Agent Based Frequent Set Meta Mining: Introducing EMADS

Kamal Ali Albashiri, Frans Coenen, and Paul Leng
Department of Computer Science, The University of Liverpool, UK

IFIP 2008,Milano, Italy

Outline

- Data Mining Process
- Multi-Agent Systems For Data Mining
- EMADS Vision
- Meta ARM Application
    - Experimentation and Results
- Conclusion and Current Work

Data Mining Process

The DM process can be summarized as follows:

1. **Decide what you want to learn.**
   - Learn a classification model so as to *predict* some feature(s) of new example data cases *(Data Classification).*
   - Find patterns (associations) within data *(Association Rule Mining or ARM).*
   - Group data into clusters and then be able to add new example data cases to the appropriate cluster according to the defining features of the identified clusters and the new case (*Data Clustering).*
2. **Select and prepare your data.**
   - Select relevant data for the desired objective. For instance training and test sets.
   - Consolidate the data into a single *data warehouse* to which mining algorithms can be applied.
   - (Usually) some form of data *transform* (normalisation, descretisation, etc), possibly data *cleaning and so on*.

Data Mining Process cont'

3. Choose and configure the mining task or tasks.

For instance, a user may wish to *cluster users* together that visited similar Web pages, and then *derive association rules* that show how those users and pages are related.

4. Select and configure the mining algorithms.

Many data-mining algorithms are available for a given task. Algorithms *differ* not only in *the potential accuracy* of their end-product, but also in the *computational resources* they require.

5. Build the data-mining model.

The *output* from executing a data-mining task*.* The model can be viewed as an abstraction of the data suited with respect to the objective. The model might be *a neural network, a decision tree, or even a set of rules* understandable by humans.

6. Test and refine the models.

In some cases *several models may be created*, evaluated and an enhanced (in some sense) result distilled.

7. Report findings or predict future outcomes.

Finally, report the findings and/or use the generated data-mining models (for example to predict future example cases).

MultiAgent Systems For Data Mining

- **Process automation**. The current trend is towards automating as much of the data mining process as possible. "*Even those not expert in data mining can reap the benefits of data-mining technologies*," .
- **Task matching**. Agents can automatically match algorithms to a desired data-mining objective; for instance, *a clustering algorithm* to create *data clusters*, or *an association-rules algorithm* to identify *association rules*.
- **Data and Task matching**. Agents can automatically match mining tasks to relevant data.
- **Result evaluation**. Agents can evaluate the accuracy of each model with respect to existing (training) data, and select a "best" (fit for purpose) model.
- **Privacy of sources**. Individual owners of agents can preserve the privacy and security of their raw data and only share the results of data mining activities.
- **Extendibility**. Flexibility to incorporate new data mining techniques and data sources (adding/removing agents)
- **Computational efficiency**. Enhance the efficiency of parallel data mining processing.
- **Scalability and Resublity**. MAS will offer access to a wider pool of available data and allow reuse to previously generated information.

EMADS (Extendible Multi-Agent Data mining System) Vision

EMADS can (or will be able to) provide the following:

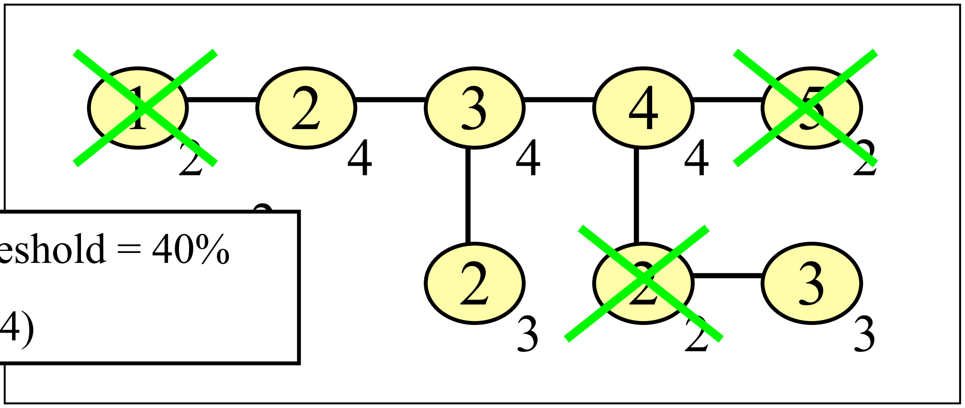
- Enable and accelerate the deployment of practical solutions to data mining problems.
- Provide a data mining agent space into which data mining agents of all sorts can be launched.
- Allow anybody who is prepared to share their data to make that data available using an appropriately defined *data* agent.
- Allow anybody who wishes to obtain data mining results the facility to launch an appropriately define *query* agent.
- Allow anybody the facility to add a new data mining agent that can apply some data mining algorithm to data without significant specialised knowledge of multi-agent based data mining.

EMADS Meta ARM Application

To illustrate some of the features of EMADS a Meta ARM scenario is considered in this paper.

ARM Problem Definition

- Given a database D we wish to find all the frequent itemsets (F) and then use this knowledge to produce high confidence association rules (ARs) of the form A→B.
- Note: Finding F is the most <u>computationally expensive</u> part, once we have the frequent sets generating ARs is straight forward



Support threshold = 40%

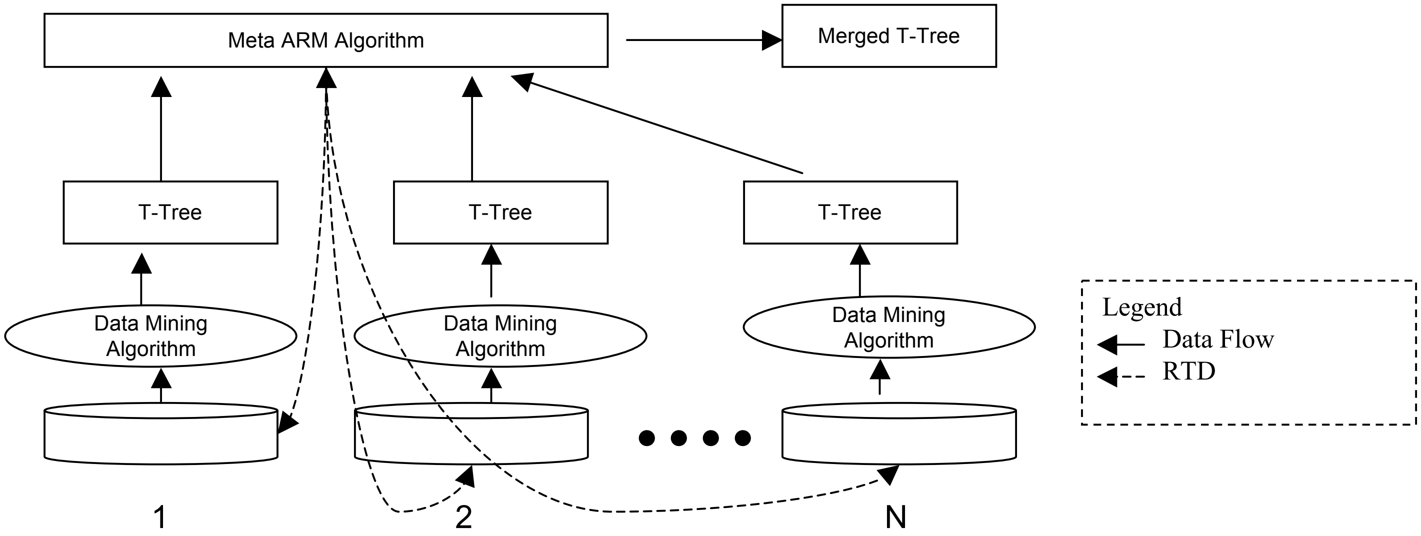
(count of 2.4)

Apriori Example (using a T-tree)

TID	Atts.
1	1 2
2	1 2 3
3	2 3 4
4	2 3 4
5	3 4 5
6	4 5

Frequent Set Meta Mining

- Meta ARM: A given ARM algorithm is applied to N raw data sets producing N collections of frequent item sets.
- The objective is then to merge the different sets of results into a single meta set of frequent itemsets with the aim of generating a set of Association Rules (ARs).
- Key issue: wherever an itemset is frequent in a data source A but not in a data source B a check for any contribution from data source B is required (to obtain a total support count).



Meta ARM approach

Meta ARM algorithms

- Issue in meta ARM is how best to combine the results from N different sources, in the most computationally efficient manner, into a single meta set of itemsets?
- Meta ARM algorithms: We can identify a number of different strategies of combining the individually obtained T-Trees of N datasets into one global T-Tree making use of Return To Data lists (RTD) to obtain additional counts.

1. Brute Force: Merge T-trees one by one generating (N) RTD lists, pruning the T-tree at end of the merge process.
2. Apriori: Merge all T-trees level by level generating (K*N) RTD lists, pruning the T-tree at each level (K = number of levels).
3. Hybrid 1: Merge top level in the Apriori manner and the rest in BF manner.
4. Hybrid 2: Merge top two levels in the Apriori manner and the rest in BF manner.
5. A “bench mark” system is described to allow for appropriate comparison.

Meta ARM algorithms Example

- Brute Force Meta ARM
  - *merge*
  - *inclusion of additional counts (RTD)*
  - *final prune.*

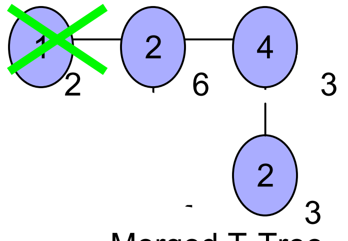
- Apriori Meta ARM

1. *K = 1*
2. *Generate candidate K-itemsets*
3. *Add supports for level K from N T-trees OR add to RTD list*
4. *Get additional support*
5. *Prune K-itemsets*
6. *K = K+1*
7. *Generate K-itemsets*
8. *End Loop*

Example: relative support =50% (3 records)

BF RTD Lists

DS1	DS2
{4}	{1}
{2,4}	{1,2}



Generated rules

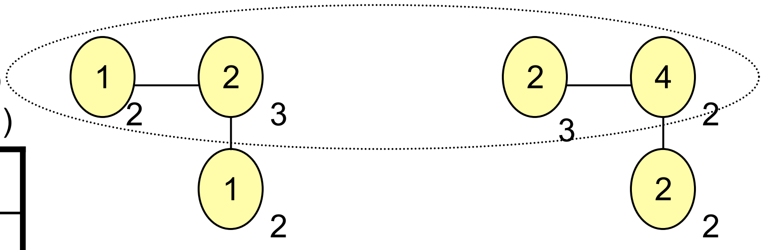
2 → 4 (50%)

4 → 2 (100%)

Merged T-Tree

Apriori RTD Lists (level 1)

DS1	DS2
{4}	{1}



T-Tree

T-Tree

RTD Lists (level 2)

DS1	DS2
{2,4}	null

1	2
1	2
2	4

Dataset 1 (3 X 4)

2	3
2	4
2	4

Dataset 2 (3 X 4)

relative support =50% (1.5 records count)

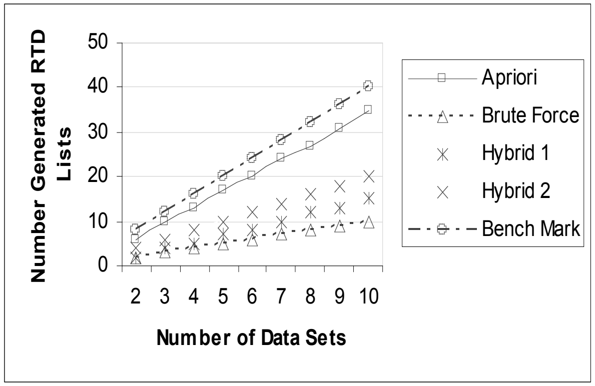
Experimentation and Results

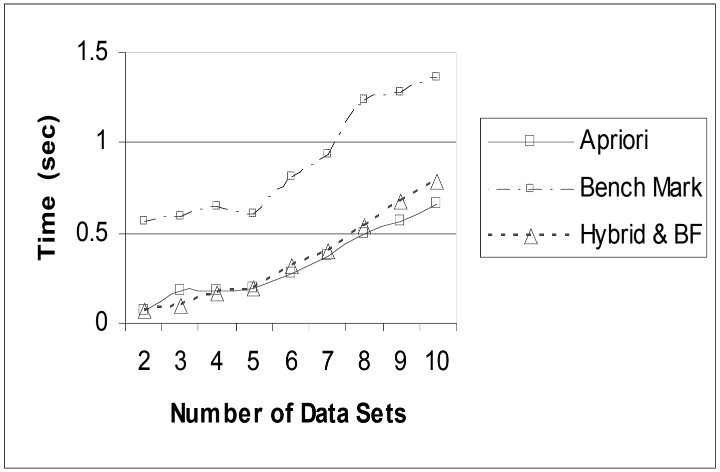
1. The number of data sources (data agents running on different machines)

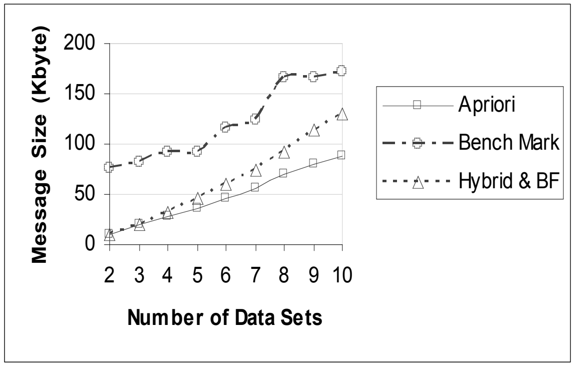
   - *Each dataset has 100,000 transactions.*

   - *Support Threshold = 1%*

- Measured:
  - □ processing time,
  - □ the size of the RTD lists (messages' size)
  - □ and the number of messages.







Conclusion and Current Work

- Conclusion
  - Multi-Agent Systems (MAS) can offer a well suited architecture for data mining in a decentralised, and anarchic manner so that much greater benefit can be obtained from current data mining capabilities and available data.
  - We described a extension of ARM where we build a meta set of frequent itemsets from a collection of component sets which have been generated in an autonomous manner without centralised control in a the EMADS environment.
  - Overall the Hybrid 2 algorithm is the best.
- Current Work
  - Classification application
  - Data normalization (data pre-processing) agents
  - Wrapper agents: agents that can be used to incorporate new mining algorithms into EMADS without significant specialised knowledge of the system.

Questions?