Dynamic Pattern Discovery using Multi-Agent Technology

GEORGE RZEVSKI, PETER SKOBELEV, IGOR MINAKOV, SEMEN VOLMAN
Magenta Corporation Ltd,
Gainsborough House, 59 - 60 Thames Street, Windsor Sl4 1TX
UK

Abstract: - A novel method for dynamic pattern discovery based on multi-agent technology has been developed by authors and tested in a number of applications. The key advantages of the new method are: (a) the discovery process is dynamic and adaptive, i.e., it re-clusters data in real time whenever a new data element arrives; (b) the method is capable of finding all clusters without a need for the user to start the process with a hypothesis; (c) records pro-actively search for suitable clusters; (d) users can prescribe what types of clusters they prefer by adjusting microeconomics of the method. The method is particularly suitable for the discovery of patterns of behaviours of website visitors as they browse.

Key-Words: - Data Mining, Pattern Discovery, Multi-Agent Systems, Data Clustering, Distributed Intelligence

1 Introduction

The problem of discovering useful patterns in data is well understood for the cases when all data that is to be analysed is known in advance. However the main limitation of all current data mining algorithms is that the user has to make a hypothesis that certain patterns exist before the procedure can even start.

There are however many important situations where data is arriving for analysis in small batches at frequent, unpredictable intervals. Perhaps the most interesting example is an Internet portal with a large number of visitors who leave behind a small but significant amount of data whenever they visit the site. To extract a coherent and up-to-date pattern of behaviour of customers, it is essential to ensure that the mining process is dynamic, that is, capable of taking into account data as it arrives. Current pattern discovery and data mining algorithms cannot cope with these conditions.

A new method for the dynamic pattern discovery is described in this paper. It uses ontology-based multi-agent systems [1] and it works autonomously. There is no need for users to start the data mining process by proposing hypotheses. The method is protected [2].

The most important concepts involved in this method are defined below.

2 Fundamental Concepts

Data Clusters are groups of data elements (records) with common features, for example, records of all customers purchasing bread and milk on a daily basis. Such data cluster represents a pattern of behaviour of customers and can be used for managing customer relations.

Data Clustering is a process by which data elements are grouped into *Clusters* according to a set of given Clustering Criteria.

Dynamic Data Clustering is a process where the data set that is being clustered changes during the clustering process in an unpredictable manner.

Clustering Propensity is the capability of Record Agents to pro-actively search for clustering partners. To achieve clustering propensity each data element is assigned a certain *Energy Level*.

The Energy Level of a record or a cluster, measured in terms of agreed energy units (eu), determines the ability of a record or a cluster to search for the optimal clustering option and thus its ability to impact the process of selforganisation (changing clustering configurations). The concept of Energy Level is also used to limit the time required to accomplish an acceptable clustering solution. Energy levels could be distributed to records equally, or by some domain–dependent rules. For example, in e-commerce applications, the energy level available to a record can be set as a commission for each item sold. Then the record of a sale of a batch of goods would be richer than a record of a sale of one item. In general, data elements that are more important for the users are given higher energy levels. It works as follows. Data elements spend their energy by paying for joining/forming/leaving clusters. Important data elements, that is, those with higher energy can enter into a greater number of clustering negotiations with a view to attaining the optimal clustering membership. In contrast, data elements of lesser importance will be limited to a smaller number of searches. Clusters attracting more important data elements accumulate large energy levels and are therefore more visible to the users.

Clustering Criteria help agents to decide how to group records together.

Cluster Valuation Formulae specify how exactly a cluster value is to be assessed by Record Agents. In general, the cluster value depends on the number of data elements belonging to the cluster, the energy levels of data members, the shape (or boundaries) of the cluster, the number of attributes of the cluster, their variety etc. A simple and effective way of determining a cluster value is to equate it to the density of the cluster. If we represent data elements belonging to a cluster in an N-dimensional space, where N is the number of attributes in the dataset, we can define cluster density as the number of data elements within the cluster volume.

Record Valuation Formulae specify how exactly a record value is to be assessed by Cluster Agents. If cluster density is to be maximised, then data elements that increase the density will be preferred.

System Value is the value of the overall clustering process. The guiding principle for the allocation of data elements to clusters is to maximise the System Value.

3 The New Clustering Method

Let us assume that a multi-agent system is given a task of allocating records to clusters and that those records arrive to the system in small batches. Times of arrival of records and their features are unpredictable.

Then, the dynamic data clustering method is as follows: (1) An agent is allocated to a new record as it arrives to the system; (2) The new Record Agent considers available clusters, selects those that appear attractive (as determined by a Cluster Valuation Formula) and sends to the appropriate Cluster Agents applications for membership; (3) Cluster Agents, which receive membership applications, evaluate the applicants using a Record Valuation Formula. Those Cluster Agents that decide that the new applicant will increase the energy level of their clusters send membership offers to the applicants; (4) The Record Agent accepts the most appropriate offer and joins a cluster; (5) If no suitable cluster is available, the Record Agent attempts to form a new cluster with other records, which may or may not belong to clusters, by sending cluster formation proposals to their agents; (6) The Record Agents, to which formation of new clusters is proposed, consider the offer. They accept the offer only if it increases the overall value of the system. By accepting the offer agents effectively reorganise the whole system – the previously established relationships between the released records and their clusters are destroyed and new relationships between different records are established increasing the overall value of the system (the process known as Selforganisation); (7) Agents representing newly created clusters and/or clusters whose properties (value, boundaries, number of records) have changed during the selforganisation, start a new negotiation round with agents of selected records, repeating the process described above; (8) The clustering process continues until all records are linked to clusters and no change of cluster membership can increase the value of the system, or until the time available for clustering is exhausted; (9) Under conditions of perpetual arrival of new data elements to the system, at some point in the clustering process agents will begin dropping out-of-date data elements from further clustering considerations (the process known as Evolution).

The clustering resembles a crystallization processes – records create clusters (structures) and these structures, in turn, participate in the formation of more complex structures. The process stops when the whole domain is clusterised (crystallized). The outcome of the process is the creation of high-level structures.

4 An Example

We have four records (data elements), which arrive to the system one by one (fig. 1). They are Record 1 (2,4), Record 2 (3,3), Record 3 (6,3) and Record 4 (7,3). The cluster valuation formula is based on the density of clusters and the negotiation rule is "first consider the nearest data element or cluster".

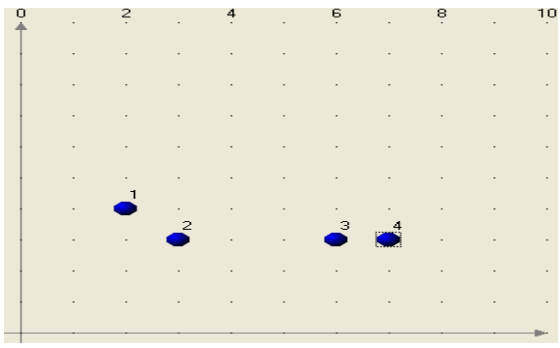

Fig. 1 Data elements available for clustering

Then the clustering steps will be as follows:

(a) Record 1 arrives to the system; (b) Record 2 arrives to the system. It forms with Record 1 a new cluster, Cluster 5. (Fig. 2)

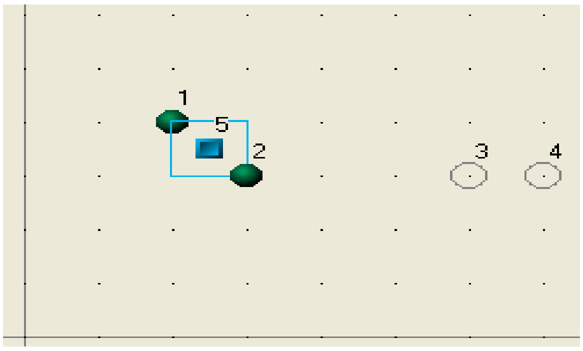

Fig. 2 Records 1 and 2 form cluster 5

(c) Record 3 arrives to the system. It applies to Cluster 5 for membership but it is rejected because its membership would reduce the cluster density. Record 3 then suggests to Cluster 5 to form a new cluster, which would include Record 3 and Cluster 5. They agree and form Cluster 6 (fig. 3).

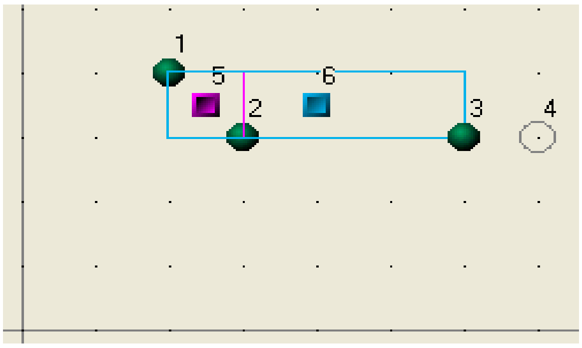

Fig. 3 Record 3 and cluster 5 form cluster 6

(d) Record 4 arrives to the system. Record 4 suggests to Record 3 to leave Cluster 6 and join Record 4 in a new cluster. Record 3 agrees because the new cluster would have a greater density than Cluster 6. Cluster 6 is destroyed and Cluster 7 is created from records 3 and 4 (fig. 4).

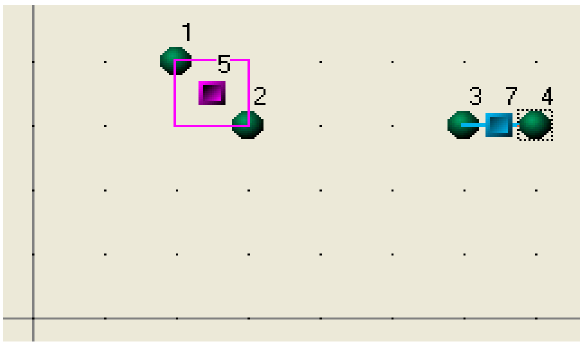

Fig. 4 Cluster 6 is destroyed; cluster 7 is formed

(e) Cluster 7 then proposes to Cluster 5 to form together a new cluster. They form Cluster 8 (fig. 5).

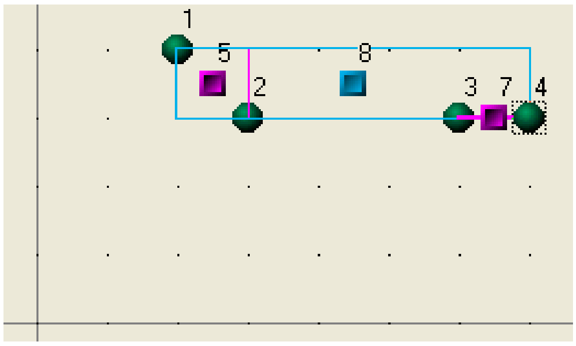

Fig. 5 Cluster 5 and 7 form cluster 8

(d) Cluster 8 realises that there are no further clustering opportunities available because all records and clusters have achieved their preferred memberships and clustering process terminates.

5 Microeconomics of Clustering

Agents representing data elements and clusters negotiate cluster memberships according to one of several available models.

5.1 The Club Model

Data elements pay membership fees to join clusters. Fees are fixed.

5.2 The Shareholder Model

Data elements purchase shares in clusters. Share prices depend on the number of data elements belonging to a cluster and their energy levels and may vary in time. Data elements have opportunities to increase their energy levels by entering or quitting a cluster at an opportune time. They can also lose energy if they make a wrong clustering decision. This model increases differentiation between clusters from the point of view of their usefulness to users.

5.3 The Tax Model

Data elements pay a tax during their membership in clusters. This model enables evolutionary changes in the system because data elements are forced to quit when they exhaust their energy levels and leave vacancies for newly arriving data. Differences in energy levels of data elements provide a mechanism for selection. This model encourages data elements as well as clusters to consider their long-term prospects when they make clustering choices. For example, a cluster that does not attract candidates for membership may decide to reduce its membership tax to bring in new members and thus prolong its life.

5.4 How Clustering Depends on the Model

Data clustering and clustering results depend on the selected model of cluster membership. In particular, the following cluster features are dependent on the clustering criteria: (a) the size of clusters – a large number of small clusters or a smaller number of large clusters; (b) equality (all records are of equal importance) versus elitism (some records are given preferences); and (c) speed of clustering.

Consider the following example (Table 1).

Buyer's name	Goods purchased	Purchase value
John	Beer	500
Bill	Whiskey	13
Paul	Beer	10
Phil	Beer	700
Ralph	Vodka	20

Table 1 Records available for clustering

If the Club Model is used and cluster membership fee is set to 3 energy units (eu), and all records are given equal amount of money, say 10 eu, the system will generate the following two clusters (Table 2).

Cluster Name	Cluster Members	Membership Cost	Cluster Energy Level
A Beer	John Paul Phil	3	9
B Drinks	John Paul Phil Bill Ralph	3	15

Table 2 Two clusters produced by the Club Model

Now let us consider the Shareholder Model. We shall assume that each data element has the amount of eu equal to the purchase value and that the amount of eu for participation in the cluster is calculated in the following way.

First record (John) can create a cluster A, called Beer, for 10% of overall money (50 eu). Now when Phil arrives to the system he can decide whether to pay the same sum of 50 eu or a larger sum, say, 10% of his own overall money (70 eu). In the latter case he will receive a larger number of cluster shares (which could be later sold when the cluster becomes richer and the record decides to leave the cluster and join another).

The average cost of entering the cluster is (50+70)/2 = 60 eu.

It is not profitable for richer clusters to be shareholders of clusters with a small entrance fee because they are poor. Therefore, the following picture emerges.

Cluster Name	Cluster Members	Members Cost	Cluster Energy Level
A Beer	John Phil	60	120
B Drinks	Paul Bill Ralph	2	6

Table 3 Two clusters produced by the Shareholder Model

We see here a dramatic difference in the way the two clusters are formed. The Shareholder Model clearly separates rich records from the poor.

And what is the difference for the client? The first model is useful if users wish to know which customer purchases what kind of beverages and the second model can tell the client who purchases certain goods in big quantities. The two models may help the client to launch two different focused advertising campaigns.

To summarise, The Club Model promotes equality and creates a larger number of clusters in earlier stages (because the membership fee is low and fixed, and it is, therefore, easier to create a new cluster); once in a cluster, data elements are reluctant to change their membership (because the energy level of new records is not sufficient to initiate the re-structuring of clusters).

The Shareholder Model promotes elitism and individualisation; it produces a clear separation of rich and poor data elements, generates a smaller number of clusters, enables a greater mobility of data elements even in the later stages (because new rich records can force the re-structuring of clusters, even ousting poorer records from rich clusters) and sets a higher speed of clustering (because the number of options available to each data element is reduced considerably – the high entrance costs prevent poor data elements to join rich clusters).

The Tax Model gives additional dimension to the clustering process. When the records have to pay money to stay in the system the structure of clusters changes with time as impoverished records are forced to leave. The Tax model is normally used alongside either of the other two models to induce evolution of clusters.

6 Representation of Clusters

A clustering problem domain is usefully represented as a data table, where each row is a record and columns contain field values. Such a table could be regarded as a multi-dimensional space, where each record is an element of this space. In general, the space is heterogeneous, since axes of the space can be of different types (integer, real, etc). Clustering is then a process of detecting dense concentration of elements within the problem domain space. A cluster is defined by a set of axes in the multi-dimensional space and by limits on each of this axes within which all elements belonging to the cluster are located. A data element (record) can belong to several clusters, which means that boundaries of segments of multi-dimensional spaces representing clusters can intersect.

It is often very convenient to represent clusters as rules of the type:

IF (condition A1) and (condition A2) and …. (condition An), THEN (condition B1) and (condition B2) and … (condition Bm).

Where Ai are conditions which include fields over which we have no control (independent fields) and Bi are conditions which include fields whose values is permissible to manipulate. An example of such a rule is: "If an order requires transportation of a cargo of 5 kilograms, then this order should be allocated to the trucks of type Gazel that belong to Trans-GAZ carrier-company".

To produce a rule from the description of a cluster as a segment in a multi-dimensional space, all axes of the space have to be divided into two groups – one containing fields over which we have control and which we can manipulate and the other containing fields which are given (and cannot be changed). Then, dependencies of fields from the first category upon fields from the second category (known as patterns) have to be determined. Clusters constructed over axes which all belong to a single category are ignored. Clustering procedure may omit values along certain axes in which case the cluster is defined over a sub-dimension of the whole space.

Note that a rule derived as described above always represents a cluster and therefore, if a clustering procedure discovered all clusters, it discovered all rules. The inverse statement is not generally true – a cluster cannot be always represented by a rule, e.g., when the cluster has elements all belonging to a single category (all independent or all dependent).

Rules are evaluated using three criteria: (1) Representativeness; (2) Confidence level; (3) Completeness

Representativeness shows how many elements of a cluster are included into a rule. This parameter does not depend upon the patterns.

Confidence level shows how many elements of a cluster that are included into the left part of the rule (IF part) do not meet conditions of the right part of the rule (THEN part). This parameter depends upon the pattern inherent in the cluster. For example, the pattern "among all ones that are parous, all are women" has a confidence level of 100% while a reverse rule has a low confidence level because "not all women are parous".

Completeness shows how many elements of a cluster tat meet conditions of the right part of the rule (THEN part) do not meet conditions of the left part of the rule (IF part). For example, the rule "if you are a human being, then you are mortal" has high level of representativeness but low level of completeness (since not all mortal beings are human beings).

The higher the value of representativeness and of confidence level, the more valuable the revealed interdependency of cluster elements.

When a rule is derived, we can try to move conditions from the IF part of the rule to the TEN part of the rule. If this operation does not decrease the confidence level of the rule, then the modified rule is more useful. For example, the rule "If an order requires transportation from Krasnoyarsk then destination location is Moscow and the order should be assigned to truck of type ZIL" is more useful than the rule "If an order requires transportation from Krasnoyarsk to Moscow, then this order should be assigned to truck of type ZIL", provided that the confidence level has not been decreased by the operation.

7 Comparison with other Methods

Conventional methods of clustering make use of partitioning algorithms, hierarchical decomposition algorithms and quantizing algorithms [3].

7.1 Partitioning algorithms

These algorithms decompose a dataset of N objects into K clusters. The known algorithms include: FOREL 2, KRAB, KRAB Heuristic, K-means, K-medoid, and CLARANS (Clustering Large Application based on RANdomaized Search) [4].

7.2 Hierarchical decomposition algorithms

These algorithms use iterations to decompose a dataset into smaller sets until given termination conditions are satisfied. Examples include FOREL, FOREL OPT, KRAB Heuristic 2, and BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies). The so called Decision Tree algorithms can be also placed into this group, i.e., C4.5, See5/C5.0, and CART (Classification and Regression Trees). The following algorithms also belong to this group: DBSCAN, and CURE (Clustering Using Representatives) [5].

7.3 Quantizing algorithms
These algorithms quantize the data space into finite number of cells and then run all operations on these cells, eg, STING (STatistical INformation Grid-based method), and CLIQUE [5, 6, 7].

It is more effective to find clusters within sub-spaces of original space for the initial task since multi-dimensional data can include noise or have evenly distributed values for some dimensions. Among all algorithms mentioned above only CLIQUE can detect clusters on sub-spaces; nevertheless, other limitations of this algorithm make it unsuitable for many applications: (a) the algorithm operates in a batch mode; (b) the algorithm subdivides the space into finite number of cells in advance. This approach is flowed, since the quality of the results will depend upon the initial subdivision of space into cells; (c) clusters are not organized into hierarchical groups.

Several software products can find if-then rules on datasets. They are based on mathematical algorithms such as bounded combinatorial search (WizWhy) or decision trees search (See5/C5.0). The description of these methods is available in [3]. However, conventional methods such as those described above search for patterns with IF part consisting of a single condition and only in a batch mode.

8 Examples of Applications

8.1 Transportation Scheduling
A logistics company required a tool capable of generating automatically a schedule that would be similar to the schedule produced manually by an experienced operator. The customer did not define metric criteria to estimate proximity and similarity of schedules and therefore the evaluation of results was done by an expert.

The customer provided a dataset of 920 transportation orders that were scheduled in the past manually.

The first task was the extraction of hidden rules from the dataset provided by the customer. For this purpose a table was created with each row representing an order.

The multi-agent Pattern Discovery Tool found 218 rules. Figure 6 shows how rules were distributed by confidence level.

As the graph shows, more than half of the rules have the confidence level of 100%, which testifies to the effectiveness of the Tool.

Rules discovered by the system were shown to an expert who confirmed most of the rules and agreed that the discovered dependencies are intrinsic characteristics of the problem domain. Moreover, the expert declared that 8% of the discovered rules were not known before the system discovered them and that these rules had a great confidence level value.

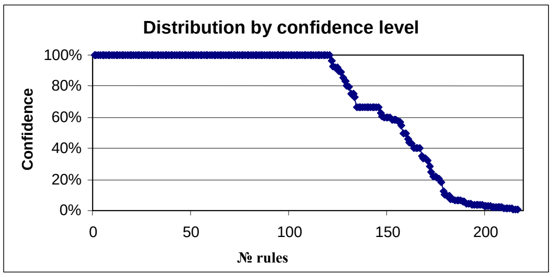

Fig. 6 Distribution of rules by Confidence Level

The discovered rules were then loaded into the knowledge base of a Magenta agent-based Scheduler with a view to performing a comparison between (a) a schedule produced by Magenta Scheduler without extracted rules; (b) a schedule produced by Magenta Scheduler with extracted rules loaded and (c) a schedule produced manually by an experienced operator

Test runs were executed on a dataset different from the one from which rules were extracted and each test run was based on one-week operation. The loading of extracted rules increased the speed of automatic scheduling considerably: 3 hours with rules versus 5 hours without rules. The schedule produced with the help of rules was more like the schedule produced manually by an experienced operator than the schedule produced without rules.

The loading of extracted rules into Magenta Scheduler improved the quality of the resulting schedule, which was quantified as follows: (a) manual rework needed decreased by 32%; (b) journey quality increased by 17%; (c) the presence of gaps in the journeys decreased by 11%; (d) Fleet mileage decreased by 16%; (e) Fleet usage decreased by 8%; (f) Estimated time required for customisation of schedules dramatically decreased from 1-2 months to 10-15 days.

Finding possible options for consolidation in transportation logistics is one of the very useful applications of clustering and clustering analysis. For this purpose appropriate filters are devised for IF and THEN parts of the rules to pass only those fields that are relevant for the problem at hand; geographical coordinates, time windows and Journey-Time Matrix information are normally of particular interest. Each cluster passed through the filter can be considered as a group of orders that are potentially amenable to consolidation.

At the next step it is necessary to define for each cluster the way in which consolidation should be applied – whether all orders in the group are to be shipped by one truck or by several trucks that are similar in characteristics. The decision depends upon the distance between locations, since time is required to load and deliver each cargo, whilst driver's working hours are strictly limited. The decision also depends upon cargo properties as in some cases a special-purpose truck may be required to deliver a cargo, for example, of chilled goods. Additional clustering run on the consolidation clusters helps to bring to light more dependencies and details.

Further information on Magenta Road Transportation Scheduler is available in [8]. Magenta tools for developing multi-agent applications are described in [9].

8.2 Forecasting

A travel agent had statistics on their customers that included personal information, travel history (previously visited country) and the country customers plan to visit next. The last item was often missing because not all customers articulated their plans when visiting the travel agent website. The task was to help the travel agent to forecast future destinations for their customers by discovering patterns in the following dataset.

From available data it was easy to derive rules that predict behaviour of customers such as: IF Age range = "18-30", Previous Destination = "France" THEN the new destination = "Netherlands". The pattern informs the company that it can expect customers from group 3293 to book their next visit to Netherlands.

8.3 Fraud/Error Detection

Fraud/error detection is a process of identifying abnormalities or unusual activities in datasets, in other words, situations that deviate from general trends. It is important and sometimes vital to consider all unusual cases, since they can represent either intentional fraud or a data entry error. For example, if most vehicles in a pool are charged from $10 to $25 for parking while two trucks pay $200, there are at least two explanations for this discrepancy: a computer operator accidentally entered an additional zero, or a person in charge diverted some funds to his account and attempted to cover this fraud with false parking charges. Multi-Agent Pattern Discovery system is capable of identifying cases where the value of an attribute in the THEN part of a rule deviates from the anticipated value. In a resent pattern discovery project a number data entry errors was discovered.

9 Conclusions

The proposed method for dynamic data mining and pattern discovery is unique because it is capable of discovering patterns dynamically, in real time, as data arrive, and is therefore highly suitable for mining visitors' footprints at websites. It represents a dramatic departure from conventional data mining methods which require users to propose hypotheses which are then tested using mathematical methods. The multi-agent approach can discover all hidden patterns in data without human intervention through a process of agent negotiation.

References:

[1] G. Rzevski, P. Skobelev, Agent Method and Computer System for Negotiating in a Virtual Environment. Patent Reference No: WO 03/067432 A1. Published 14.08.2003.

[2] G. Rzevski, P. Skobelev, I Minakov, Data Mining. Patent Reference No: GB 2 411 015 A. Published 17.08.2005.

[3] Glenn W. Milligan, Martha C. Cooper Methodology Review: Clustering Methods, *Applied Psychological Measurement*. Vol. 11, 1987, pp. 329-354

[4] R. Ng, and J. Han, CLARANS: A Method for Clustering Objects for Spatial Data Mining, *IEEE Trans. Knowledge & Data Engineering*, Vol. 14, No. 5, 2002, pp 1003-1016.

[5] Daniel T. Larose, *Discovering Knowledge in data*. Wiley Interscience, 2005.

[6] Rakesh Agrawal, Johannes Gehrke, Dimitrios Gunopulos, Prabhakar Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. *Proceeding of the ACM SIGMOD Conference on Management of Data*, Seattle, WA, 1998, pp. 94-105.

[7] M. Ester, H.Kriegel, J. Sander, X. Xu. A Density-Based Algorithm for Discovering Clustering in Large Spatial Database with Noise. *Proceeding of 2nd International Conference on KDD, 1996.*

[8] J. Himoff, G. Rzevski , P. Skobelev, Multi-Agent Logistics i-Scheduler for Road Transportation, *AAMAS'06*, May 8-12, 2006, Hakodate, Hokkaido, Japan.

[9] G. Rzevski, P. Skobelev, V. Andreev, *MagentaToolkit: A Set of Multi-Agent Tools for Developing Adaptive Real-Time Multi-Agent Applications*. Magenta Whitepaper, 2006.