Chapter 1

Towards Semantics-Enabled Infrastructure for Knowledge Acquisition from Distributed Data

Vasant Honavar

Iowa State University

Doina Caragea

Kansas State University

Abstract We summarize progress on algorithms and software knowledge acquisition from large, distributed, autonomous, and semantically disparate information sources. Some key results include: scalable algorithms for constructing predictive models from data based on a novel decomposition of learning algorithms that interleave queries for sufficient statistics from data with computations using the statistics; provably exact algorithms from distributed data (relative to their centralized counterparts); and statistically sound approaches to learning predictive models from partially specified data that arise in settings where the schema and the data semantics and hence the granularity of data differ across the different sources.

1.1 Introduction

Recent development of high throughput data acquisition technologies in a number of domains (e.g., biological sciences, atmospheric sciences, space sciences, commerce) together with advances in digital storage, computing, and communications technologies have resulted in the proliferation of a multitude of physically distributed data repositories created and maintained by

autonomous entities (e.g., scientists, organizations). The resulting increasingly data rich domains offer unprecedented opportunities in computer assisted data-driven knowledge acquisition in a number of applications including in particular, data-driven scientific discovery in bioinformatics (e.g., characterization of protein sequence-structure-function relationships in computational molecular biology), environmental informatics, and health informatics. Machine learning algorithms [54, 66, 39, 11] offer some of the most cost-effective approaches to knowledge acquisition (discovery of features, correlations, and other complex relationships and hypotheses that describe potentially interesting regularities) from large data sets. However, the applicability of current approaches to machine learning in emerging data rich applications in practice is severely limited by a number of factors:

- Data repositories are large in size, dynamic, and physically distributed. Consequently, it is neither desirable nor feasible to gather all of the data in a centralized location for analysis. In some domains, the ability of autonomous organizations to share raw data may be limited due to a variety of reasons (e.g., privacy considerations) [68]. In both cases, there is a need for efficient algorithms for learning from multiple distributed data sources without the need to transmit large amounts of data.

- Autonomously developed and operated data sources often differ in their structure and organization (relational databases, flat files, etc.). Furthermore, the data sources often limit the operations that can be performed (e.g., types of queries - relational queries, restricted subsets of relational queries, statistical queries; execution of user-supplied code to compute answers to queries that are not directly supported by the data source). Hence, there is a need for effective strategies for efficiently obtaining the information needed for learning under the operational constraints imposed by the data sources, and theoretical guarantees about the performance of the resulting classifiers relative to the setting in which the learning algorithm has unconstrained access to a centralized data set.

- Autonomously developed data sources differ with respect to data *semantics*. The Semantic Web enterprise [9] is aimed at making the contents of the Web machine interpretable. Data and resources on the Web are annotated and linked by associating metadata 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. The increasing need for information sharing between organizations, individuals and scientific communities has led to several community-wide efforts aimed at the construction of ontologies in several domains. Explicit specification of the ontology associated with a data repository helps standardize the semantics to an extent. Collaborative scientific discovery applications often require users to be able to analyze data from

multiple, semantically disparate data sources from different perspectives in different contexts. In particular, there is no single privileged perspective that can serve all users, or for that matter, even a single user, in every context. Hence, there is a need for methods that can efficiently obtain from a federation of autonomous, distributed, and semantically heterogeneous data sources, the information needed for learning (e.g., statistics) based on user-specified semantic constraints between user ontology and data-source ontologies.

Against this background, we consider the problem of data driven knowledge acquisition from autonomous, distributed, semantically heterogeneous, data sources.

1.2 Learning from Distributed Data

Given a data set D, a hypothesis class H, and a performance criterion P, an algorithm L for learning (from centralized data D) outputs a hypothesis $h \in H$ that optimizes P. In pattern classification applications, h is a classifier (e.g., a decision tree.) The data D consists of a (multi)set of training examples. Each training example is an ordered tuple of attribute values, where one of the attributes corresponds to a class label and the remaining attributes represent inputs to the classifier. The goal of learning is to produce a hypothesis that optimizes the performance criterion (e.g., minimizing classification error on the training data) and the complexity of the hypothesis. In a distributed setting, a data set D is distributed among the sites $1, \cdots, n$ containing data set fragments $D_1, \cdots, D_n$. Two simple (and common) types of data fragmentation are: horizontal fragmentation and vertical fragmentation. More generally, the data may be fragmented into a set of relations (as in the case of tables of a relational database, but distributed across multiple sites). We assume that the individual data sets $D_1, \cdots, D_n$ collectively contain (in principle) all the information needed to construct the data set D.

The distributed setting typically imposes a set of constraints Z on the learner that are absent in the centralized setting. For example, the constraints Z may prohibit the transfer of raw data from each of the sites to a central location while allowing the learner to obtain certain types of statistics from the individual sites (e.g., counts of instances that have specified values for some subset of attributes), or in the case of knowledge discovery from clinical records, Z might include constraints designed to protect the privacy of patients.

The problem of learning from distributed data can be stated as follows: Given the fragments $D_1, \cdots, D_n$ of a data set D distributed across the sites $1, \cdots, n$, a set of constraints Z, a hypothesis class H, and a performance cri-

terion P, the task of the learner L_d is to output a hypothesis that optimizes P using only operations allowed by Z. Clearly, the problem of learning from a centralized data set D is a special case of learning from distributed data where $n = 1$ and $Z = \phi$. Having defined the problem of learning from distributed data, we proceed to define some criteria that can be used to evaluate the quality of the hypothesis produced by an algorithm L_d for learning from distributed data relative to its centralized counterpart. We say that an algorithm L_d for learning from distributed data sets $D_1, \cdots, D_n$ is *exact* relative to its centralized counterpart L if the hypothesis produced by L_d is identical to that obtained by L from the data set D obtained by appropriately combining the data sets $D_1, \cdots, D_n$.

Example [19] Let L be a centralized algorithm for learning a decision tree classifier [Quinlan, 1993] $h : \mathbf{X} \rightarrow C$ (where $\mathbf{X}$ is an instance space and C is a finite set of class labels) from data set $D \subset \mathbf{X} \times C$. Let L_d be an algorithm for learning a decision tree classifier $h_d : \mathbf{X} \rightarrow C$ under a set of specified constraints Z from horizontally fragmented distributed data $D_1, \cdots, D_n$, where each $D_i \subset D \subset \mathbf{X} \times C$. Suppose further that $D = \cup_{i=1}^{n} D_i$. Then we say that L_d is exact with respect to L if and only if $\forall X \in \mathbf{X}, h(X) = h_d(X)$.

Proof of exactness of an algorithm for learning from distributed data relative to its centralized counterpart ensures that a large collection of existing theoretical (e.g., sample complexity, error bounds) as well as empirical results obtained in the centralized setting carry over to the distributed setting.

1.2.1 General Strategy for Transforming Centralized Learners into Distributed Learners

Our general strategy for designing an algorithm for learning from distributed data that is provably exact with respect to its centralized counterpart (in the sense defined above) follows from the observation that most of the learning algorithms use only some *statistics* computed from the data D in the process of generating the hypotheses that they output. (Recall that a statistic is simply a function of the data.) This yields a natural decomposition of a learning algorithm into two components (Figure 1.1):

- an information extraction component that formulates and sends a statistical query to a data source and
- a hypothesis generation component that uses the resulting statistic to modify a partially constructed hypothesis (and further invokes the information extraction component as needed).

A statistic $s(D)$ is called a sufficient statistic for a parameter θ if $s(D)$, loosely speaking, provides all the information needed for estimating the parameter from data D [33]. Thus, sample mean is a sufficient statistic for the mean of a Gaussian distribution.

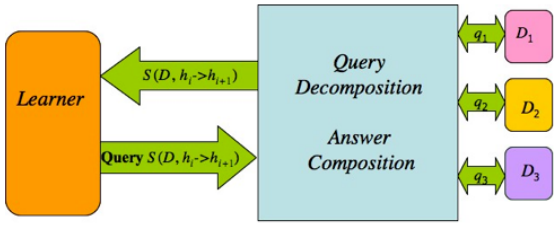

FIGURE 1.1: Learning = Statistical Query Answering + Hypothesis construction

Inspired by theoretical work on PAC learning from statistical queries [44], we have generalized this notion of a sufficient statistic for a parameter θ into a sufficient statistic $s_{L,h}(D)$ for learning a hypothesis h using a learning algorithm L applied to a data set D [19, 21]. Trivially, the data D and the hypothesis h are both sufficient statistics for learning h using L. We are typically interested in statistics that are minimal or at the very least, substantially smaller in size (in terms of the number of bits needed for encoding) than the data set D. In some simple cases, it is possible to extract a sufficient statistic $s_{L,h}(D)$ for constructing a hypothesis h in one step (e.g., by querying the data source for a set of conditional probability estimates when L is the standard algorithm for learning a Naive Bayes classifier). In a more general setting, h is constructed by L by interleaving information extraction (statistical query) and hypothesis construction operations. Thus, a decision tree learning algorithm would start with an empty initial hypothesis h_0, obtain the sufficient statistics (expected information concerning the class membership of an instance associated with each of the attributes) for the root of the decision tree (a partial hypothesis h_1), and recursively generate queries for additional statistics needed to iteratively refine h_1 to obtain a succession of partial hypotheses $h_1, h_2, \cdots$ culminating in h (See Figure 1.2).

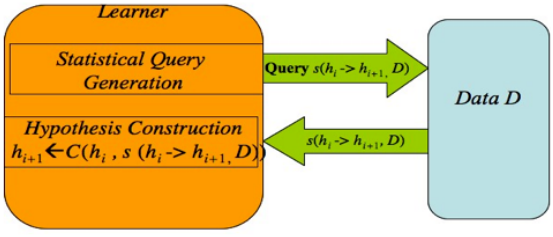

FIGURE 1.2: Learning from Distributed Data = Statistical Query Answering from Distributed Data + Hypothesis construction

In this model, the only interaction of the learner with the repository of data D is through queries for the relevant statistics. Information extraction from distributed data entails decomposing each statistical query q posed by the information extraction component of the learner into sub queries $q_1, \cdots, q_n$ that can be answered by the individual data sources $D_1, \cdots, D_n$ respectively, and a procedure for combining the answers to the sub queries into an answer for the original query q (See Figure 1.2).

We have shown that this general strategy for learning classifiers from distributed data is applicable to a broad class of algorithms for learning classifiers from data [19]. Consequently, for these algorithms, we can devise a strategy (plan) for computing h from the data D using sufficient statistics. When the learner's access to data sources is subject to constraints Z, the resulting plan for information extraction has to be executable without violating the constraints Z. Given provably correct query decomposition and answer composition procedures, it is easy to establish the *exactness* of the algorithm L_d for learning from distributed data relative to its centralized counterpart.

We have applied the general framework described above for construction of algorithms for learning classifiers from distributed data to design provably exact algorithms for learning Naive Bayes, Nearest Neighbor, Bayes Network, Neural Network, and Decision Tree classifiers from distributed data under horizontal and vertical data fragmentation [19], and Support Vector Machine (SVM) classifiers under horizontal data fragmentation (at the expense of multiple passes through the distributed data). We have also established the precise conditions under which the proposed algorithms offer significant savings in bandwidth, memory, and/or computation time (relative to their centralized counterparts) [19].

1.2.2 Related Work on Learning Classifiers from Distributed Data

Srivastava et al. [63] propose methods for distributing a large centralized data set to multiple processors to exploit parallel processing to speed up learning. Grossman and Guo [35] and Provost and Kolluri [58] survey several methods that exploit parallel processing for scaling up data mining algorithms to work with large data sets. In contrast, the focus of our work is on learning classifiers from a set of autonomous distributed data sources. The autonomous nature of the data sources implies that the learner has little control over the manner in which the data are distributed among the different sources. Distributed data mining has received considerable attention in the literature [55]. Domingos [29] and Prodromidis et al. [57] propose an *ensemble of classifiers* approach to learning from horizontally fragmented distributed data which essentially involves learning separate classifiers from each data set and combining them typically using a weighted voting scheme. This requires gathering a subset of data from each of the data sources at a central site to determine the weights to be assigned to the individual hy-

potheses (or shipping the ensemble of classifiers and associated weights to the individual data sources where they can be executed on local data to set the weights). In contrast, our approach is applicable even in scenarios which preclude transmission of data or execution of user-supplied code at the individual data sources but allow transmission of minimal sufficient statistics needed by the learning algorithm. A second potential drawback of the ensemble of classifiers approach to learning from distributed data is that the resulting ensemble of classifiers is typically much harder to comprehend than a single classifier. A third important limitation of the ensemble classifier approach to learning from distributed data is the lack of strong guarantees concerning accuracy of the resulting hypothesis relative to the hypothesis obtained in the centralized setting. Bhatnagar and Srinivasan [10] propose an algorithm for learning decision tree classifiers from vertically fragmented distributed data. Kargupta et al. [43] describe an algorithm for learning decision trees from vertically fragmented distributed data using a technique proposed by Mansour [53] for approximating a decision tree using Fourier coefficients corresponding to attribute combinations whose size is at most logarithmic in the number of nodes in the tree. At each data source, the learner estimates the Fourier coefficients from the local data, and transmits them to a central site where they are combined to obtain a set of Fourier coefficients for the decision tree (a process which requires a subset of the data from each source to be transmitted to the central site). However, a given set of Fourier coefficients can correspond to multiple decision trees. Furthermore, there are no guarantees concerning the performance of the hypothesis obtained in the distributed setting relative to that obtained in the centralized setting.

Relative to the large body of work on learning classifiers from distributed data, the distinguishing feature of our approach is a clear separation of concerns between hypothesis construction and extraction of sufficient statistics from data. This makes it possible to explore the use of sophisticated techniques for query optimization that yield optimal plans for gathering sufficient statistics from distributed data sources under a specified set of constraints describing the query capabilities of the data sources, operations permitted by the data sources, and available computation, bandwidth, and memory resources. It also opens up the possibility of exploring algorithms that learn from distributed data a hypothesis h_ϵ whose error is small relative to the error of a hypothesis h (obtained in the setting when the learner has unrestricted access to D), in scenarios where the constraints Z make it impossible to guarantee *exactness* in the sense defined above. Our approach also lends itself to adaptation to learning from semantically heterogeneous data sources.

1.3 Learning from Semantically Heterogeneous Data

In order to extend our approach to learning from distributed data (which assumes a common ontology that is shared by all of the data sources) into effective algorithms for learning classifiers from *semantically heterogeneous* distributed data, techniques need to be developed for answering the statistical queries posed by the learner in terms of the learner's ontology O from the heterogeneous data sources (where each data source D_i has an associated ontology O_i). Thus, we have to solve a variant of the problem of integrated access to distributed data repositories - the data integration problem [49, 17] in order to be able to use machine learning approaches to acquire knowledge from semantically heterogeneous data.

This problem is best illustrated by an example (Figure 1.3). Consider two

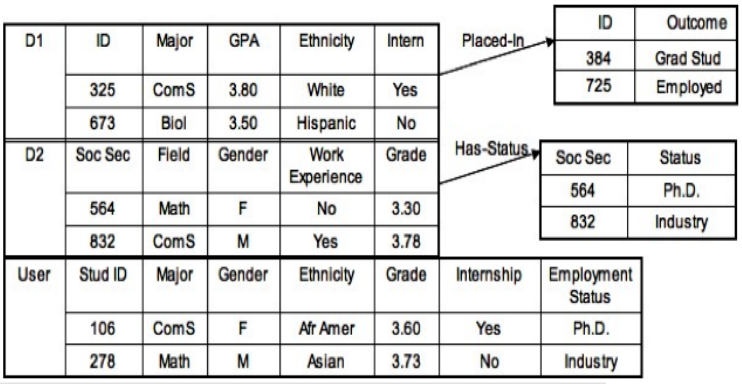

FIGURE 1.3: Student data collected by two departments from a statistician's perspective.

academic departments that independently collect information about their students. Suppose a data set D_1 collected by the first department is organized in two tables: *Student*, and *Outcome*, linked by a *Placed-In* Relation using ID as the common key. Students are described by *ID*, *Major*, *GPA*, *Ethnicity* and *Intern*. Suppose a data set D_2 collected by the second department has a *Student* table and a *Status* table, linked by *Has-Status* relation using *Soc Sec* as the common key. Suppose *Student* in D_2 is described by the attributes *SocSec*, *Field*, *Gender*, *Work-Experience* and *Grade*.

Consider a user, e.g., a university statistician, interested in constructing a predictive model based on data from two departments of interest from his or her own perspective, where the representative attributes are *Student ID*,

Major, *Gender*, *Ethnicity*, and *Grade*, *Internship* and *Employment Status*. For example, the statistician may want to construct a model that can be used to infer whether a typical student (represented as in the entry corresponding to D_U in Figure 1.3) is likely go on to get a *Ph.D.* This requires the ability to perform queries over the two data sources associated with the departments of interest from the user's perspective (e.g., *fraction of students with internship experience that go onto Ph.D*). However, because the structure (schema) and data semantics of the data sources differ from the statistician's perspective, he/she must establish the correspondences between the attributes of the data and the values that make up their domains.

We adopt a federated, query-centric approach to answering statistical queries from semantically heterogeneous data sources, based on ontology-extended relational algebra [12]. Specifically, we associate explicit ontologies with data sources to obtain *ontology extended relational data sources* (OERDS). An OERDS is a tuple $\mathcal{D} = \{D, S, O\}$, where D is the actual data set in the data source, S the data source schema and O the data source ontology [19, 20].

A relational *data set* D is an instantiation $I(S)$ of a schema S. 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 (entities, and attributes of entities that appear in data source schema S); and *content ontology*, O_I, that defines the semantics of the data instances (values and relationships between values that the attributes can take in instantiations of schema S). Of particular interest are ontologies that take the form of *is-a* hierarchies and *has-part* hierarchies. For example, the values of the *Status* attribute in data source D_2 are organized in an *is-a* hierarchy. A *user's view of data sources* $D_1, D_2, \cdots, D_n$ *is specified* by user schema S_U, user ontology O_U, together with a set of *semantic constraints* IC, and the associated set of *mappings* from the user schema S_U to the data source schemas $S_1, \cdots, S_n$ and from user ontology O_U to the data source ontologies $O_1, \cdots, O_n$ [20]. Figure 1.4 shows examples of ontologies that take the form of *is-a* hierarchies over attribute values. Figure 1.5 shows some simple examples of user-specified semantic constraints between the user perspective and the data sources D_1 and D_2, respectively.

How can we answer a statistical query in a setting in which autonomous data sources differ in terms of the levels of abstraction at which data are described? For example: Consider the data source ontologies O_1 and O_2 and the user ontology O_U shown in Figure 1.4. The attribute *Status* in data source D_2 is specified in greater detail (lower level of abstraction) than the corresponding attribute *Outcome* is in D_1. That is, data source D_2 carries information about the precise status of students after they graduate (specific advanced degree program e.g., *Ph.D.*, *M.S.* that the student has been accepted into, or the type of employment that the student has accepted), whereas data source D_1 makes no distinctions between the types of graduate degrees or types of employment. Suppose we want to answer the query: What fraction of the students in the two data sources got into a *Ph.D.* program? Answering this query is complicated by the fact that the *Outcome* of students in data source D_1 are only *partially*

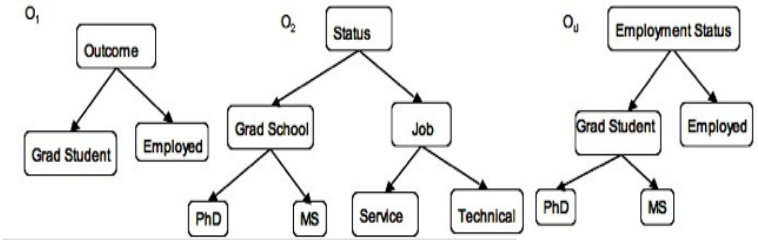

FIGURE 1.4: Attribute value taxonomies (ontologies) O_1 and O_2 associated with the attributes *Outcome* and *Status* in two data sources of interest. O_U is the ontology for *Employment Status* from the user's perspective.

$O_1 \rightarrow O_U$	$O_2 \rightarrow O_U$
ID:O_1=Stud ID:O_U	Soc Sec:O_2=Stud ID:O_U
Major:O_1=Major:O_U	Field:O_2=Major:O_U
GPA:O_1=Grade:O_U	Grade:O_2=Grade:O_U
Ethnicity:O_1=Ethnicity :O_U	
	Gender:O_2=Gender:O_U
Intern:O_1=Internship:O_U	Work-Experience:O_2=Internship:O_U
Outcome:O_1=Employment-Status:O_U	Status:O_2=Employment-Status:O_U

FIGURE 1.5: An example of user-specified semantic correspondences between the user ontology O_U and data source ontologies.

specified [74, 75, 76] with respect to the ontology O_U. Consequently, we can never know the precise fraction of students that got into a *Ph.D.* program based on the information available in the two data sources. In such cases, answering statistical queries from semantically heterogeneous data sources requires the user to supply not only the mapping between the ontology and the ontologies associated with the data sources but also *additional assumptions of a statistical nature* (e.g., that grad program admits in D_1 and D_2 can be modeled by the same underlying distribution). The validity of the answer returned depends on the validity of the assumptions and the soundness of the procedure that computes the answer based on the supplied assumptions.

Given a means of answering statistical queries from semantically heterogeneous data, we can devise a general framework for learning predictive models from such data (See Figure 1.6). Based on this framework, we have imple-

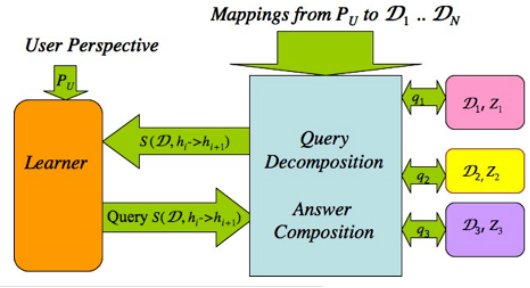

FIGURE 1.6: General Framework for learning classifiers from semantically heterogeneous distributed data.

mented a prototype of an *intelligent data understanding system* (INDUS) that supports: execution of statistical queries against semantically heterogeneous ontology extended data sources, and the construction of predictive models (e.g., classifiers) from such data sources (See Figure 1.7). More precisely, INDUS system enables a user with some familiarity with the relevant data to query multiple data sources from his or her own point of view by selecting data sources of interest, specifying the user perspective and the necessary mappings all without having to write any code. Queries posed by the user are sent to a query-answering engine (QAE) that automatically decomposes the user query q_U expressed in terms of the user ontology O_U into queries $q_1, \cdots, q_n$ that can be answered by the individual data sources. QAE combines the answers to individual queries (after applying the necessary mappings) to generate the answer for the user query q_U . The soundness of the data integration process (relative to a set of user-specified mappings between ontologies) follows from the soundness of the query decomposition procedure, the correctness of the behavior of the query answering engines associated with the individual data sources,

and the answer composition procedure [20, 23]. The current implementation

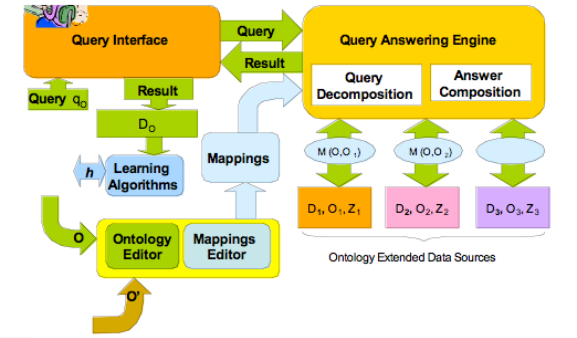

FIGURE 1.7: The INDUS System

of INDUS (http://www.cild.iastate.edu/software/indus.html) which has been released under Gnu public license includes support for:

- Import and reuse of selected fragments of existing ontologies and editing of ontologies. We have more recently developed semantic importing mechanism (based on a formalization of localized semantics) [5, 6, 4].
- Specification of semantic correspondences between a user ontology O_U and data source ontologies [19]. Semantic correspondences between ontologies can be defined at two levels: schema level (between attributes that define data source schemas) and attribute level (between values of attributes). Implementation of an efficient reasoning algorithm for verifying the consistency of subsumption and equivalence relationships [7].
- Registration of a new ontology-extended data source using a data-source editor for defining the schema of the data source, location, data source ontology, and data source constraints (Z).
- Specification and execution of queries across multiple semantically heterogeneous, distributed data sources. Each user can choose relevant data sources from a list of data sources that have been previously registered with INDUS and specify a user perspective (by selecting a user schema and user ontology from a list of available options or defining new ones if needed). The user can map between user perspective and data sources by choosing from existing mappings (or defining new mappings).
- Storage and further manipulation of results of queries. The results returned by a user query can be temporarily stored in a local rela-

tional database. This in effect, represents a materialized relational view (modulo the mappings between user and data source specific ontologies) across distributed, heterogeneous (and not necessarily relational) data repositories.

In summary, INDUS offers the basic functionality necessary to flexibly integrate information from multiple heterogeneous data sources and structure the results according to a user-supplied ontology.

1.3.1 Related Work on Data Integration

Hull [42], Davidson et al. [25], Eckman [32], Calvanese and De Giacomo [17], Doan and Halevy [27], Halevy et al. [38] survey alternative approaches to data integration. These include multi-database systems [62, 8, 14], and mediator based approaches [70, 34, 24, 2, 45, 50, 48, 31]. Tomasic et al. [67] proposed an approach to scaling up access to heterogeneous data sources. Haas et al. [37, 36] investigated optimization of queries across heterogeneous data sources. Rodriguez-Martinez and Roussoloulos [61] proposed a code shipping approach to design an extensible middleware system for distributed data sources. Lambrecht et al. [46] proposed a planning framework for gathering information from distributed sources. Lenzerini [47], Dou et al. [30], Cali et al. [16], Calvanese et al. [17, 18] have developed logic-based approaches to data integration. Bonatti et al. [12] proposed ontology-extended relational algebra for integrating relational data sources. These efforts addressed, and to varying degrees, solved the following problems in data integration: design of query languages and rules for decomposing queries into sub queries and composing the answers to sub queries into answers to the initial query through schema integration. Maluf and Wiederhold [52] proposed an ontology algebra for merging of ontologies. Others have explored approaches to mapping between schema [15, 51, 59] and discovering or learning mappings [26, 28]. The design of INDUS [60, 20, 23, 22] was necessitated by the lack of publicly available open source data integration platforms that could be used as a basis for learning classifiers from semantically heterogeneous distributed data. The INDUS approach to data integration draws on logic-based approaches to ontology-based schema integration and ontology-based relational algebra for bridging gaps in data semantics. To the best of our knowledge, INDUS is one of the few systems that support bridging of semantic gaps in both schema and data semantics.

1.4 Summary

The research summarized in this paper has led to:

(a) The development of a general theoretical framework for learning predictive models (e.g., classifiers) from large, physically distributed data sources where it is neither desirable nor feasible to gather all of the data in a centralized location for analysis [21]. This framework offers a general recipe for the design of algorithms for learning from distributed data that are provably exact with respect to their centralized counterparts (in the sense that the model constructed from a collection of physically distributed data sets is provably identical to that obtained in the setting where the learning algorithm has access to the entire data set). A key feature of this framework is the clear separation of concerns between hypothesis construction and extraction and refinement of sufficient statistics needed by the learning algorithm from data which reduces the problem of learning from data to a problem of decomposing a query for sufficient statistics across multiple data sources and combining the answers returned by the data sources to obtain the answer for the original query. This work has resulted in the identification of sufficient statistics for a large family of learning algorithms including in particular, algorithms for learning decision trees [21], neural networks, support vector machines and Bayesian networks, and consequently, provably exact algorithms for learning the corresponding classifiers from distributed data [23].

(b) The development of theoretically sound yet practical variants of a large class of algorithms [21, 23] for learning predictive models (classifiers) from distributed data sources under a variety of assumptions (motivated by practical applications) concerning the nature of data fragmentation, and the query capabilities and operations permitted by the data sources (e.g., execution of user supplied procedures), and precise characterization of the complexity (computation, memory, and communication requirements) of the resulting algorithms relative to their centralized counterparts.

(c) The development of a theoretically sound approach to formulation and execution of statistical queries across semantically heterogeneous data sources [20]. This work has demonstrated how to use semantic correspondences and mappings specified by users from a set of terms and relationships among terms (user ontology) to terms and relations in data source specific ontologies to construct a sound procedure for answering queries for sufficient statistics needed for learning classifiers from semantically heterogeneous data. An important component of this work has to do with the development of statistically sound approaches to learning classifiers from *partially specified data* resulting from data described at different levels of abstraction across different data sources [74, 75, 76].

(d) The development of INDUS, a modular, extensible, open-source software

toolkit[1] for data-driven knowledge acquisition from large, distributed, autonomous, semantically heterogeneous data sources [22, 20].

(e) Applications of the resulting approaches to computational biology applications involving exploration of protein sequence-structure-function relationships [22]. Examples include: construction of classifiers for assigning proteins to functional families [69, 1] and for sequence-based prediction of protein-protein [71, 72], protein-DNA [73], and protein-RNA [65, 64] interfaces.

Work in progress is aimed at

(a) Design, implementation, and evaluation of scalable algorithms with provable performance guarantees (in terms of accuracy of results, bandwidth and computational efforts), relative to their centralized counterparts, for learning predictive models from distributed, semantically heterogeneous, alternately structured data, including in particular, multi-relational data, sequence data, network data (e.g., 3-dimensional molecular structures, social networks, macromolecular interaction networks), multi-modal data (e.g., text, images) sources under a variety of constraints on the operations supported by the data sources (queries for data, constraints on the types of queries allowed, queries for statistics, execution of user-supplied code at the data source).

(b) Systematic experimental analysis of the resulting algorithms on both real-world and synthetic datasets as a function of the characteristics of data sources (complexity of data source schema, ontologies, and mappings; data source query and processing capabilities, size of the data sets, prevalence of partially missing attribute values as a consequence of integration of data described at multiple levels of granularity), errors or inconsistencies in semantic interoperation constraints and mappings; characteristics of the algorithms (e.g., types of statistics needed for learning), and performance criteria (quality of results produced relative to the centralized counterparts, computational resource, bandwidth, and storage usage); and different sets of data source access, processing, bandwidth constraints captured by alternative cost models.

(c) Investigation of ontology and inter-ontology mapping languages, including in particular, distributed and modular ontology languages, including distributed discription logics [13], E-connections [56], and package-based description logics [4] to support selective integration of ontologies in open environments (e.g., the world-wide web).

The resulting algorithms and software for information integration and distributed data mining will not only advance the state of the art in machine

[1] `http://www.cild.iastate.edu/software/indus.html`

learning but also extend the range of applications of machine learning in emerging data-rich domains e.g., bioinformatics, security informatics, materials informatics, and social informatics. These technical advances, together with a distributed testbed for experimenting with the resulting algorithms, will contribute to the development of critical elements of the cyberinfrastructure for e-science [3, 40, 41].

1.5 Acknowledgments

This research has been supported in part by grants from the National Science Foundation (NSF IIS 0219699, NSF IIS 0639230, and NSF IIS 0711356). Several current and former members of the Iowa State University Artificial Intelligence Research Laboratory have contributed to the work summarized in this chapter. We would like to acknowledge in particular, the contributions of Jie Bao (modular ontologies, INDUS implementation), Cornelia Caragea (bioinformatics applications and relational learning), Drena Dobbs (bioinformatics applications), Neeraj Koul (INDUS implementation), Jyotishman Pathak (INDUS implementation), Jaime Reinoso-Castillo (an early prototype of the data integration component of INDUS), Adrian Silvescu (learning from distributed and partially specified data), Gioea Slutzki (modular ontologies), George Voutsadakis (modular ontologies), and Jun Zhang (learning from partially specified data).

References

[1] C. Andorf, D. Dobbs, and V. Honavar. Exploring inconsistencies in genome-wide function annotations. *BMC Bioinformatics*, 8:284, 2006.

[2] Y. Arens, C. Chin, C. Hsu, and C. Knoblock. Retrieving and integrating data from multiple information sources. *International Journal on Intelligent and Cooperative Information Systems*, 2(2):127–158, 1993.

[3] D. E. Atkins, K. K. Droegemeier, S. I. Feldman, H. Garcia-Molina, and P. Klein, M. L. Messina. Revolutionizing science and engineering through cyberinfrastructure. In *Report of the National Science Foundation Blue-Ribbon Advisory Panel on Cyberinfrastructure*, Washington, DC: National Science Foundation, 2003.

[4] G. Bao, J.and Slutzki and V. Honavar. A semantic importing approach to knowledge reuse from multiple ontologies. In *AAAI 2007*, pages 1304–1309, 2007.

[5] J. Bao, D. Caragea, , and V. Honavar. Modular ontologies - a formal investigation of semantics and expressivity. In *Proceedings of the First Asian Semantic Web Conference*, volume 4185, pages 616–631, Beijing, China, 2006. Springer-Verlag, Lecture Notes in Computer Science. Best paper award.

[6] J. Bao, D. Caragea, and V. Honavar. On the semantics of linking and importing in modular ontologies. In *Proceedings of the International Semantic Web Conference*, volume 4273, Athens, Georgia, USA, 2006. Springer-Verlag Lecture Notes in Computer Science.

[7] J. Bao, D. Caragea, and V. Honavar. A tableau-based federated reasoning algorithm for modular ontologies. In *Proceedings of the ACM/IEEE/WIC Conference on Web Intelligence*, pages 404–410, Hong Kong, 2006.

[8] T. Barsalou and D. Gangopadhyay. M(dm): An open framework for interoperation of multimodel multidatabase systems. *IEEE Data Engineering*, 1992.

[9] T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American*, May 2001.

[10] R. Bhatnagar and S. Srinivasan. Pattern discovery in distributed databases. In *Proceedings of the Fourteenth AAAI Conference*, pages 503–508, Providence, RI, 1997. AAAI Press/The MIT Press.

[11] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag, 2006.

[12] 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.

[13] A. Borgida and L. Serafini. Distributed description logics: Directed domain correspondences in federated information sources, 2002.

[14] M.W. Bright, A.R. Hurson, and S.H. Pakzad. A taxonomy and current issues in multibatabase systems. *Computer Journal*, 25(3):5–60, 1992.

[15] A. Cali, D. Calvanese, G. De Giacomo, and M. Lenzerini. Accessing data integration systems through conceptual schemas. In *SEBD 2002*, pages 161–168, 2002.

[16] A. Cali, D. Calvanese, G. De Giacomo, and M Lenzerini. Data integration under integrity constraints. *Information Systems*, 29(2):147–163, 2004.

[17] D. Calvanese and D. De Giacomo. Data integration: A logic-based perspective. *AI Magazine*, 26:59–70, 2005.

[18] D. Calvanese, G. De Giacomo, M. Lenzerini, and R. Rosati. Logical foundations of peer-to-peer data integration. In *PODS 2004*, pages 241–251, 2004.

[19] D. Caragea. *Learning classifiers from Distributed, Semantically Heterogeneous, Autonomous Data Sources*. Ph.d. thesis, Department of Computer Science. Iowa State University, Ames, Iowa, USA, 2004.

[20] D. Caragea, J. Pathak, and V. Honavar. Learning classifiers from semantically heterogeneous data. In *Proceedings of the International Conference on Ontologies, Databases, and Applications of Semantics for Large Scale Information Systems*, pages 963–980, 2004.

[21] D. Caragea, A. Silvescu, and V. Honavar. A framework for learning from distributed data using sufficient statistics and its application to learning decision trees. *International Journal of Hybrid Intelligent Systems*, 1(2):80–89, 2004.

[22] D. Caragea, A. Silvescu, J. Pathak, J. Bao, C. Andorf, D. Dobbs, and V. Honavar. Information integration and knowledge acquisition from semantically heterogeneous biological data sources. In *Proceedings of the Second International Workshop on Data Integration in Life Sciences,*

(DILS 2005), San Diego, CA, 2005. Berlin: Springer-Verlag. Lecture Notes in Computer Science.

[23] D. Caragea, J. Zhang, J. Bao, J. Pathak, , and V. Honavar. Algorithms and software for collaborative discov-ery from autonomous, semantically heterogeneous, distributed information sources. In *Proceedings of the Conference on Algorithmic Learning Theory*, volume 3734, pages 13–44, Berlin, 2005. LNCS, Springer-Verlag.

[24] C. K. Chang and H. Garcia-Molina. Mind your vocabulary: query mapping across heterogeneous information sources. In *ACM SIGMOD International Conference On Management of Data*, Philadelphia, PA, June 1999.

[25] S. Davidson, J. Crabtree, B. Brunk, J. Schug, V. Tannen, G. Overton, and C. Stoeckert. K2/Kleisli and GUS: experiments in integrated access to genomic data sources. *IBM Journal*, 40(2), 2001.

[26] R. Dhamankar, Y. Lee, A. Doan, A. Halevy, and P. Domingos. imap: Discovering complex mappings between database schemas. In *SIGMOD Conference 2004*, pages 383–394, 2004.

[27] 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.

[28] A. Doan, J. Madhavan, P. Domingos, and A. Halevy. Ontology matching: A machine learning approach. In S. Staab and R. Studer, editors, *Handbook on Ontologies in Information Systems*, pages 397–416. Springer-Velag, 2004. Invited paper.

[29] P. Domingos. Knowledge acquisition from examples via multiple models. In *Proceedings of the Fourteenth International Conference on Machine Learning*, pages 98–106, Nashville, TN, 1997. Morgan Kaufmann.

[30] D. Dou, D.and McDermott and P. Qi. Ontology translation on the semantic web. *Journal of Data Semantics*, 2:35–57, 2005.

[31] D. Draper, A. Y. Halevy, and D. S. Weld. The nimble XML data integration system. In *ICDE*, pages 155–160, 2001.

[32] B. Eckman. A practitioner's guide to data management and data integration in bioinformatics. *Bioinformatics*, pages 3–74, 2003.

[33] R. A. Fisher. On the mathematical foundations of theoretical statistics. *Philosophical Transactions of the Royal Society*, 222:309–368, 1922.

[34] H. Garcia-Molina, Y. Papakonstantinou, D. Quass, A. Rajaraman, Y. Sagiv, J. Ullman, V. Vassalos, and J. Widom. The TSIMMIS approach to mediation: data models and languages. *Journal of Intelligent Information Systems, Special Issue on Next Generation Information Technologies and Systems*, 8(2), 1997.

[35] L.R. Grossman and Y. Guo. Parallel methods for scaling data mining algorithms to large data sets. In J.M. Zytkow, editor, *Handbook on Data Mining and Knowledge Discovery*. Oxford University Press, 2001.

[36] L. Haas, M. Hernndez, H. Ho, L. Popa, and M. Roth. Clio grows up: from research prototype to industrial tool. In *Proceedings of SIGMOD Conference 2005*, pages 805–810, 2005.

[37] L.M. Haas, D. Kossmann, E. Wimmers, and J. Yan. Optimizing queries across diverse sources. In *Proceedings of the 23rd VLDB Conference*, pages 267–285, Athens, Greece, 1997.

[38] A. Halevy, A. Rajaraman, , and J. Ordille. Data integration: The teenage years. In *Proceedings of Very Large Databases, 2006*, 2006.

[39] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning : Data Mining, Inference, and Prediction*. Springer-Verlag, 2001.

[40] J. Hendler and D. De Roure. E-science: the grid and the semantic web. *IEEE Intelligent Systems*, 19(1):65–71, 2004.

[41] T. Hey and A. E. Trefethen. Cyberinfrastructure for e-Science. *Science*, 308(5723):817–821, 2005.

[42] R. Hull. Managing semantic heterogeneity in databases: A theoretical perspective. In *PODS*, pages 51–61, Tucson, Arizona, 1997.

[43] H. Kargupta, B.H. Park, D. Hershberger, and E. Johnson. Collective data mining: A new perspective toward distributed data mining. In H. Kargupta and P. Chan, editors, *Advances in Distributed and Parallel Knowledge Discovery*. MIT Press, 1999.

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

[45] C.A. Knoblock, S. Minton, J.L. Ambite, N. Ashish, I. Muslea, A. Philpot, and S. Tejada. The ariadne approach to Web-based information integration. *International Journal of Cooperative Information Systems*, 10(1-2):145–169, 2001.

[46] E. Lambrecht, S. Kambhampati, and S. Gnanaprakasam. Optimizing recursive information-gathering plans. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 1204–1211. AAAI Press, 1999.

[47] M. Lenzerini. Data integration: A theoretical perspective. In *PODS 2002*, pages 233–246, 2002.

[48] A. Levy. The information manifold approach to data integration. *IEEE Intelligent Systems*, 13, 1998.

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

[50] J. Lu, G. Moerkotte, J. Schue, and V.S. Subrahmanian. Efficient maintenance of materialized mediated views. In *Proceedings of 1995 ACM SIGMOD Conference on Management of Data*, San Jose, CA, 1995.

[51] J. Madhavan, P. Bernstein, P. Domingos, and A. Halevy. Representing and reasoning about mappings between domain models. In *AAAI/IAAI 2002*, pages 80–86, 2002.

[52] D. Maluf and G. Wiederhold. Abstraction of representation in interoperation. *Lecture Notes in AI*, 1315, 1997.

[53] J. Mansour. Learning boolean functions via the fourier transform. In *Theoretical Advances in Neural Computation and Learning*. Kluwer, 1994.

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

[55] B. Park and H. Kargupta. Constructing simpler decision trees from ensemble models using Fourier analysis. In *Proceedings of the 7th Workshop on Research Issues in Data Mining and Knowledge Discovery (DMKD'2002)*, Madison, WI, 2002.

[56] Bijan Parsia and Bernardo Cuenca Grau. Generalized link properties for expressive e-connections of description logics. In *Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI-2005)*, 2005.

[57] A.L. Prodromidis, P. Chan, and S.J. Stolfo. Meta-learning in distributed data mining systems: issues and approaches. In H. Kargupta and P. Chan, editors, *Advances of Distributed Data Mining*. AAAI Press, 2000.

[58] Foster J. Provost and Venkateswarlu Kolluri. A survey of methods for scaling up inductive algorithms. *Data Mining and Knowledge Discovery*, 3(2):131–169, 1999.

[59] E. Rahm and P. Bernstein. A survey of approaches to automatic schema matching. *VLDB J.*, 10(4):334–350, 2001.

[60] J. Reinoso-Castillo, A. Silvescu, D. Caragea, J. Pathak, and V. Honavar. Information extraction and integration from heterogeneous, distributed, autonomous information sources: a federated, query-centric approach. In *IEEE International Conference on Information Integration and Reuse*, Las Vegas, Nevada, November 2003.

[61] M. Rodriguez-Martinez and R. Roussopoulos. MOCHA: a self-extensible database middleware system for distributed data sources. In *Proceedings*

of the 2000 ACM SIGMOD International Conference on Management of Data, pages 213–224, Dallas, TX, 2000.

[62] A. Sheth and J. Larson. Federated databases: architectures and issues. *ACM Computing Surveys*, 22(3):183–236, 1990.

[63] A. Srivastava, E. Han, V. Kumar, and V. Singh. Parallel formulations of decision-tree classification algorithms. *Data Mining and Knowledge Discovery*, 3(3):237–261, 1999.

[64] M. Terribilini, Lee. J-H., C. Yan, S. Carpenter, R. Jernigan, V. Honavar, and D. Dobbs. Identifying interaction sites in recalcitrant proteins: predicted protein and rna binding sites in hiv-1 and eiav agree with experimental data. In *Pacific Symposium on Biocomputing*, volume 11, pages 415–426, Hawaii, 2006. World Scientific.

[65] M. Terribilini, J.H. Lee, C. Yan, R. Jernigan, V. Honavar, and D. Dobbs. Computational prediction of protein-rna interfaces. *RNA Journal*, 12(1450):1462, 2006.

[66] S. Thrun, C. Faloutsos, M. Mitchell, and L. Wasserman. Automated learning and discovery: State-of-the-art and research topics in a rapidly growing field. *AI Magazine*, 1999.

[67] A. Tomasic, L. Rashid, and P. Valduriez. Scaling heterogeneous databases and design of DISCO. *IEEE Transactions on Knowledge and Data Engineering*, 10(5):808–823, 1998.

[68] J. Vaidya and C. Clifton. Privacy-preserving data mining: Why, how, and when. *IEEE Security & Privacy*, 2(6):19–27, 2004.

[69] X. Wang, D. Schroeder, D. Dobbs, and V. Honavar. Automated data-driven discovery of motif-based protein function classifiers. *Information Sciences*, 155:1–18, 2003.

[70] G. Wiederhold and M. Genesereth. The conceptual basis for mediation services. *IEEE Expert*, 12:38–47, 1997.

[71] C. Yan, D. Dobbs, and V. Honavar. A two-stage classifier for identification of protein-protein interface residues. *Bioinformatics*, 20:371–378, 2004.

[72] C. Yan, V. Honavar, and D. Dobbs. Identifying protein-protein interaction sites from surface residues - a support vector machine approach. *Neural Computing Applications*, 13:123–129, 2004.

[73] C. Yan, M Terribilini, F. Wu, R.L. Jernigan, D. Dobbs, and V. Honavar. Identifying amino acid residues involved in protein-dna interactions from sequence. *BMC Bioinformatics*, 2006.

[74] J. Zhang and V. Honavar. Learning decision tree classifiers from attribute-value taxonomies and partially specified data. In T. Fawcett

and N. Mishra, editors, *Proceedings of the International Conference on Machine Learning*, pages 880–887, Washington, DC, 2003.

[75] J. Zhang and V. Honavar. AVT-NBL: An algorithm for learning compact and accurate naive bayes classifiers from attribute value taxonomies and data. In *Proceedings of the Fourth IEEE International Conference on Data Mining*, pages 289–298, Brighton, UK, 2004. IEEE Press.

[76] J. Zhang, A. Silvescu, D-K. Kang, and V. Honavar. Learning compact and accurate naive bayes clas-sifiers from attribute value taxonomies and partially specified data. *Knowledge and Information Systems*, 9:157–179, 2006.