategy is similar to that adopted in [Zhang *et al.*, 2005] in learning naive Bayes

classifiers from a single flat table, in the presence of attribute value taxonomies and partially specified data. Such an approach, in addition to trading off the complexity of the classifier against accuracy on training data (with choice of more abstract cuts through hierarchies that make up the user ontology yielding *simpler* classifiers), offers a defensive strategy in the presence of partially specified data: the more abstract the cut through user ontology, the lower the chances of encountering partially specified data during learning.

The efficiency of the proposed approach (relative to the centralized setting) depends on the specifics of access constraints and query answering capabilities associated with the individual OERDSs. At present, many data sources on the Web offer query interfaces that can only be used to retrieve small subsets of the data that match a limited set of conditions that can be selected by the user. In order for Web data sources to serve the needs of communities of users interested in building predictive models from the data (e.g., in e-science, and other emerging data-rich applications), it would be extremely useful to equip the data sources with statistical query answering capabilities.

In summary, we have demonstrated, under fairly general assumptions, how to exploit data sources annotated with relevant meta data in building predictive models (e.g., classifiers) from several related OERDSs, without the need for a centralized data warehouse, while offering strong guarantees of *exactness* of the learned classifiers wrt the centralized traditional relational learning counterparts. Some interesting directions for future research include: exploring the effect of using different ontologies and mappings, use of the proposed framework to evaluate mappings, study of the quality of the classifier with respect to the set of mappings used, etc.

References

[Berners-Lee *et al.*, 2001] T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American*, 284:28–37, May 2001.

[Bonatti *et al.*, 2003] P. Bonatti, Y. Deng, and V. Subrahmanian. An ontology-extended relational algebra. In *Proceedings of the IEEE Conference on Information Integration and Reuse*, pages 192–199. IEEE Press, 2003.

[Calvanese *et al.*, 2005] Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, and Moshe Y. Vardi. View-based query processing: On the relationship between rewriting, answering and losslessness. In *Proceedings of the 10th International Conference on Database Theory (ICDT 2005)*, volume 3363 of *LNCS*, pages 321–336. Springer, 2005.

[Caragea *et al.*, 2005] D. Caragea, J. Zhang, J. Bao, J. Pathak, and V. Honavar. Algorithms and software for collaborative discovery from autonomous, semantically heterogeneous information sources. In *Proceedings of the 16th International Conference on Algorithmic Learning Theory*, volume 3734 of *LNCS*, pages 13–44, Singapore, 2005. Berlin: Springer-Verlag.

[Doan and Halevy, 2005] A. Doan and A. Halevy. Semantic Integration Research in the Database Community: A Brief Survey. *AI Magazine, Special Issue on Semantic Integration*, 26(1):83–94, 2005.

[Getoor *et al.*, 2001] L. Getoor, N. Friedman, D. Koller, and A. Pfeffer. Learning probabilistic relational models. In S. Dzeroski and Eds. N. Lavrac, editors, *Relational Data Mining*. Springer-Verlag, 2001.

[Hotho *et al.*, 2003] A. Hotho, Steffen Staab, and Gerd Stumme. Ontologies improve text document clustering. In *Proceedings of The Third IEEE International Conference on Data Mining*, 2003.

[Kalfoglou and Schorlemmer, 2005] Yannis Kalfoglou and Marco Schorlemmer. Ontology mapping: The state of the art. In *Dagstuhl Seminar Proceedings: Semantic Interoperability and Integration*, Dagstuhl, Germany, 2005.

[Kargupta and Chan, 2000] H. Kargupta and P. Chan. *Advances in Distributed and Parallel Knowledge Discovery*. AAAI/MIT, 2000.

[Kearns, 1998] M. Kearns. Efficient noise-tolerant learning from statistical queries. *Journal of the ACM*, 45(6):983–1006, 1998.

[Kushmerick *et al.*, 2005] Nicholas Kushmerick, Fabio Ciravegna, AnHai Doan, Craig Knoblock, and Steffen Staab, editors. *Proceedings of Dagstuhl Seminar on Machine Learning for the Semantic Web*, 2005.

[Levy, 2000] A. Levy. Logic-based techniques in data integration. In *Logic-based artificial intelligence*, pages 575–595. Kluwer Academic Publishers, 2000.

[Little and Rubin, 2002] Roderick J. A. Little and Donald B. Rubin. *Statistical Analysis with Missing Data*. Wiley, 2002.

[McCallum *et al.*, 2000] Andrew McCallum, Kamal Nigam, Jason Rennie, and Kristie Seymore. Automating the contruction of internet portals with machine learning. *Information Retrieval Journal*, 3:127–163, 2000.

[Mitchell, 1997] T.M. Mitchell. *Machine Learning*. McGraw Hill, 1997.

[Neville *et al.*, 2003] J. Neville, D. Jensen, and B. Gallagher. Simple estimators for relational bayesian classifiers. In *Proceedings of the Third IEEE International Conference on Data Mining*. IEEE Press, 2003.

[Noy and Stuckenschmidt, 2005] N. Noy and H. Stuckenschmidt. Ontology Alignment: An annotated Bibliography. In Y. Kalfoglou, M. Schorlemmer, A. Sheth, S. Staab, and M. Uschold, editors, *Semantic Interoperability and Integration*, number 04391 in Dagstuhl Seminar Proceedings, 2005.

[Noy, 2004] N.F. Noy. Semantic Integration: A Survey Of Ontology-Based Approaches. *SIGMOD Record, Special Issue on Semantic Integration*, 33(4), 2004.

[Zhang *et al.*, 2005] J. Zhang, D-K. Kang, A. Silvescu, and V. Honavar. Learning compact and accurate naive bayes classifiers from attribute value taxonomies and data. *Knowledge and Information Systems*, 2005.