MRE-KDD+: A MULTI-RESOLUTION, ENSEMBLE-BASED MODEL FOR ADVANCED KNOLWEDGE DISCOVERY

Alfredo Cuzzocrea
Department of Electronics, Computer Science, and Systems
University of Calabria, Italy

Keywords: On-Line Analytical Processing, Data Mining, On-Line Analytical Mining, Knowledge Discovery from Large Databases and Data Warehouses, Cooperative Information Systems.

Abstract: In data-intensive scenarios, data repositories expose very different formats, and knowledge representation schemes are very heterogeneous accordingly. As a consequence, a relevant research challenge is how to efficiently integrate, process and mine such distributed knowledge in order to make available it to end-users/applications in an integrated and summarized manner. Starting from these considerations, in this paper we propose an OLAM-based model for advanced knowledge discovery, called M*ulti-Resolution* E*nsemble-based Model for Advanced* K*nowledge* D*iscovery in Large* D*atabases and Data Warehouses* (MRE-KDD$^+$). MRE-KDD$^+$ integrates in a meaningfully manner several theoretical amenities coming from *On-Line Analytical Processing* (OLAP), *Data Mining* (DM) and *Knowledge Discovery in Databases* (KDD), and results to be an effective model for supporting advanced decision-support processes in many fields of real-life data-intensive applications.

1 INTRODUCTION

In data-intensive scenarios, intelligent applications run on top of enormous-in-size, heterogeneous data sources in order to implement advanced decision-support processes. Data sources range from transactional data to XML data, and from workflow-process log-data to sensor network data; here, collected data are typically represented, stored and queried in large databases and data warehouses, which, without any loss of generality, define a collection of *distributed and heterogeneous data sources*, each of them executing as a singleton software component (e.g., DBMS server, DW server, XDBMS server etc). Contrarily to this so-delineated distributed setting, intelligent applications wish to extract *integrated*, *summarized knowledge* from such data sources, in order to make strategic decisions for their business. Nevertheless, heterogeneity of data and platforms, and distribution of architectures and systems represent a serious limitation for the achievement of this goal. As a consequence, research communities have devoted a great deal of attention to this problem, with a wide set of proposals (Fayyad et al., 1996) ranging from *Data Mining* (DM) tools, which concern algorithms for extracting *patterns* and *regularities* from data, to *Knowledge Discovery in Databases* (KDD) techniques, which concern the overall process of discovering useful knowledge from data.

Among the plethora of techniques proposed in literature to overcome the above-highlighted gap between data and knowledge, *On-Line Analytical Mining* (OLAM) (Han, 1997) is a successful solution that integrates *On-Line Analytical Processing* (OLAP) (Gray et al., 1997) with DM in order to provide an integrated methodology for extracting useful knowledge from large databases and data warehouses. The benefits of OLAM have been already put-in-evidence (Han, 1997): (*i*) DM algorithms can execute on integrated, *OLAP-based multidimensional views* that are already pre-processed and cleaned; (*ii*) users/applications can take advantages from the interactive, exploratory nature of OLAP tools to decisively enhance the knowledge fruition experience; (*iii*) users/applications can take advantages from the flexibility of OLAP tools in making available a wide set of DM solutions for a given KDD task, so that, thanks to OLAP, different DM algorithms become easily *interchangeable* in order to decisively enhance the benefits coming from cross-comparative

Cuzzocrea A. (2007).
MRE-KDD+: A MULTI-RESOLUTION, ENSEMBLE-BASED MODEL FOR ADVANCED KNOLWEDGE DISCOVERY.
In *Proceedings of the Ninth International Conference on Enterprise Information Systems - AIDSS*, pages 152-158
DOI: 10.5220/0002404001520158

data analysis methodologies over large amounts of data.

Starting from these considerations, in this paper we propose an OLAM-based framework for advanced knowledge discovery, along with a formal model underlying this framework, called *Multi-Resolution Ensemble-based Model for Advanced Knowledge Discovery in Large Databases and Data Warehouses* ($\mathcal{MRE}$-$\mathcal{KDD}^+$). On the basis of OLAP principles, $\mathcal{MRE}$-$\mathcal{KDD}^+$, which can be reasonably considered as an innovative contribution in this research field, provides a formal, rigorous methodology for implementing advanced KDD processes in data-intensive settings, but with particular regard to two specialized instances represented by (*i*) a general application scenario populated by distributed and heterogeneous data sources, such as a conventional distributed data warehousing environment (e.g., like those that one can find in B2B and B2C *e*-commerce systems), and (*ii*) the integration/data layer of cooperative information systems, where different data sources are integrated in a unique middleware in order to make KDD processes against these data sources transparent-to-the-user.

Besides the widely-accepted benefits coming from integrating OLAM within its core layer (Han, 1997), $\mathcal{MRE}$-$\mathcal{KDD}^+$ allows data-intensive applications adhering to the methodology it defines to take advantages from other relevant characteristics, among which we recall the following: (*i*) the *multi-resolution support* offered by popular OLAP operators/tools (Han & Kamber, 2000), which allow us to execute DM algorithms over integrated and summarized multidimensional views of data at different *level of granularity* and *perspective of analysis*, thus sensitively improving the quality of KDD processes; (*ii*) the *ensemble-based support*, which, briefly, consists in meaningfully combining results coming from different DM algorithms executed over a collection of multidimensional views in order to generate the final knowledge, and provide facilities at the knowledge fruition layer.

The remaining part of this paper is organized as follows. In Section 2, we outline related work. In Section 3, we present in detail $\mathcal{MRE}$-$\mathcal{KDD}^+$. Finally, in Section 4 we outline conclusions of our work, and further activities in this research field.

2 RELATED WORK

OLAM is a powerful technology for supporting knowledge discovery from large databases and data warehouses that mixes together OLAP functionalities for representing/processing data, and DM algorithms for extracting regularities (e.g., patterns, association rules, clusters etc) from data. In doing this, OLAM realizes a proper KDD process.

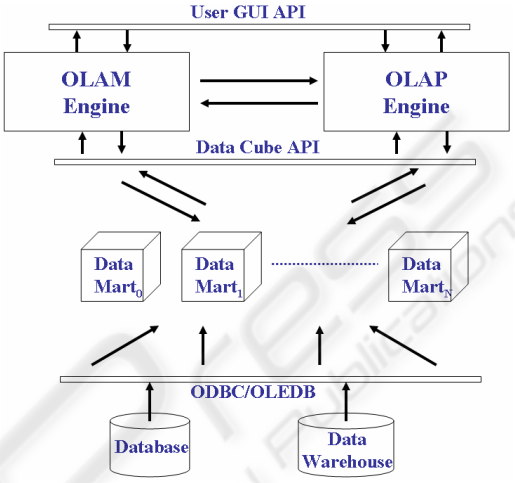

Figure 1: A reference architecture for OLAM.

OLAM was proposed by Han in his fundamental paper (Han, 1997), along with the OLAP-based DM system DBMiner (Han et al., 1996), which can be reasonably considered as the practical implementation of OLAM. In order to emphasize and refine the capability of discovering useful knowledge from huge amounts of data, OLAM gets the best of both technologies (i.e., OLAP and DM). From OLAP, (*i*) the excellent capability of storing data, which has been of relevant interest during the last years (e.g., (Harinarayan et al., 1996)), (*ii*) the support for *multidimensional and multi-resolution data analysis* (Chaudhuri et al., 1997); (*iii*) the richness of OLAP operators (Han & Kamber, 200), such as *roll-up*, *drill-down*, *slice-&-dice*, *pivot* etc; (*iv*) the wide availability of a number of query classes, such as *range-queries* (Ho et al., 1997), which have been extensively studied during the last years, and can be used as baseline for implementing even-complex KDD tasks. From DM, the broad collection of techniques available in literature, each of them oriented to cover a specific KDD task; among these techniques, some are relevant for OLAM, such as: *mining association rules in transactional or relational databases*, *mining classification rules*, *cluster analysis*, *summarizing and generalizing data using data cube* or *attribute-oriented inductive* approaches.

A reference architecture for OLAM is depicted in Figure 1 (Han & Kamber, 2000). Here, the *OLAP Engine* and the *OLAM Engine* run in a combined manner in order to extract useful knowledge from a collection of subject-oriented data marts. Beyond the above-described OLAM features, this architecture also supports a leading OLAM functionality, the so-called *On-Line Interactive Mining* (Han & Kamber, 2000), which consists in iteratively executing DM algorithms over *different* views extracted from the *same* data mart. In this case, the effective "add-on" value given by OLAP is represented by a powerful *information gain* which cannot be easily supported by traditional OLTP operators/tools, without introducing excessive computational overheads.

While there are in literature a plethora of data representation techniques and DM algorithms, each of then developed for a particular application scenario, frameworks that integrate with-a-large-vision several techniques coming from different contexts via synthesizing data warehousing, DM and KDD principles are very few. Furthermore, while there exist an extremely wide set of DM and KDD tools (a comprehensive overview can be found in (Goebel & Gruenwald, 1999)), mainly focused to cover a specific KDD task (e.g., association rule discovery, classification, clustering etc), very few of them integrate heterogeneous KDD-oriented techniques and methodologies in a unique environment. Along these, the most significant experiences that have deeply influenced our work are DBMiner and WEKA (Witten & Frank, 2005). In the following, we refer to both the environments in the vest of "realizations" of the respective underlying models.

DBMiner is a powerful OLAM-inspired system which allows us to (*i*) extract and represent knowledge from large databases and data warehouses, and (*ii*) mine knowledge via a wide set of very useful data analysis functionalities, mainly OLAP-inspired, such as data/patterns/results browse, exploration, visualization and intelligent querying. Specifically, at the representation/storage layer, DBMiner makes use of the popular *data cube* model (the foundation of OLAP), first proposed by Gray et al. (1997), where relational data are aggregated on the basis of a multidimensional and multi-resolution vision of data. Based on the data cube model, DBMiner makes available to the user a wide set of innovative functionalities ranging from *time-series analysis* to *prediction* of the data distribution of relational attributes to mining of complex objects (like those that one can find in a GIS); furthermore, DBMiner also offers a *data mining query language*, called *DMQL*, for supporting the standardization of DM functionalities and their integration with conventional DBMS. Finally, the graphical user interface of DBMiner supports various attracting, user-friendly forms implementing the above-listed features.

WEKA is a *Machine Learning* (ML) environment for efficiently supporting DM activities against large databases, and it has been designed to aid in decision support processes in order to understand which information is relevant for the specific context, and, consequently, make prediction faster. Similarly to DBMiner, WEKA offers a graphical environment where users can (*i*) edit a ML technique, (*ii*) test it against external data sets, and (*iii*) study its performance under the stressing of various metrics. Moreover, WEKA users, just like DBMiner users, are allowed to mine the output knowledge of ML techniques by means of several advanced intelligent visualization components. Contrarily to DBMiner, WEKA does not make use of a particular data-representation/storage solution to improve data access/management/processing.

Finally, due to the nature and the goals of both the outlined environments/models, we can claim that DBMiner is closer to our work rather than WEKA.

3 $\mathcal{MRE}$-$\mathcal{KDD}$+: PRINCIPLES AND THEORETICAL FOUNDATIONS

$\mathcal{MRE}$-$\mathcal{KDD}^{+}$ is the innovative model we propose, and it has been designed to efficiently support advanced knowledge discovery from large databases and data warehouses according to a multi-resolution, ensemble-based approach. Basically, $\mathcal{MRE}$-$\mathcal{KDD}^{+}$ follows the approach (Han, 1997), which, as highlighted in Section 2, is the state-of-the-art for OLAM.

3.1 $\mathcal{MRE}$-$\mathcal{KDD}^{+}$ OLAP-based Data Representation and Management Layer

Let $S = \{S_0, S_1, \ldots, S_{K-1}\}$ be a set of K distributed and heterogeneous data sources, and $\mathcal{D} = \{\mathcal{D}_0, \mathcal{D}_1, \ldots, \mathcal{D}_{P-1}\}$ be a set of P data marts defined over data sources in S. The first component of $\mathcal{MRE}$-$\mathcal{KDD}^{+}$ is the so-called *Multidimensional Mapping Function* MMF, defined as a tuple $\mathrm{MMF} = \langle \mathrm{MMF}^{\mathcal{H}}, \mathrm{MMF}^{\mathcal{F}} \rangle$,

which takes as input a sub-set of M data sources in S, denoted by $S^{\mathcal{M}} = \{S_m, S_{m+1}, \ldots, S_{m+M-1}\}$, and returns as output a data mart $\mathcal{D}_k$ in $\mathcal{D}$, computed over data sources in $S^{\mathcal{M}}$ according to the construct $\mathsf{MMF}^{\mathcal{H}}$ that models the *definition* of $\mathcal{D}_k$. $\mathsf{MMF}^{\mathcal{H}}$ is in turn implemented as a conventional OLAP conceptual schema, such as *star-* or *snowflake-schemas* (Han & Kamber, 2000). $\mathsf{MMF}^{\mathcal{F}}$ is the construct of MMF that properly models the underlying function, defined as follows:

$$\mathsf{MMF}^{\mathcal{F}}: S \rightarrow \mathcal{D} \quad (1)$$

Given a MMF $\mathcal{G}$, we introduce the concept of *degree* of $\mathcal{G}$, denoted by $\mathcal{G}^{\Delta}$, which is defined as the number of data sources in S over which the data mart provided by $\mathcal{G}$ (i.e., $\mathcal{D}_k$) is computed, i.e. $\mathcal{G}^{\Delta} \equiv |S^{\mathcal{M}}|$.

Due to the strongly "data-centric" nature of $\mathcal{MRE}$-$\mathcal{KDD}^{+}$, management of OLAP data assumes a critical role, also with respect to performance issues, which must be taken in relevant consideration in data-intensive applications like those addressed by OLAM. To this end, we introduce the *Multidimensional Cubing Function* MCF, defined as a tuple $\mathsf{MCF} = \langle \mathsf{MCF}^{\mathcal{H}}, \mathsf{MCF}^{\mathcal{F}} \rangle$, which takes as input a data mart $\mathcal{D}_k$ in $\mathcal{D}$, and returns as output a data mart $\mathcal{D}_h$ in $\mathcal{D}$, according to the construct $\mathsf{MCF}^{\mathcal{H}}$ that models an OLAP operator/tool. In more detail, $\mathsf{MCF}^{\mathcal{H}}$ can be one of the following OLAP operators/tools:

- *Multidimensional View Extraction* $\mathcal{V}$, which computes $\mathcal{D}_h$ as a multidimensional view extracted from $\mathcal{D}_k$ by means of a set of ranges $R_0, R_1, \ldots, R_{N-1}$ defined on the N dimensions of $\mathcal{D}_k$ $d_0, d_1, \ldots, d_{N-1}$, respectively, being each range R_j defined as a tuple $R_j = \langle L_l, L_u \rangle$, with $L_l < L_u$, such that L_l is the lower and L_u is the upper bound on d_j, respectively;
- *Range Aggregate Query* $\mathcal{Q}$, which computes $\mathcal{D}_h$ as a one-dimensional view (i.e., an *aggregate value*) given by the application of a SQL aggregate operator (such as SUM, COUNT, AVG etc) applied to the collection of (OLAP) cells contained within a multidimensional view extracted from $\mathcal{D}_k$ by means of the operator $\mathcal{V}$;
- *Top-K Query* $\mathcal{K}$, which computes $\mathcal{D}_h$ as a multidimensional view extracted from $\mathcal{D}_k$ by means of the operator $\mathcal{V}$, and containing the (OLAP) cells of $\mathcal{D}_k$ whose values are the first K greatest values among cells in $\mathcal{D}_k$;
- *Drill-Down* $\mathcal{U}$, which computes $\mathcal{D}_h$ via decreasing the level of detail of data in $\mathcal{D}_k$;
- *Roll-Up* $\mathcal{R}$, which computes $\mathcal{D}_h$ via increasing the level of detail of data in $\mathcal{D}_k$;
- *Pivot* $\mathcal{P}$, which computes $\mathcal{D}_h$ via re-structuring the dimensions of $\mathcal{D}_k$ (e.g., changing the ordering of the dimensions).

Formally, $\mathsf{MCF}^{\mathcal{H}} = \{\mathcal{V}, \mathcal{Q}, \mathcal{K}, \mathcal{U}, \mathcal{R}, \mathcal{P}\}$. Finally, $\mathsf{MCF}^{\mathcal{F}}$ is the construct of MCF that properly models the underlying function, defined as follows:

$$\mathsf{MCF}^{\mathcal{F}}: \mathcal{D} \rightarrow \mathcal{D} \quad (2)$$

It should be noted that the construct $\mathsf{MCF}^{\mathcal{H}}$ of MCF operates on a singleton data mart to extract another data mart. In order to improve the quality of the overall KDD process, we also introduce the *Extended Multidimensional Cubing Function* $\mathsf{MCF}_{\mathsf{E}}$, defined as a tuple $\mathsf{MCF}_{\mathsf{E}} = \langle \mathsf{MCF}_{\mathsf{E}}^{\mathcal{H}}, \mathsf{MCF}^{\mathcal{F}} \rangle$, which extends MCF by providing a different, complex OLAP operator/tool (i.e., $\mathsf{MCF}_{\mathsf{E}}^{\mathcal{H}}$) instead that the "basic" $\mathsf{MCF}^{\mathcal{H}}$. $\mathsf{MCF}_{\mathsf{E}}^{\mathcal{H}}$ supports the amenity of executing $\mathsf{MCF}^{\mathcal{H}}$ over multiple data marts, modeled as a sub-set of B data marts in $\mathcal{D}$, denoted by $\mathcal{D}^{\mathcal{B}} = \{\mathcal{D}_b, \mathcal{D}_{b+1}, \ldots, \mathcal{D}_{b+B-1}\}$, being these data marts combined by means of the operator JOIN performed with respect to schemas of data marts. Specifically, $\mathsf{MCF}_{\mathsf{E}}^{\mathcal{H}}$ operates according to two variants: (*i*) in the first one, we first apply an instance of $\mathsf{MCF}^{\mathcal{H}}$ to each data mart in $\mathcal{D}^{\mathcal{B}}$, thus obtaining a set of *transformed* data marts $\mathcal{D}^{\mathcal{T}}$, and then the operator JOIN to data marts in $\mathcal{D}^{\mathcal{T}}$; (*ii*) in the second one, we first apply the operator JOIN to data marts in $\mathcal{D}^{\mathcal{B}}$, thus obtaining a *unique* data mart $\mathcal{D}^{\mathcal{U}}$, and then an instance of $\mathsf{MCF}^{\mathcal{H}}$ to the data mart $\mathcal{D}^{\mathcal{U}}$.

To give examples, let $\mathcal{D}^{\mathcal{B}} = \{\mathcal{D}_0, \mathcal{D}_1, \mathcal{D}_2\}$ be the target sub-set of data marts, then, according to the first variant, a possible instance of $\mathsf{MCF}_{\mathsf{E}}^{\mathcal{H}}$ could be: $\mathcal{V}(\mathcal{D}_0) \rhd\lhd \mathcal{K}(\mathcal{D}_1) \rhd\lhd \mathcal{U}(\mathcal{D}_2)$; contrarily to this, according to the second variant, a possible instance of $\mathsf{MCF}_{\mathsf{E}}^{\mathcal{H}}$ could be: $\mathcal{U}(\mathcal{D}_0 \rhd\lhd \mathcal{D}_1 \rhd\lhd \mathcal{D}_2)$. Note that, in both cases, the result of the operation is still a data mart belonging to the set of data marts $\mathcal{D}$ of $\mathcal{MRE}$-$\mathcal{KDD}^{+}$.

Formally, we model $\mathsf{MCF}_E^{\mathcal{H}}$ as a tuple $\mathsf{MCF}_E^{\mathcal{H}} = \langle \mathcal{D}^B, \mathcal{Y} \rangle$, such that (*i*) $\mathcal{D}^B$ is the sub-set of data marts in $\mathcal{D}$ on which $\mathsf{MCF}_E^{\mathcal{H}}$ operates to extract the final data mart, and (*ii*) $\mathcal{Y}$ is the set of instances of $\mathsf{MCF}^{\mathcal{H}}$ used to accomplish this goal.

3.2 $\mathcal{MRE}$-$\mathcal{KDD}^+$ Data Mining Layer

DM algorithms defined in $\mathcal{MRE}$-$\mathcal{KDD}^+$ are modeled by the set $\mathcal{A} = \{\mathcal{A}_0, \mathcal{A}_1, \ldots, \mathcal{A}_{T-1}\}$; these are classical DM algorithms focused to cover specific instances of consolidated KDD tasks, such as discovery of patterns and regularities, discovery of association rules, classification, clustering etc, with the novelty of being applied to multidimensional views (or, equally, data marts) extracted from the data mart domain $\mathcal{D}$ of $\mathcal{MRE}$-$\mathcal{KDD}^+$ via complex OLAP operators/tools implemented by the components MCF and MCF_E. Formally, an algorithm $\mathcal{A}_h$ of $\mathcal{A}$ in $\mathcal{MRE}$-$\mathcal{KDD}^+$ is modeled as a tuple $\mathcal{A}_h = \langle \mathcal{I}_h, \mathcal{D}_h, O_h \rangle$, such that: (*i*) $\mathcal{I}_h$ is the instance of $\mathcal{A}_h$ (properly, $\mathcal{A}_h$ models the *class* of the particular DM algorithm), (*ii*) $\mathcal{D}_h$ is the data mart on which $\mathcal{A}_h$ executes to extract knowledge, and (*iii*) O_h is the output knowledge of $\mathcal{A}_h$. Specifically, O_h representation depends on the nature of algorithm $\mathcal{A}_h$, meaning that if, for instance, $\mathcal{A}_h$ is a clustering algorithm, then O_h is represented as a collection of clusters (reasonably, modeled as sets of items) extracted from $\mathcal{D}_h$.

KDD process in $\mathcal{MRE}$-$\mathcal{KDD}^+$ are governed by the component *Execution Scheme*, denoted by ES, which rigorously models *how* algorithms in $\mathcal{A}$ must be executed over multidimensional views of $\mathcal{D}$. To this end, ES establishes (*i*) how to combine multidimensional views and DM algorithms (i.e., which algorithm must be executed on which view), and (*ii*) the temporal sequence of executions of DM algorithms over multidimensional views. To formal model this aspect of the framework, we introduce the *Knowledge Discovery Function* KDF, which takes as input a collection of R algorithms $\mathcal{A}^{\mathcal{R}} = \{\mathcal{A}_r, \mathcal{A}_{r+1}, \ldots, \mathcal{A}_{r+R-1}\}$ and a collection of W data marts $\mathcal{D}^W = \{\mathcal{D}_w, \mathcal{D}_{w+1}, \ldots, \mathcal{D}_{w+W-1}\}$, and returns as output an execution scheme ES_p. KDF is defined as follows:

$$\mathsf{KDF}: \mathcal{A}^{\mathcal{R}} \times \mathcal{D}^W \rightarrow \langle \mathcal{I}^R \times \mathcal{D}^T, \varphi \rangle \quad (3)$$

such that: (*i*) $\mathcal{I}^R$ is a collection of instances of algorithms in $\mathcal{A}^{\mathcal{R}}$, (*ii*) $\mathcal{D}^T$ is a collection of transformed data marts obtained from $\mathcal{D}^W$ by means of cubing operations provided by the components MCF or MCF_E of the framework, and (*iii*) φ is a collection determining the temporal sequence of instances of algorithms in $\mathcal{I}^R$ over data marts in $\mathcal{D}^T$ in terms of ordered pairs $\langle \mathcal{I}_r, \mathcal{D}^T_k \rangle$, such that the ordering of pairs indicates the temporal ordering of executions. From (3), we derive the formal definition of the component ES of $\mathcal{MRE}$-$\mathcal{KDD}^+$ as follows:

$$\mathsf{ES} = \langle \mathcal{I} \times \mathcal{D}, \varphi \rangle \quad (4)$$

Finally, the execution scheme ES_p provided by KDF can be one of the following alternatives:

- *Singleton Execution* $\langle \mathcal{I}_r \times \mathcal{D}^T_k, \varphi \rangle$: execution of the instance $\mathcal{I}_r$ of the algorithm $\mathcal{A}_r$ over the transformed data mart $\mathcal{D}^T_k$, with $\varphi = \{\langle \mathcal{I}_r, \mathcal{D}^T_k \rangle\}$.
- *1 × N Multiple Execution* $\langle \mathcal{I}_r \times \{\mathcal{D}^T_k, \mathcal{D}^T_{k+1}, \ldots, \mathcal{D}^T_{k+N-1}\}, \varphi \rangle$: execution of the instance $\mathcal{I}_r$ of the algorithm $\mathcal{A}_r$ over the collection of transformed data marts $\{\mathcal{D}^T_k, \mathcal{D}^T_{k+1}, \ldots, \mathcal{D}^T_{k+N-1}\}$, with $\varphi = \{\langle \mathcal{I}_r, \mathcal{D}^T_k \rangle, \langle \mathcal{I}_r, \mathcal{D}^T_{k+1} \rangle, \ldots, \langle \mathcal{I}_r, \mathcal{D}^T_{k+N-1} \rangle\}$.
- *N × 1 Multiple Execution* $\langle \{\mathcal{I}_r, \mathcal{I}_{r+1}, \ldots, \mathcal{I}_{r+N-1}\} \times \mathcal{D}^T_k, \varphi \rangle$: execution of the collection of instances $\{\mathcal{I}_r, \mathcal{I}_{r+1}, \ldots, \mathcal{I}_{r+N-1}\}$ of the algorithms $\{\mathcal{A}_r, \mathcal{A}_{r+1}, \ldots, \mathcal{A}_{r+N-1}\}$ over the transformed data mart $\mathcal{D}^T_k$, with $\varphi = \{\langle \mathcal{I}_r, \mathcal{D}^T_k \rangle, \langle \mathcal{I}_{r+1}, \mathcal{D}^T_k \rangle, \ldots, \langle \mathcal{I}_{r+N-1}, \mathcal{D}^T_k \rangle\}$.
- *N × M Multiple Execution* $\langle \{\mathcal{I}_r, \mathcal{I}_{r+1}, \ldots, \mathcal{I}_{r+N-1}\} \times \{\mathcal{D}^T_k, \mathcal{D}^T_{k+1}, \ldots, \mathcal{D}^T_{k+M-1}\}, \varphi \rangle$: execution of the collection of instances $\{\mathcal{I}_r, \mathcal{I}_{r+1}, \ldots, \mathcal{I}_{r+N-1}\}$ of the algorithms $\{\mathcal{A}_r, \mathcal{A}_{r+1}, \ldots, \mathcal{A}_{r+N-1}\}$ over the collection of transformed data marts $\{\mathcal{D}^T_k, \mathcal{D}^T_{k+1}, \ldots, \mathcal{D}^T_{k+M-1}\}$, with $\varphi = \{\ldots, \langle \mathcal{I}_{r+p}, \mathcal{D}^T_{k+q} \rangle, \ldots\}$, such that $0 \leq p \leq N-1$ and $0 \leq q \leq M-1$.

3.3 $\mathcal{MRE}$-$\mathcal{KDD}^+$ Ensemble Layer

As stated in Section 1, at the output layer, $\mathcal{MRE}$-$\mathcal{KDD}^+$ adopts an ensemble-based approach. The so-called *Mining Results* (MR) coming from the executions of DM algorithms over collections of

data marts must be finally merged in order to provide the end-user/application with the extracted knowledge. It should be noted that this is a relevant task in our proposed framework, as very often end-users/applications are interested in extracting useful knowledge by means of *correlated*, *cross-comparative* KDD tasks, rather than a singleton KDD task, according to real-life DM scenarios. Combining results coming from different DM algorithms is a non-trivial research issue, as recognized in literature. In fact, as highlighted in Section 4.2, the output of a DM algorithm depends on the nature of that algorithm, so that in some cases MR coming from very different algorithms cannot be combined directly.

In $\mathcal{MRE}$-$\mathcal{KDD}^{+}$, we face-off this problematic issue by making use of OLAP technology again. We build multidimensional views over MR provided by execution schemes of KDF, thus giving support to a *unifying manner* of exploring and analyzing final results. It should be noted that this approach is well-motivated under noticing that usually end-user/applications are interested in analyzing final results based on a certain *mining metrics* provided by KDD processes (e.g., *confidence interval of association rules*, *density of clusters*, *recall of IR-style tasks* etc), and this way-to-do is perfectly suitable to be implemented within OLAP data cubes where (*i*) data source is the output of DM algorithms (e.g., item sets), (*ii*) (OLAP) dimensions are user-selected features of the output of DM algorithms, and (*iii*) (OLAP) measures are the above-mentioned mining metrics. Furthermore, this approach also involves in the benefit of efficiently supporting the *visualization* of final results by mean of attracting user-friendly, graphical formats/tools such as multidimensional bars, charts, plots etc, similarly to the functionalities supported by DBMiner and WEKA.

Multidimensional Ensembling Function MEF is the component of $\mathcal{MRE}$-$\mathcal{KDD}^{+}$ which is in charge of supporting the above-described knowledge presentation/delivery task. It takes as input a collection of Q output results $O = \{O_0, O_1, \ldots, O_{Q-1}\}$ provided by KDF-formatted execution schemes and the definition of a data mart $\mathcal{Z}$, and returns as output a data mart $\mathcal{L}$, which we name as *Knowledge Visualization Data Mart* (KVDM), built over data in O according to $\mathcal{Z}$. Formally, MEF is defined as follows:

$$\mathsf{MEF}: \langle O, \mathcal{Z} \rangle \rightarrow \mathcal{D} \qquad (5)$$

It is a matter to note that the KVDM $\mathcal{L}$ becomes part of the set of data marts $\mathcal{D}$ of $\mathcal{MRE}$-$\mathcal{KDD}^{+}$, but, contrarily to the previous data marts, which are used to knowledge processing purposes, it is used to knowledge exploration/visualization purposes.

4 CONCLUSIONS AND FUTURE WORK

Starting from successful OLAM technologies, in this paper we have presented $\mathcal{MRE}$-$\mathcal{KDD}^{+}$, a model for supporting advanced knowledge discovery from large databases and data warehouses, which is useful for any data-intensive setting.

Future work is oriented along two main directions: (*i*) testing the performance of $\mathcal{MRE}$-$\mathcal{KDD}^{+}$ against real-life scenarios such as those drawn by distributed corporate data warehousing environments in B2B and B2C *e*-commerce systems, and (*ii*) extending the actual capabilities of $\mathcal{MRE}$-$\mathcal{KDD}^{+}$ as to embed novel functionalities for supporting prediction of events in new DM activities edited by users/applications on the basis of the "history" given by logs of previous KDD processes implemented in similar or correlated application scenarios.

REFERENCES

Chaudhuri, S., and Dayal, U., 1997. An Overview of Data Warehousing and OLAP Technology. In *SIGMOD Record*, Vol. 26, No. 1, pp. 65-74.

Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P, 1996. From Data Mining to Knowledge Discovery: An Overview. In *Fayyad, U., Piatetsky-Shapiro, G., Smyth, P, and Uthurusamy, R. (eds.), "Advances in Knowledge Discovery and Data Mining"*, AAAI/MIT Press, Menlo Park, CA, USA, pp. 1-35.

Gray, J., Chaudhuri, S., Bosworth, A., Layman, A., Reichart, D., Venkatrao, M., Pellow, F., and Pirahesh, H., 1997. Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tabs, and Sub-Totals. In *Data Mining and Knowledge Discovery*, Vol. 1, No. 1, pp.29-54.

Goebel, M., and Gruenwald L., 1999. A Survey of Data Mining and Knowledge Discovery Software Tools. In *SIGKDD Explorations*, Vol. 1, No. 1, pp. 0-33.

Han, J., 1997. OLAP Mining: An Integration of OLAP with Data Mining. In *Proc. of the 7th IFIP 2.6 DS Work. Conf.*, pp. 1-9.

Han, J., Fu, Y., Wang, W., Chiang, J., Gong, W., Koperski, K., Li, D., Lu, Y., Rajan, A., Stefanovic, N.,

Xia, B., and Zaiane, O.R., 1996. DBMiner: A System for Mining Knowledge in Large Relational Databases. In *Proc. of the 1996 KDD Int. Conf.*, pp. 250-255.

Han, J., and Kamber, M., 2000. *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, San Francisco, CA, USA.

Harinarayan, V., Rajaraman, A., and Ullman, J., 1996. Implementing Data Cubes Efficiently. In *Proc. of the 1996 ACM SIGMOD Int. Conf.*, pp. 205-216.

Ho, C.-T., Agrawal, R., Megiddo, N., and Srikant, R., 1997. Range Queries in OLAP Data Cubes. In *Proc. of the 1997 ACM SIGMOD Int. Conf.*, pp. 73-88.

Witten, I., and Frank, E., 2005. *Data Mining: Practical Machine Learning Tools and Techniques, 2nd ed.*, Morgan Kaufmann Publishers, San Francisco, CA, USA.