A Platform for Collaborative and Distributed KDD Process Design

Claudia Diamantini, Domenico Potena and Emanuele Storti
Dipartimento di Ingegneria dell’Informazione
Università Politecnica delle Marche
Via Brecce Bianche, 60131 Ancona
{c.diamantini, d.potena, e.storti}@univpm.it

POSTER PAPER

ABSTRACT

Knowledge Discovery in Databases (KDD) is a complex and computationally intensive process aimed at extracting knowledge from large amounts of data. To provide effective support to users, especially non-experts, in this work we propose a knowledge-centric platform specifically aimed at supporting collaborative design of KDD processes in distributed environments.

KEYWORDS: Collaborative technologies; KDD Process; Knowledge Discovery in Databases; Process Design

1. INTRODUCTION

The rapid growth of databases in last years asks organizations to deal with issues related to the management of large amounts of data, which represent a valuable resource for decision making. Although technologies for data management/storage are widely available, much effort is still needed to provide users with systems for effectively analyzing and understanding data. We use the term Knowledge Discovery in Databases (KDD) to refer to the non-trivial process of extracting interesting, valid and useful patterns from data. As well as scientific experimentation in e-Science, it is a complex and computationally intensive process aimed at gaining knowledge from a huge set of data. Often performed in distributed settings, KDD projects usually involve a deep interaction among heterogeneous tools and several users with specific expertise. Given the high complexity of the process, such users need effective support to achieve their goal of knowledge extraction.

Various systems have been proposed for KDD processes design and management (refer to [1] for a survey). However, most of such proposals are mostly concerned with large-scale and high-performance issues, or focusing mainly on Data Mining phase without considering the KDD process as a whole, and still lack advanced tools for effective retrieval of processes and support to process design, such as semi-automatic process composition, versioning management and tools to specify and reuse best practices.

Then, although many supporting environments have been designed for cooperative work [2], only recent ones consider collaboration in KDD, and very few of them with a knowledge representation perspective.

This work presents an overview of the Knowledge Discovery in Databases Virtual Mart (KDDVM), a user- and knowledge-centric framework aimed at supporting the design of KDD processes in a highly distributed and collaborative scenario, in which computational resources and actors dynamically interoperate to share and elaborate knowledge. The contribution of the work is two-fold: firstly, a conceptual systematization of the relevant knowledge is provided, with the aim to formalize, through semantic technologies, each element taking part in the design and execution of a KDD process, including computational resources, data and actors; secondly, we propose an implementation of the framework as an open, modular and extensible Service-Oriented platform (an early version is available in [3]), in which several services are available both to perform basic operations of data manipulations and to support more advanced functionalities.
2. PLATFORM

Knowledge about each resource taking part in a KDD project (computational units, data, actors and process) is fully described from different levels of abstraction by a Knowledge Layer that includes specific languages and descriptors for each of such levels, which are interrelated in order to provide machine-readable mappings among them: an algorithm is an abstract prototype of a tool, whereas a tool is an implementation of an algorithm in a concrete programming language. Finally, a service is a tool running on a server, offering its interface through standard Web Services protocols.

At its conceptual level, the Knowledge Layer includes the formal ontologies KDDONTO and TeamONTO. The former is aimed to describe data and algorithms with their properties: task, methods, performances, inputs and outputs. The latter describes users, their competencies and skills, their projects and organizations, and allows to support team building functionalities.

The exploitation of semantic information is one of the key elements of our knowledge-centric approach, as they provide, at a conceptual level, the needed terminology which the concrete level descriptors can refer to. For such reasons, a service descriptor is semantically annotated through KDDONTO concepts, in order to explain its I/O interface and functionalities.

At concrete level, service descriptors are stored in a Service Repository, while processes are described in an XML-language and managed by a Process Repository. The Knowledge Layer provides the data model on which an open and modular Service-Oriented platform is built, which includes both basic and support services. The former are the above-mentioned heterogeneous and possibly distributed tools for every kind of KDD task (e.g., classification, regression, clustering) that have been wrapped as standard web services, while the latter are services providing all the needed back-end and advanced functionalities for discovery and composition.

Discovery services support the browsing and the retrieval of information concerning basic services, processes and sub-processes, respectively available in Service and Process Repositories. Apart a syntactic search by name, the Service Discovery allows a semantic search based on functionalities or I/O interface, by exploiting semantic annotations available in the descriptors, while Process Discovery, relying on a repository index, is aimed to find processes containing a certain service or holding a given structure. In order to provide effective support, KDDVM includes composition functionalities. In particular, a Matchmaking service assesses whether two I/O interfaces are semantically compatible (a cost function evaluates the ontological similarity). By exploiting such a service, a semi-automatic procedure to generate abstract processes is defined, useful to provide non-expert users with process prototypes that can be used as reference during design.

Besides these functionalities for specific support to KDD, KDDVM offers also collaborative services enabling cooperation, in particular Team building, aimed at setting up a virtual team for a project by exploiting TeamONTO, collaborative design of processes through a Designer (see Figure 1), task assignment and communication systems.

3. CONCLUSION

In this work we introduce a framework and a platform to support the design of KDD processes in distributed and collaborative environments.

What makes KDDVM original with respect to other proposals is the systematic use of semantic information, a loosely coupled and layered architecture, a cooperative approach and its flexibility. While most solutions focus only on Data Mining or on local KDD support systems, our proposal is more general and natively conceived for an open, distributed and collaborative environment.

REFERENCES

