Learning from Semantically Heterogeneous Data

Doina Caragea*
Department of Computing and Information Sciences
Kansas State University
234 Nichols Hall
Manhattan, KS 66506
USA
voice: +1 785-532-7908
fax: +1 785-532-7353
email: dcaragea@ksu.edu

Vasant Honavar
Department of Computer Science
Iowa State University
226 Atanasoff Hall
Ames, IA 50011
USA
voice: +1 515-294-0258
email: honavar@cs.iastate.edu

(* Corresponding author)

Learning from Semantically Heterogeneous Data

Doina Caragea, Kansas State University, USA

Vasant Honavar, Iowa State University, USA

ABSTRACT

Advances in the Semantic Web technologies present unprecedented opportunities for exploiting multiple related data sources to discover useful knowledge in many application domains. We have precisely formulated the problem of learning classifiers from a collection of several related ontology extended data sources, which make *explicit* (the typically implicit) ontologies associated with the data sources of interest, and have presented a solution to this problem. User-specific mappings between a user ontology and data source ontologies are used to answer statistical queries that provide the sufficient statistics needed for learning classifiers from semantically heterogeneous data.

INTRODUCTION

Recent advances in sensors, digital storage, computing and communications technologies 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, e-science, e-government, 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 heterogeneous sources.

The Semantic Web enterprise (Berners-Lee, Hendler, & Lassila, 2001) is aimed at making the contents of the Web machine interpretable, so that heterogeneous data sources can be used together. 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.

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 (Halevy, Rajaraman, & Ordille, 2006; Noy, 2004; Doan, & Halevy, 2005; Dou, & LePendu, 2006), as well as techniques for learning ontologies (Cimiano, 2006; Cimiano, Blohm, & Stemle, 2007) and discovering semantic correspondences between ontologies to assist in this process (Kalfoglou, & Schorlemmer, 2005; Noy, & Stuckenschmidt, 2005; Do, & Rahm, 2007). Such advances in Semantic Web technologies present unprecedented opportunities for exploiting multiple related data sources, each annotated with its own metadata, in discovering useful knowledge in many application domains.

However, there has been relatively little work on machine learning approaches to knowledge acquisition from data sources annotated with metadata that exposes the structure (schema) and semantics (ontology). The purpose of this chapter is to precisely define the

problem of learning classifiers from semantically heterogeneous data sources and summarize recent advances that have led to a solution to this problem.

BACKGROUND

Eckman (2003), Calvanese, & De Giacomo (2005), Doan, & Halevy (2005) and Halevy, Rajaraman, & Ordille (2006) survey alternative approaches to data integration, including multi-database systems, mediator based approaches, etc. Such 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 subqueries into answers to the initial query through schema integration. However, neither of the existing data integration systems currently support learning from semantically heterogeneous distributed data without first assembling a single data set. While it is possible to retrieve the *data* necessary for learning from a set of heterogeneous data sources, store the retrieved data in a local database, and then apply standard (centralized) learning algorithms, such approach is not feasible when the amounts of data involved are large, and bandwidth and memory are limited, or when the query capabilities of the data sources are limited to providing statistical summaries (e.g., counts of instances that satisfy certain constraints on the values of their attributes) as opposed to data instances.

While there has been significant work on applying machine learning to problems such as ontology construction, information extraction from text and discovery of mappings between ontologies (Kushmerick, Ciravegna, Doan, Knoblock, & Staab, 2005), there has been not much work on learning classifiers from semantically heterogeneous data sources. Therefore, solutions to the problem of learning from semantically heterogeneous data sources are greatly needed.

MAIN FOCUS

The problem addressed is best illustrated by an example. Consider two academic departments that independently collect information about their students (**Figure 1**). 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 *Student ID*, *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

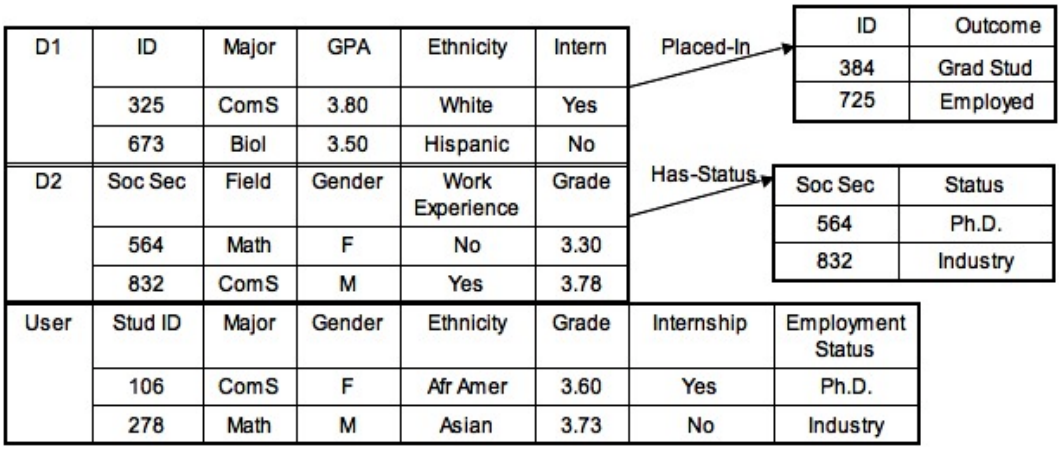

Figure 1: Student data collected by two departments from a statistician's perspective.

representative attributes are *Student ID*, *Major, Gender*, *Ethnicity*, *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**) 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 must establish the correspondences between the user attributes and the data source attributes.

In our framework each data source has associated with it a data source description (i.e., the schema and ontology of the data source). We call the resulting data sources, *ontology extended data sources* (OEDS). An OEDS 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 (Caragea, Zhang, Bao, Pathak, & Honavar, 2005). The formal semantics of OEDS are based on ontology-extended relational algebra (Bonatti, Deng, & Subrahmanian, 2003).

A *data set D* is an instantiation $\mathcal{I}(S)$ of a schema. The *ontology O* of an OEDS $\mathcal{D}$ 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 into an *is-a* hierarchy.

Because 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, our framework allows each user or a community of users to select the ontological commitments that they deem useful in a specific context. A *user's view of data sources* $\mathcal{D}_1, \mathcal{D}_2 \ldots \mathcal{D}_n$ *is specified* by user schema S_U, user ontology O_U, together with a set of semantic *correspondence constraints IC*, and the associated set of *mappings* from the user schema S_U to the data source schemas

$S_1,...,S_n$ and from user ontology O_U to the data source ontologies $O_1,\ldots,O_n$ (Caragea, Zhang, Bao, Pathak, & Honavar, 2005). We consider the following types of semantic correspondence constraints: $x \leq y$ (x is semantically subsumed by y), $x \geq y$ (x semantically subsumes y), $x = y$ (x is semantically equivalent to y), $x \neq y$ (x is semantically incompatible with y), $x \approx y$ (x is semantically compatible with y).

Figure 2 shows examples of ontologies that take the form of *is-a* hierarchies over attribute values. **Figure 3** shows some simple examples of user-specified semantic correspondence constraints between the user perspective and the data sources $\mathcal{D}_1$ and $\mathcal{D}_2$ (respectively).

Let $O_1,...,O_n$ be a set of ontologies associated with the data sources $D_1,...,D_n$, respectively, and

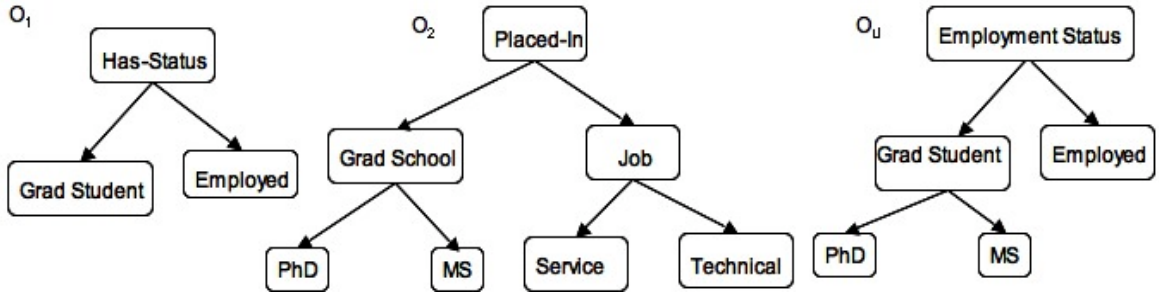

Figure 2: Attribute value taxonomies (ontologies) O_1 and O_2 associated with the attributes *Has-Status* and *Placed-In* in two data sources of interest. O_U is the ontology for *Employment Status* from the user's perspective.

$P_U=(O_U,IC)$ a user perspective with respect to these ontologies. We say that the ontologies $O_1,...,O_n$, are *integrable* according to the user ontology O_U in the presence of semantic correspondence constraints IC if there exist n partial injective mappings $\Psi(O_U, O_1),\ldots,\Psi(O_U, O_n)$ from $O_1,...,O_n$, respectively, to O_U.

$O_1 \rightarrow O_U$	$O_2 \rightarrow O_U$
ID:O_1=Stud ID:O_U	SocSec:O_1=Stud ID:O_U
Major:O_1=Major:O_U	Field:O_1=Major:O_U
GPA:O_1=Grade:O_U	Grade:O_1=Grade:O_U
Ethnicity:O_1=Ethnicity :O_U	
	Gender:O_2=Gender:O_U
Ethnicity:O_1=Ethnicity :O_U	
Intern :O_1=Internship:O_U	Work-Experience:O_2=Internship:O_U
Placed-In :O_1=Employment-Status:O_U	Has-Status: O_2=Employment-Status:O_U

Figure 3: An example of user-specified semantic correspondences between the user ontology O_U and data source ontologies O_1 and O_2 (from **Figure 2**).

Problem Definition

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, a support vector machine, etc.). The data D typically consists of a 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 the case of semantically heterogeneous data sources, we assume the existence of:

1. A collection of several related OEDSs $\mathcal{D}_1 = \{ D_1,S_1,O_1\}$, $\mathcal{D}_2 = \{ D_2,S_2,O_2\},\ldots,\mathcal{D}_n = \{D_n,S_n,O_n\}$ for which schemas and ontologies are made explicit, and instances in the data sources are labeled according to some criterion of interest to a user (e.g., employment status).
2. A user view, consisting of a user ontology O_U and a set of mappings $\{\Psi_k\}$ that relate the user ontology O_U to the data source ontologies $O_1,\ldots,O_p$. The user view implicitly specifies a user level of abstraction, corresponding to the leaf nodes of the hierarchies in O_U. The mappings $\{\Psi_k\}$ 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 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 OEDSs 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 a criterion P, via the mappings $\{\Psi_k\}$.

We say that an algorithm L_S for learning from OEDSs $\mathcal{D}_1, \mathcal{D}_2 \ldots \mathcal{D}_n$, via the mappings $\{\Psi_k\}$ is *exact* relative to its centralized counterpart L_C, if the hypothesis produced by L_S (federated approach) is identical to that obtained by L_C from the data warehouse D constructed by integrating the data sources $\mathcal{D}_1, \mathcal{D}_2 \ldots \mathcal{D}_n$, according to the user view, via the same mappings $\{\Psi_k\}$ (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 $\mathcal{D}_1, \mathcal{D}_2 \ldots \mathcal{D}_n$ 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).

Partially Specified Data

Consider the data source ontologies O_1 and O_2 and the user ontology O_U shown in **Figure 2**. The attribute *Has-Status* in data source D_2 is specified in greater detail (lower level of abstraction) than the corresponding attribute *Placed-In* 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. We say that the *Status* of students in data source D_1 are only *partially specified* (Zhang, Kang, Silvescu, & Honavar, 2005) with respect to the ontology O_U.

In such cases, answering statistical queries from semantically heterogeneous data sources requires the user to supply not only the mapping between the user 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.

Sufficient Statistics Based Solution

Our approach to the problem of learning classifiers from OEDSs 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).

In the case of learning classifiers from semantically disparate OEDSs, 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 OEDSs. 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 4**).

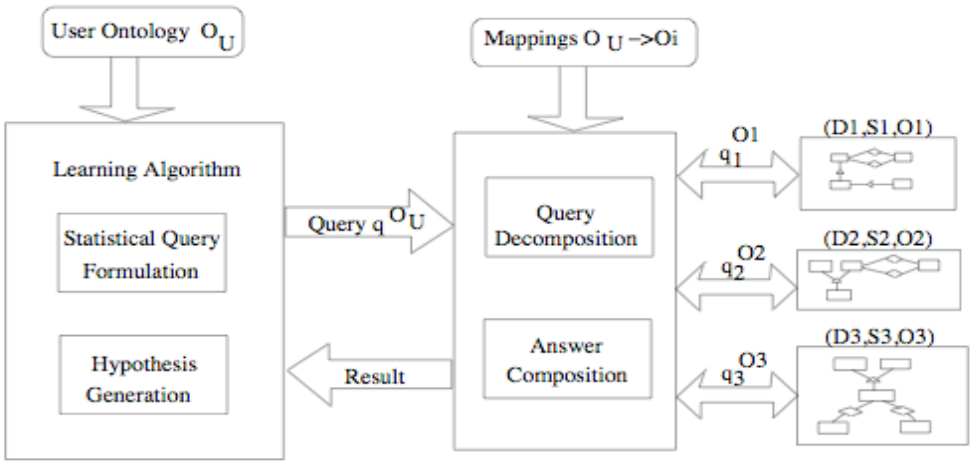

Figure 4: **Learning classifiers from OEDSs**

The resulting algorithms for learning from OEDSs 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 (e.g., naïve Bayes, decision trees, etc.).

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 OEDSs. 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.

FUTURE TRENDS

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.

CONCLUSION

In this chapter, we have precisely formulated the problem of learning classifiers from a collection of several related OEDSs, which make *explicit* (the typically implicit) ontologies associated with the data sources of interest. We have shown how to exploit data sources annotated with relevant metadata in building predictive models (e.g., classifiers) from several related OEDSs, 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. 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 OEDSs.

REFERENCES

1. Berners-Lee, T., Hendler, J., & Lassila, O. (2001). The semantic web. *Scientific American*.
2. Bonatti, P., Deng, Y., & Subrahmanian, V. (2003). An ontology-extended relational algebra. In: *Proceedings of the IEEE Conference on Information Integration and Reuse*, 192–199.

3. Calvanese, D. and De Giacomo, D. (2005). Data integration: A logic-based perspective, AI Magazine, 26(1):59-70.
4. Caragea, D., Zhang, J., Bao, J., Pathak, J., & Honavar, V. (2005). Algorithms and software for collaborative discovery from autonomous, semantically heterogeneous information sources. In: *Proceedings of the 16th International Conference on Algorithmic Learning Theory*, vol. 3734.
5. Cimiano, P. (2006). Ontology Learning and Population from Text: Algorithms, Evaluation and Applications. Springer-Verlag.
6. Cimiano, P., Blohm, S., & Stemle, E. (2007). Harvesting Relations from the Web -- Quantifying the Impact of Filtering Functions. In *Proceedings of the 22nd International Conference on Artificial Conference (AAAI'07)*.
7. Do, H.H., & Rahm, E. (2007). Matching large schemas: Approaches and evaluation, Information Systems, v.32 n.6, p.857-885.
8. Doan, A., & Halevy, A. (2005). Semantic Integration Research in the Database Community: A Brief Survey, *AI Magazine, Special Issue on Semantic Integration*.
9. Dou, D. and LePendu, P. (2006). Ontology-based integration for relational databases. SAC 2006: 461-466.
10. Eckman, B. (2003). A Practitioner's guide to data management and data integration in Bioinformatics. In: Bioinformatics, Lacroix, Z., and Crithlow, T. (Ed). Palo Alto, CA: Morgan Kaufmann. 2003. pp. 35-74.
11. Halevy, A., Rajaraman, A., & Ordille, J. (2006). Data Integration: The Teenage Years. In proceedings of VLDB, 2006.

12. Kalfoglou, Y., & Schorlemmer, M. (2005). Ontology mapping: The state of the art. In *Dagstuhl Seminar Proceedings: Semantic Interoperability and Integration*, Dagstuhl, Germany.
13. M. Kearns. (1998). Efficient noise-tolerant learning from statistical queries. *Journal of the ACM*, 45(6):983–1006.
14. Kushmerick, N., Ciravegna, F., Doan, A., Knoblock, C., & Staab, S. (2005). *Proc. Dagstuhl Seminar on Machine Learning for the Semantic Web*, 2005.
15. Noy, N. (2004). Semantic Integration: A Survey Of Ontology-Based Approaches. *SIGMOD Record, Special Issue on Semantic Integration*, 33(4).
16. Noy, N., & Stuckenschmidt, H. (2005). Ontology Alignment: An annotated Bibliography. In: *Semantic Interoperability and Integration.* Y. Kalfoglou, M. Schorlemmer, A. Sheth, S. Staab, and M. Uschold, editors,.
17. Zhang, J., Kang, D-K., Silvescu, A., & Honavar, V. (2005). Learning compact and accurate naive Bayes classifiers from attribute value taxonomies and data. *Knowledge and Information Systems*.

KEY TERMS AND THEIR DEFINITIONS

Ontology: Assumptions concerning the objects that exist in the world, the properties or attributes of the objects, relationships between objects, the possible values of attributes, and their intended meaning, as well as the granularity or level of abstraction at which objects and their properties are described.

Structure Ontology: Defines the semantics of a data source schema (entities and attributes of entities that appear in data source schema *S*).

Content Ontology: Defines the semantics of the data source instances (values and relationships between values that the attributes can take in instantiations of the schema).

Ontology-Extended Data Sources: An OEDS consists of a data set (representing the instantiation of a schema), a data source schema and a data source ontology.

User View: A user view of a set of OEDS is specified by a user schema, a user ontology and a set of mapping from the user schema to the data sources schemas, and from the user ontology to the data source ontologies.

Classification Task: A task for which the learner is given experience in the form of labeled examples and learns to classify new unlabeled examples. In a classification task, the output of the learning algorithm is called hypothesis or classifier (e.g., a decision tree, a support vector machine, etc.)

S**ufficient Statistics:** A statistic is called a sufficient statistic for a parameter if the statistic captures all the information about the parameter, contained in the data. A statistic is called a sufficient statistic for learning a hypothesis using a particular learning algorithm applied to a given data set, if there exists an algorithm that takes as input the statistic and outputs the desired hypothesis.

Learning Task Decomposition: A learning algorithm can be decomposed in two components: a *statistics gathering* component that formulates and sends queries to a data source and a *hypothesis refinement* component that uses the result of the query to modify a partially constructed algorithm output (and further invokes the information extraction component if needed to generate the final algorithm output).

Learning from Semantically Heterogeneous Data: Given a set of related OEDS, a user view (schema, ontology and mappings), a hypothesis class and a performance criterion, the task of the

learner is to output a hypothesis that optimizes the performance criterion, via the user mappings.

Exact Learning: We say that an algorithm for learning from OEDS via a set of mappings is *exact* relative to its centralized counterpart, if the hypothesis it produces is identical to that produced by the centralized learning from the data warehouse constructed by integrating the OEDS, via the same set of mappings.