Paper from: *Data Mining III*, A Zanasi, CA Brebbia, NFF Ebecken & P Melli (Editors).
ISBN 1-85312-925-9

Implementing data mining algorithms with Microsoft SQL Server

C. L. Curotto[1] & N. F. F. Ebecken[2]
[1]*CESEC/UFPR – Civil Engineering Graduate Program, Brazil*
[2]*COPPE/UFRJ – Civil Engineering Graduate Program, Brazil*

Abstract

The OLE DB for DM (Microsoft's object-based technology for sharing information and services across process and machine boundaries focused on database mining applications) specification provides an industry standard for implementation of data mining algorithms aggregated with Microsoft SQL Server 2000. The Simple Naive Bayes classifier is implemented using the OLE DB for DM Resource Kit. Numeric input attributes, multiple prediction trees and incremental classification are considered. All necessary steps to implement this algorithm are explained and discussed. Some results are shown to illustrate the capabilities of the implementation.

1 Introduction

Nowadays database system managers like MS (Microsoft) SQL (Standard Query Language) Server [1] are available, with resources for manipulation of terabytes of data with parallel processing of queries (with multiprocessor servers) using microcomputers [2]. This situation suggests the integration of DM technology by using database managers to enlarge the scope of this technology at a low cost.

This approach of integration, achieved by tightly coupling DM and OLAP (On-Line Analytical Processing) techniques in database application development environments, is matter of current interest. It has been discussed in conferences such as ICDM'98 (First International Conference on DM), happened on September 1998 in Rio de Janeiro – Brazil, ICDM'00 (Second International Conference on DM), happened on July 2000, Cambridge – UK and more recent ones.

Agrawal [3] presented a methodology for tightly coupling of DM application to relational database system - IBM DB2/CS – based on utilization of user

Paper from: *Data Mining III*, A Zanasi, CA Brebbia, NFF Ebecken & P Melli (Editors).
ISBN 1-85312-925-9

defined SQL functions. Sarawagi [4] considered a spectrum of architectural alternatives to achieve this coupling. Han [5] leads a group of researchers who develop techniques for DM and OLAP integration. One of his papers shows a project of a query language for DM: DMQL (Data Mining Query Language). Many other papers of this group show the use of OLAP techniques implemented in a database system with DM resources. John [6] supplied some suggestions relative to the use of SQL for machine learning algorithms. Freitas [7] carried out an exhaustive work of research to define SQL primitives used by DM algorithms. Sousa [8-11] implemented a decision tree classification algorithm by using PL/SQL in an Oracle parallel multiprocessor server.

In this way, the MS OLE DB for DM specification [12] was released in July 2000. This specification provides an industry standard for DM so that different DM algorithms from various DM developers can be easily plugged into user applications. OLE DB for DM specifies the API between DM consumers (applications that use DM features) and DM providers (software packages that provide DM algorithms). In September 2000, the MS SQL Server 2000 software was released with an important component: the Analysis Services. Included, the first OLE DB for DM provider supporting two algorithms: one for decision trees classification [13,14] and other for clustering [15]. In March 2001, the DM Aggregator feature for MS SQL Server 2000 Analysis Services was included in the Service Pack 1 [16]. To complete the set of tools the OLE DB for DM Resource Kit [17] was released in June 2001. This kit includes a DM Sample Provider that implements the Naive Bayes algorithm. Netz [18,19] presents an useful overview that describes the OLE DB for DM technology. Siedman brings out an excellent reference to SQL for DM [20].

However, the task of creating a provider and implementing an algorithm remains complex due to the lack of discussion about experiences in doing this work. The main objective of this paper is to describe an experience in this field, making this task affordable for DM algorithm developers. In this direction an enhanced version of the Simple Naive Bayes classifier is implemented considering numeric input attributes, multiple prediction trees and incremental classification.

2 The tools

The implementation was made with an IBM PC compatible microcomputer, Intel Pentium III 500 MHz processor inside, 512 MB of RAM memory, 30 MB hard disk. The operating system is the MS Windows 2000 Advanced Server SP2 (Service Pack 2) with MS SQL Server 2000 Enterprise SP2 installed. The development tools are MS Visual Studio 6.0 SP5 with Visual C++, Visual Java and Visual Basic compilers [21]; MS Platform SDK February 2001 Edition [22] and Sandstone Visual++ Parse 4.00 [23]. The template for developing the DM provider is the Sample Provider of OLE DB for DM Resource Kit [17]. Also Kim´s [24] utility DMSamp is used.

Paper from: *Data Mining III*, A Zanasi, CA Brebbia, NFF Ebecken & P Melli (Editors).
ISBN 1-85312-925-9

3 The implementation

3.1 The start point

The source code included with Sample Provider of OLE DB for DM Resource Kit [17] includes the complete implementation of an aggregated provider as well as the following:

a. All required OLE DB objects, such as session, command, and rowset;
b. The OLE DB for DM Syntax Parser;
c. Tokenization of input data;
d. Query processing engine;
e. A sample Naive Bayes algorithm;
f. Model archiving in XML and binary formats.

The Sample Provider must be prepared to receive the implementation of a new algorithm. The first step is to build the Sample Provider and then handle with a few installations problems (Kim [25]), the source code will be ready to be modified. The next step is to make an extensive correction of the main file of the syntax parser because this file is not clean and when any modification is made to change or insert a new algorithm, Visual Parse crashes and produces a corrupted file. The Sample Provider source code implements an aggregated DM provider. To implement a standalone provider that also can be used in aggregated mode some modifications must be made. This feature is useful for debugging the new algorithm and for using the provider with standalone applications without MS SQL Server. All of these tasks will be described in detail in [26].

After this task is completed, only modifications directly related to the new algorithm should be carried out.

3.2 The Simple Naive Bayes algorithm

This algorithm will not be described in details because it is well known through many papers and books: Han & Kamber book [27] is an excellent and didactic example. Also, many implementations of this algorithm exist, including the one carried out by Witten & Frank [28] in Weka project, used to compare the results of the DM provider implemented.

The main objective of a DM algorithm is to predict attributes based on a set of cases of input attributes. Succinctly the Simple Naive Bayes Classifier uses counts of occurrences of categorical and numeric attributes and means and standard deviations of numeric attributes to do this task. For supporting incremental update of the case set, it is enough to store the sum and the square sum of numeric attributes values, computing means and deviations as necessary. Multiple trees of prediction are supported by an adequate data structure.

3.3 OLE DB for DM

A complete specification of this technology is found in [12]. Netz [18,19] describes the basic philosophy and design decisions leading to the present specifi-

Paper from: *Data Mining III*, A Zanasi, CA Brebbia, NFF Ebecken & P Melli (Editors).
ISBN 1-85312-925-9

cation of OLE DB for DM. He stated precisely the key operations to be supported by a DM provider algorithm on DM models, reproduced as follows:

a. Define a mining model, identifying the set of attributes of data to be predicted, the set of attributes to be used for prediction, and the algorithm used to build the mining model.

b. Populate a mining model from training data using the algorithm specified.

c. Predict attributes for new data using a mining model that has been populated.

d. Browse a mining model for reporting and visualization applications.

These key operations will be described as follows in context of the DM provider implementation. Table 1 presents the results taken from AllEletronics customer database [27] that will be used as a training data set to illustrate the implementation steps. Table 2 presents the marginal model statistics.

Table 1. *AllElet* training data set.

Case	Attribute values				
	Age	**Income**	**Student**	**CreditRating**	**BuysComputer**
1	<=30	high	no	fair	no
2	<=30	high	no	excellent	no
3	31-40	high	no	fair	yes
4	>40	medium	no	fair	yes
5	>40	low	yes	fair	yes
6	>40	low	yes	excellent	no
7	31-40	low	yes	excellent	yes
8	<=30	medium	no	fair	no
9	<=30	low	yes	fair	yes
10	>40	medium	yes	fair	yes
11	<=30	medium	yes	excellent	yes
12	31-40	medium	no	excellent	yes
13	31-40	high	yes	fair	yes
14	>40	medium	no	excellent	no

Table 2. *AllElet* training data set - marginal model statistics

	Attribute values/Number of occurrences				
	Age	**Income**	**Student**	**Credit Rating**	**Buys Computer**
Attribute states	<=30 5	high 4	no 7	fair 8	no 5
	31-40 4	medium 6	yes 7	excellent 6	yes 9
	>40 5	low 4			
Missing	0	0	0	0	0

Paper from: *Data Mining III*, A Zanasi, CA Brebbia, NFF Ebecken & P Melli (Editors).
ISBN 1-85312-925-9

3.4 Creating the mining model

The syntax of the new algorithm is defined using Visual Parse and a few modifications must to be made in Sample Provider projects to insert the support for this new algorithm.

The Relational Mining Model Editor of Analysis Services Manager can be used to create DM SQL commands. Details about this language are found in MS SQL Server Books onLine [1], OLE DB for DM specification [12] and Siedman [20]. Concerning the Sample Provider, the OLE DB for DM Syntax Parser does all necessary steps to create the mining model.

The syntax of SQL for DM command to create a model with two predictive attributes of data set of the Table 1 is shown bellow:

```
CREATE MINING MODEL [AllElet_SNB'S] ([NumReg] LONG   KEY, [Age] TEXT
DISCRETE  , [Income] TEXT   DISCRETE  , [Student] TEXT   DISCRETE  PREDICT,
[CreditRating] TEXT   DISCRETE  , [BuysComputer] TEXT   DISCRETE  PREDICT)
USING Simple_Naive_Bayes
```

3.5 Populating the mining model

The next lines show the syntax of SQL for DM command to populate the mining model using data from MS SQL Server:

```
INSERT INTO [AllElet_SNB'S] (SKIP, [Age], [Income], [Student], [CreditRating],
[BuysComputer]) OPENROWSET ('SQLOLEDB.1', 'Provider=SQLOLEDB;Integrated
Security=SSPI;Persist Security Info=False;Initial Catalog=AllElet;Data Source=CLC',
'SELECT "dbo"."AllEletTrain"."NumReg" AS "NumReg", "dbo"."AllEletTrain"."Age" AS
"Age", "dbo"."AllEletTrain"."Income" AS "Income", "dbo"."AllEletTrain"."Student" AS
"Student", "dbo"."AllEletTrain"."CreditRating" AS "CreditRating",
"dbo"."AllEletTrain"."BuysComputer" AS "BuysComputer" FROM "dbo"."AllEletTrain"')
```

This SQL is generated automatically by Analysis Services Manager. Support for inserting cases in the mining model must be developed for new algorithms. The developer must aim special attention when doing this task. The data structure that represents the model of the algorithm is defined and all functions related to training the data set, assembling the model tree, saving and loading this model are developed. This data structure must support the processing of the two following operations.

3.6 Predicting attributes

The syntax of SQL for DM command to predict attributes must be made by the user using an application such as Kim´s [24] DMSamp or using the Data Transformation Services of MS SQL Server. Support for this task must be developed for new algorithms. An example of this command is shown bellow:

Paper from: *Data Mining III*, A Zanasi, CA Brebbia, NFF Ebecken & P Melli (Editors).
ISBN 1-85312-925-9


```
SELECT FLATTENED [T1].[NumReg], [T1].[Age], [T1].[Income], [T1].[Student],
[T1].[CreditRating], [T1].[BuysComputer], [AllElet_SNB].[BuysComputer] as
BuysComputer FROM [AllElet_SNB] PREDICTION JOIN OPENROWSET
('SQLOLEDB.1', 'Provider=SQLOLEDB.1;Integrated Security=SSPI;Persist Security
Info=False;Initial Catalog=allelet;Data Source=CLC', 'SELECT "NumReg", "Age", "Income",
"Student", "CreditRating", "BuysComputer" FROM "AllElet" ORDER BY "NumReg"') AS
[T1] ON [AllElet_SNB].[NumReg] = [T1].[NumReg] AND [AllElet_SNB].[Age] =
[T1].[Age] AND [AllElet_SNB].[Income] = [T1].[Income] AND [AllElet_SNB].[Student] =
[T1].[Student] AND [AllElet_SNB].[CreditRating] = [T1].[CreditRating] AND
[AllElet_SNB].[BuysComputer] = [T1].[BuysComputer]
```

3.7 Browsing a mining model

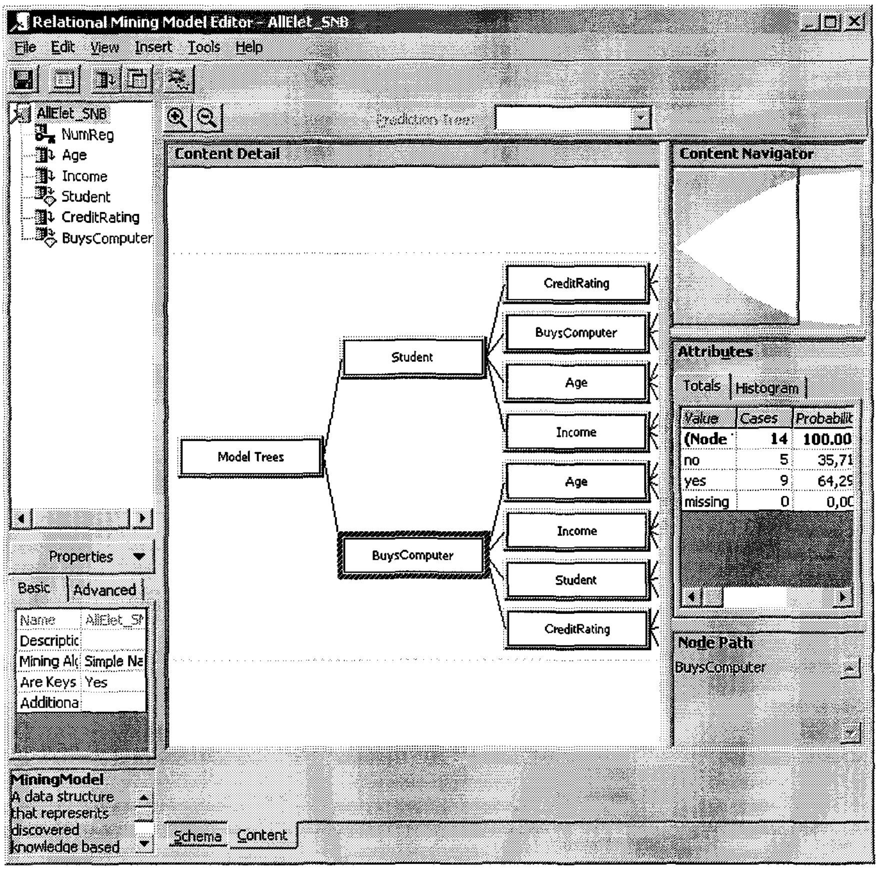

Figure 1. *AllElet* Prediction Model Trees.

Figure 1 shows the Relational Mining Model Editor browsing the two prediction model trees of the test sample and Figure 2 shows the complete tree of BuysComputer prediction attribute. It can be observed that the prediction tree list

Paper from: *Data Mining III*, A Zanasi, CA Brebbia, NFF Ebecken & P Melli (Editors).
ISBN 1-85312-925-9

box is empty. It happens because Microsoft, against itself specification, does not permit this kind of model to be shown normally in your browser. The prediction tree list box is hard coded to look for the Microsoft_Decision_Trees algorithm and other algorithms can't get their content to appear there. Tree node types can't be used without crashing the browser, but using a simple trick this problem is avoided: the root node will be the model node and their children will be the roots of the prediction trees.

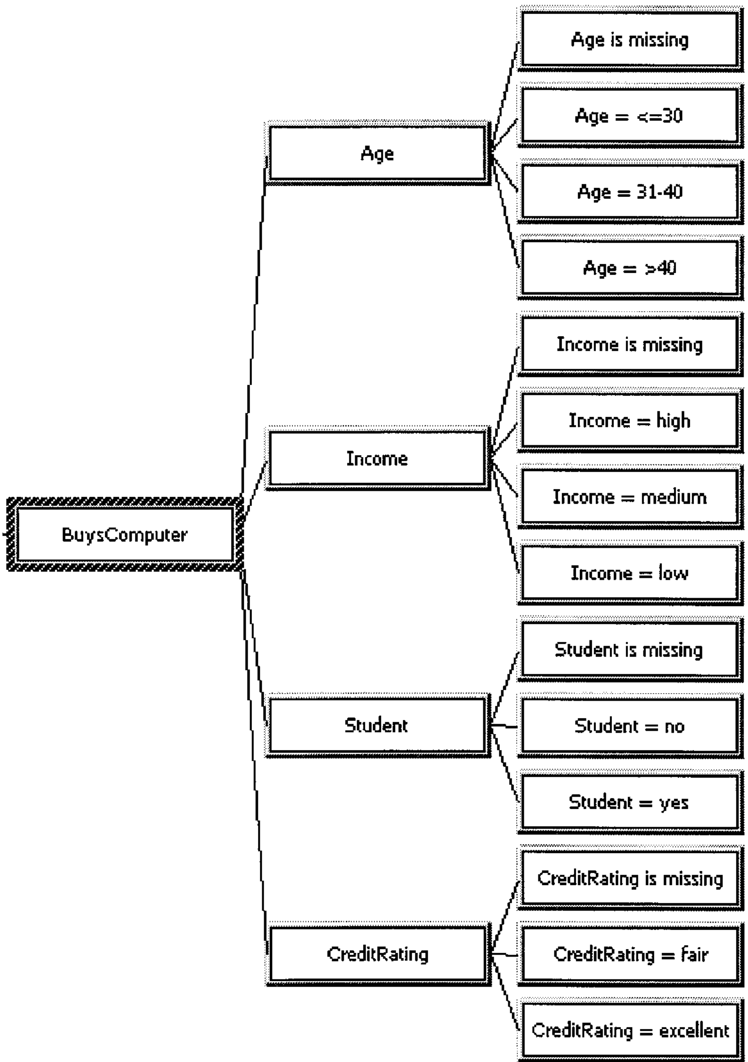

Figure 2. *AllElet* BuysComputer prediction tree.

Paper from: *Data Mining III*, A Zanasi, CA Brebbia, NFF Ebecken & P Melli (Editors).
ISBN 1-85312-925-9

For debugging purpouses and for use in client applications DM SQL queries can be made to retrieve the rowset contents of the DM model. Figure 3 brings out Kim´s [24] utility DMSamp showing part of the results of the query that retrieves the entire contents of the test model:

SELECT * FROM [AllElet_SNB].CONTENT

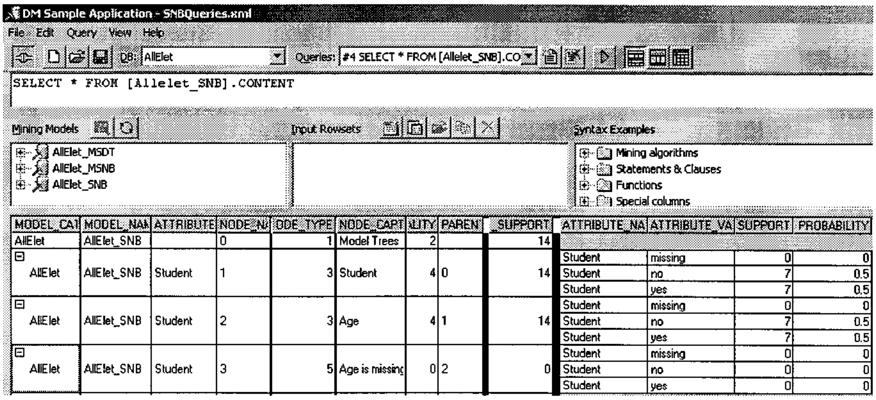

Figure 3. *AllElet* Model rowset contents.

4 Conclusion

The experience of implementing a simple algorithm in OLE DB for DM technology was very useful as a start point for doing more complex tasks. The sharing of this experience will help other developers and researchers doing similar work. This technology seems to be an excelent tool to do efficient implementation of data mining algorithms to achieve the complete database querying and mining integration.

Acknowledgments

To CAPES and UFPR – Federal University of Paraná for its financial support. To Kris Ganjam, Claude Seidman, Raman Iyer, Peter Kim and Jamie MacLennan for their valuable help.

References

[1] Microsoft Corporation, *SQL Server 2000*, Microsoft Corporation, Redmond, Washington, USA, 2000. http://www.microsoft.com/sql.

[2] Microsoft Corporation, *Visual Studio Developing for the Enterprise*, Microsoft Corporation, Redmond, Washington, USA, 1998. http://www.microsoft.com.

Paper from: *Data Mining III*, A Zanasi, CA Brebbia, NFF Ebecken & P Melli (Editors).
ISBN 1-85312-925-9

[3] Agrawal, R. & Shim, K., *Developing Tightly-Coupled Applications on IBM DB2/CS Relational Database System: Methodology and Experience*, IBM Research Report RJ 10005, IBM Almaden Research Center, San Jose, California, USA, 1995. http://www.almaden.ibm.com/cs/people/ragrawal/papers/udfrj.ps.

[4] Sarawagi, S., Thomas, S. & Agrawal, R., *Integrating Association Rule Mining with Databases: Alternatives and Implications*, Proc. of the ACM SIGMOD 1998 Int'l Conf. on Management of Data, Seattle, Washington, USA, 1998. http://www.almaden.ibm.com/cs/people/ragrawal/papers/sigmod98_dbi_rj.ps.

[5] Han, J. et alii, *DMQL: A Data Mining Query Language for Relational Databases*, Proc. of the DMKD'96 - SIGMOD'96 Workshop on Research Issues on Data Mining and Knowledge Discovery, Montreal, Canada, 1996. ftp://ftp.fas.sfu.ca/pub/cs/han/kdd/dmql96.ps.

[6] John, G. H., *Enhancements to the Data Mining Process*, Ph.D. Dissertation, Stanford University, CA, USA, 1997. http://robotics.stanford.edu/users/gjohn.

[7] Freitas, A. A., *Generic, Set-Oriented Primitives to Support Data-Parallel Knowledge Discovery in Relational Database Systems*, Ph.D. Thesis, University of Essex, UK, 1997. http://www.ppgia.pucpr.br/~alex/thesis.html.

[8] Sousa, M. S., Mattoso, M. L. Q. & Ebecken, N. F. F., *Data Mining on Parallel Database Systems*, Proc.of the Int. Conf. on PDPTA, Special Session on Parallel Data Warehousing, CSREA Press, Las Vegas, NV, USA, 1998. http://www.cos.ufrj.br/~mauros/.

[9] Sousa, M. S., Mattoso, M. L. Q. & Ebecken, N. F. F., *Data Mining: A Tightly- Coupled Implementation using a Parallel Database Server*, Proc. of the Int. Conf. on DEXA Workshop "Parallel Databases: innovative applications and new architecture", IEEE CS, Vienna, Austria, 1998. http://www.cos.ufrj.br/~mauros/.

[10] Sousa, M. S., Mattoso, M. L. Q. & Ebecken, N. F. F., *Data Mining: a Database Perspective*, Data Mining (Proc. of ICDM'98 - 1st Int'l Conf. on Data Mining, Rio de Janeiro, Brazil), Ebecken, N. F. F. (editor), WIT Press, pp 7-20, 1998. http://www.cos.ufrj.br/~mauros/.

[11] Sousa, M. S., *Mineração de Dados: uma Implementação Fortemente Acoplada a um Sistema Gerenciador de Banco de Dados Paralelo (Data Mining: A Tightly- Coupled Implementation using a Parallel Database Server)*, Tese de Mestrado (M.Sc. Thesis), COPPE/UFPR – Rio de Janeiro – RJ – Brasil, 1998. http://www.cos.ufrj.br/~mauros/.

[12] Microsoft Corporation, *OLE DB for Data Mining Specification Version 1.0*, Microsoft Corporation, Redmond, Washington, USA, 2000. http://www.microsoft.com/data/oledb/dm.htm.

[13] Chickering, D. M., Geiger, D. & Heckerman, D., *Learning Bayesian Networks: The Combination of Knowledge and Statistical Data*, Technical Report MSR-TR-94-09, Microsoft Research, Microsoft Corporation, Redmond, Washington, USA, 1994. ftp://ftp.research.microsoft.com/pub/tr/tr-94-09.ps.

Paper from: *Data Mining III*, A Zanasi, CA Brebbia, NFF Ebecken & P Melli (Editors).
ISBN 1-85312-925-9

[14] Bernhardt J., Chaudhuri S. and Fayyad U. M., *Scalable Classification over SQL Databases*, Proceedings of 15th Int'l Conf. on Data Engineering, Sydney, Australia, 1999. ftp://ftp.research.microsoft.com/users/AutoAdmin/icde99.pdf.

[15] Bradley, P. S., Fayyad, U. M & Reina, C. A., *Scaling EM (Expectation Maximization) Clustering to Large Databases*, Technical Report MSR-TR-98-35, Microsoft Research, Microsoft Corporation, Redmond, Washington, USA, 1999. ftp://ftp.research.microsoft.com/pub/tr/tr-98-35.pdf.

[16] Seth, P.l, *Third-Party Data Mining Providers*, White paper, Microsoft Research, Microsoft Corporation, Redmond, Washington, USA, 2001. http://www.microsoft.com/sql/techinfo/BI/2000/DMAggregator.doc.

[17] Microsoft Corporation, *OLE DB for Data Mining Resource Kit*, Microsoft Corporation, Redmond, Washington, USA, 2002. http://www.microsoft.com/data/oledb/DMResKit.htm.

[18] Netz, A, Chaudhuri, S., Bernhardt, J. & Fayyad, U. M, *Integration of Data Mining and Relational Databases*, Proc. of 26th Int'l Conf. on Very Large Data Bases, Cairo, Egypt, 2000. ftp://ftp.research.microsoft.com/users/AutoAdmin/vldb00DM.pdf.

[19] Netz A., Bernhardt J., Chaudhuri S. and Fayyad U., *Integrating Data Mining with SQL Databases: OLE DB for Data Mining*, Proc. of 17th Int'l Conf. on Data Engineering, Heidelberg, Germany, 2001. ftp://ftp.research.microsoft.com/users/AutoAdmin/netza_oledb.pdf.

[20] Seidman, C., *Data Mining with Microsoft SQL Server 2000 - Technical Reference*, Microsoft Press, Redmond, Washington, USA, 2001.

[21] Microsoft Corporation, *Visual Studio 6.0*, Microsoft Corporation, Redmond, Washington, USA, 1998. http://msdn.microsoft.com/vstudio/.

[22] Microsoft Corporation, *Microsoft Platform Software Development Kit*, Microsoft Corporation, Redmond, Washington, USA, 2001. http://www.microsoft.com/msdownload/platformsdk/setuplauncher.asp.

[23] Sandstone Tecnology Inc., *Visual Parse++ Version 4.00*, Sandstone Tecnology Inc., Carlsbad, California, USA, 2000. http://www.sand-stone.com.

[24] Kim, P. & Carroll, M., *Making OLE DB for Data Mining queries against a DM provider*, Visual Basic utility application hosted by Data Mining Community Web Site, 2002. http://communities.msn.com/AnalysisServicesDataMining.

[25] Kim, P., *Microsoft.public.sqlserver.datamining FAQ (Frequently Asked Questions and Answers)*, Resource hosted by Data Mining Community Web Site, 2002. http://communities.msn.com/AnalysisServicesDataMining.

[26] Curotto, C. L. & Ebecken, N. F. F., *Implementing New Data Mining Algorithms in the SQL Server 2000*, to appear, 2002.

[27] Han, J. & Kamber, M., *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, San Francisco, CA, USA, 2001.

[28] Witten, I. H. & Frank, E., *Data Mining: Practical Machine Learning Tools with Java Implementations*, Morgan Kaufmann Publishers, CA, USA, 2000. http://www.cs.waikato.ac.nz/~ml.