MINING MODEL CONTENT

Aastha Trehan, Ritika Grover, Prateek Puri
Dronacharya College Of Engineering, Gurgaon

***Abstract*- This paper highlights about the mining model content used in data mining. The mining model is complete after you have designed and processed a mining model using data from the underlying mining structure and contains mining model content. You can use this content to make predictions or analyze your data. This paper describes in-depth the structure of mining model content, nodes in the model content, mining model content by algorithm type and tools for viewing and querying mining model content.**

***Index Terms*- Mining Model Conent, Mining Model, Mining Content Nodes, Tools**

I. INTRODUCTION

Mining model content includes metadata about the model, statistics about the data, and patterns discovered by the mining algorithm. Depending on the algorithm that was used, the model content may include regression formulas, the definitions of rules and itemsets, or weights and other statistics.

Regardless of the algorithm that was used, mining model content is presented in a standard structure. You can browse the structure in the Microsoft Generic Content Tree Viewer, provided in SQL Server Data Tools (SSDT), and then switch to one of the custom viewers to see how the information is interpreted and displayed graphically for each model type. You can also create queries against the mining model content by using any client that supports the MINING_MODEL_CONTENT schema rowset.

II. STRUCTURE OF MINING MODEL CONTENT

The content of each model is presented as a series of *nodes*. A node is an object within a mining model that contains metadata and information about a portion of the model. Nodes are arranged in a hierarchy. The exact arrangement of nodes in the hierarchy, and the meaning of the hierarchy, depends on the algorithm that you used. For example, if you create a decision trees model, the model can contain multiple trees, all connected to the model root; if you create a neural network model, the model may contain one or more networks, plus a statistics node.

NODE_TYPE ID	Node Label	Node Contents
1	Model	Metadata and root content node. Applies to all model types.
2	Tree	Root node of a classification tree. Applies to decision tree models.
3	Interior	Interior split node in a tree. Applies to decision tree models.
4	Distribution	Terminal node of a tree. Applies to decision tree models.
5	Cluster	Cluster detected by the algorithm. Applies to clustering models and sequence clustering models.
6	Unknown	Unknown node type.
7	ItemSet	Itemset detected by the algorithm. Applies to association models or

		sequence clustering models.
8	AssociationRule	Association rule detected by the algorithm. Applies to association models or sequence clustering models.
9	PredictableAttribute	Predictable attribute. Applies to all model types.
10	InputAttribute	Input attribute. Applies to decision trees and Naïve Bayes models.
11	InputAttributeState	Statistics about the states of an input attribute. Applies to decision trees and Naïve Bayes models.
13	Sequence	Top node for a Markov model component of a sequence cluster. Applies to sequence clustering models.
14	Transition	Markov transition matrix. Applies to sequence clustering models.
15	TimeSeries	Non-root node of a time series tree. Applies to time series models only.
16	TsTree	Root node of a time series tree that corresponds to a predictable time series. Applies to time series models, and only if the model was created using the MIXED parameter.
17	NNetSubnetwork	One sub-network. Applies to neural network models.
18	NNetInputLayer	Group that contains the nodes of the input layer. Applies to neural network models.
19	NNetHiddenLayer	Groups that contains the nodes that describe the hidden layer. Applies to neural network models.
21	NNetOutputLayer	Groups that contains the nodes of the output layer. Applies to neural network models.
21	NNetInputNode	Node in the input layer that matches an input attribute with the corresponding states. Applies to neural network models.
22	NNetHiddenNode	Node in the hidden layer. Applies to neural network models.
23	NNetOutputNode	Node in the output layer. This node will usually match an output attribute and the corresponding states. Applies to neural network models.
24	NNetMarginalNode	Marginal statistics about the training set. Applies to neural network models.
25	RegressionTreeRoot	Root of a regression tree. Applies to linear regression models and to decision trees models that contains continuous input attributes.
26	NaiveBayesMarginalStatNode	Marginal statistics about the training set. Applies to Naïve Bayes models.
27	ArimaRoot	Root node of an ARIMA model. Applies to only those time series models

		that use the ARIMA algorithm.
28	ArimaPeriodicStructure	A periodic structure in an ARIMA model. Applies to only those time series models that use the ARIMA algorithm.
29	ArimaAutoRegressive	Autoregressive coefficient for a single term in an ARIMA model. Applies to only those time series models that use the ARIMA algorithm.
30	ArimaMovingAverage	Moving average coefficient for a single term in an ARIMA model. Applies to only those time series models that use the ARIMA algorithm.
1000	CustomBase	Starting point for custom node types. Custom node types must be integers greater in value than this constant. Applies to models created by using custom plug-in algorithms.

The first node in each model is called the *root node*, or the *model parent* node. Every model has a root node (NODE_TYPE = 1). The root node typically contains some metadata about the model, and the number of child nodes, but little additional information about the patterns discovered by the model.

Depending on which algorithm you used to create the model, the root node has a varying number of child nodes. Child nodes have different meanings and contain different content, depending on the algorithm and the depth and complexity of the data.

III. Nodes In Mining Model Content

In a mining model, a node is a general-purpose container that stores a piece of information about all or part of the model. The structure of each node is always the same, and contains the columns defined by the data mining schema rowset. For more information, see DMSCHEMA_MINING_MODEL_CONTENT Rowset.

Each node includes metadata about the node, including an identifier that is unique within each model, the ID of the parent node, and the number of child nodes that the node has. The metadata identifies the model to which the node belongs, and the database catalog where that particular model is stored. Additional content provided in the node differs depending on the type of algorithm you used to create the model, and might include the following:

- Count of cases in the training data that supports a particular predicted value.
- Statistics, such as mean, standard deviation, or variance.
- Coefficients and formulas.
- Definition of rules and lateral pointers.
- XML fragments that describe a portion of the model.

A. List Of Mining Content Node Types

The following table lists the different types of nodes that are output in data mining models. Because each algorithm processes information differently, each model generates only a few specific kinds of nodes. If you change the algorithm, the type of nodes may change. Also, if you reprocess the model, the content of each node may change.

B. Node ID, Name, Caption And Description

The root node of any model always has the unique ID (**NODE_UNIQUE_NAME**) of 0. All node IDs are assigned automatically by Analysis Services and cannot be modified.

The root node for each model also contains some basic metadata about the model. This metadata includes the Analysis Services database where the model is stored (**MODEL_CATALOG**), the schema (**MODEL_SCHEMA)**, and the name of the model (**MODEL_NAME)**. However, this information is repeated in all the nodes of the model, so you do not need to query the root node to get this metadata.

the In addition to a name used as the unique identifier, each node has a *name* (**NODE_NAME**). This name is automatically created by the algorithm for display purposes and cannot be edited.

The *caption* and *description* for each node are automatically generated by the algorithm, and serve as labels to help you understand the content of the node. The text generated for each field depends on the model type. In some cases, the name, caption, and description may contain exactly the same string, but in some models, the description may contain additional information. See the topic about the individual model type for details of implementation.

C. Parents, Node Children, And Node Cardinality Node

The relationship between parent and child nodes in a tree structure is determined by the value of the PARENT_UNIQUE_NAME column. This value is stored in the child node and tells you the ID of the parent node. Some examples follow of how this information might be used:

- A PARENT_UNIQUE_NAME that is NULL means that the node is the top node of the model.
- If the value of PARENT_UNIQUE_NAME is 0, the node must be a direct descendant of the top node in the model. This is because the ID of the root node is always 0.
- You can use functions within a Data Mining Extensions (DMX) query to find descendants or parents of a particular node. For more information about using functions in queries, see Data Mining Queries.

Cardinality refers to the number of items in a set. In the context of a processed mining model, cardinality tells you the number of children in a particular node. For example, if a decision tree model has a node for [Yearly Income], and that node has two child nodes, one for the condition [Yearly Income] = High and one for the condition, [Yearly Income] = Low, the value of CHILDREN_CARDINALITY for the [Yearly Income] node would be 2.

Although cardinality is counted in the same way for all models, how you interpret or use the cardinality value differs depending on the model type. For example, in a clustering model, the cardinality of the top node tells you the total number of clusters that were found. In other types of models, cardinality may always have a set value depending on the node type. For more information about how to interpret cardinality, see the topic about the individual model type.

D. Node Distribution

The NODE_DISTRIBUTION column contains a nested table that in many nodes provides important and detailed information about the patterns discovered by the algorithm. The exact statistics provided in this table change depending on the model type, the position of the node in the tree, and whether the predictable attribute is a continuous numeric value or a discrete value; however, they can include the minimum and maximum values of an attribute, weights assigned to values, the number of cases in a node, coefficients used in a regression formula, and statistical measures such as standard deviation and variance. For more information about how to interpret node distribution, see the topic for the specific type of model type that you are working with.

The nested table, NODE_DISTRIBUTION, always contains the following columns. The content of each column varies depending on the model type.

ATTRIBUTE_NAME

Content varies by algorithm. Can be the name of a column, such as a predictable attribute, a rule, an itemset, or a piece of information internal to the algorithm, such as part of a formula.

This column can also contain an attribute-value pair.

ATTRIBUTE_VALUE

Value of the attribute named in ATTRIBUTE_NAME.

If the attribute name is a column, then in the most straightforward case, the ATTRIBUTE_VALUE contains one of the discrete values for that column.

Depending on how the algorithm processes values, the ATTRIBUTE_VALUE can also contain a flag that tells you whether a value exists for the attribute (**Existing**), or whether the value is null (**Missing**).

For example, if your model is set up to find customers who have purchased a particular item at least once, the ATTRIBUTE_NAME column might contain the attribute-value pair that defines the item of interest, such as Model = 'Water bottle', and the ATTRIBUTE_VALUE column would contain only the keyword **Existing** or **Missing**.

SUPPORT

Count of the cases that have this attribute-value pair, or that contain this itemset or rule.

In general, for each node, the support value tells you how many cases in the training set are included in the current node. In most model types, support represents an exact count of cases. Support values are useful because you can view the distribution of data within your training cases without having to query the training data. The Analysis Services server also uses these stored values to calculate stored probability versus prior probability, to determine whether inference is strong or weak.

For example, in a classification tree, the support value indicates the number of cases that have the described combination of attributes. In a decision tree, the sum of support at each level of a tree sums to the support of its parent node. For example, if a model containing 1200 cases is divided equally by gender, and then subdivided equally by three values for Income—Low, Medium, and High—the child nodes of node (2), which are nodes (4), (5) and (6), always sum to the same number of cases as node (2).

Node ID and node attributes	Support count
(1) Model root	1200
(2) Gender = Male	600
(3) Gender = Female	600
(4) Gender = Male and Income = High	200
(5) Gender = Male and Income = Medium	200
(6) Gender = Male and Income = Low	200
(7) Gender = Female and Income = High	200
(8) Gender = Female and Income = Medium	200
(9) Gender = Female and Income = Low	200

For a clustering model, the number for support can be weighted to include the probabilities of belonging to multiple clusters. Multiple cluster membership is the default clustering method. In this scenario, because each case does not necessarily belong to one and only one cluster, support in these models might not add up to 100 percent across all clusters.

E. Node Score

The meaning of the node score differs depending on the model type, and can be specific to the node type as well. For information about how NODE_SCORE is calculated for each model and node type, see Mining Model Content by Algorithm Type.

F. Node Probability and Marginal Probability

The mining model schema rowset includes the columns NODE_PROBABILITY and MARGINAL_PROBABILITY for all model types. These columns contain values only in nodes where a probability value is meaningful. For example, the root node of a model never contains a probability score.

In those nodes that do provide probability scores, the node probability and marginal probabilities represent different calculations.

- **Marginal probability** is the probability of reaching the node from its parent.
- **Node probability** is the probability of reaching the node from the root.
- **Node probability** is always less than or equal to **marginal probability**.

IV. Mining Model Content By Algorithm Type

Each algorithm stores different types of information as part of its content schema. For example, the Microsoft Clustering Algorithm generates many child nodes, each of which represents a possible cluster. Each cluster node contains rules that describe characteristics shared by items in the cluster. In contrast, the Microsoft Linear Regression algorithm does not contain any child nodes; instead, the parent node for the model contains the equation that describes the linear relationship discovered by analysis.

The following table provides links to topics for each type of algorithm.

- **Model content topics:** Explain the meaning of each node type for each algorithm type, and provide guidance about which nodes are of most interest in a particular model type.
- **Querying topics:** Provide examples of queries against a particular model type and guidance on how to interpret the results.

Algorithm or Model Type	Model Content	Querying Mining Models
Association rules models	Mining Model Content for Association Models (Analysis Services - Data Mining)	Association Model Query Examples
Clustering models	Mining Model Content for Decision Tree Models (Analysis Services - Data Mining)	Clustering Model Query Examples
Decision trees model	Mining Model Content for Decision Tree Models (Analysis Services - Data Mining)	Decision Trees Model Query Examples
Linear regression models	Mining Model Content for Linear Regression Models (Analysis Services - Data Mining)	Linear Regression Model Query Examples
Logistic regression models	Mining Model Content for Logistic Regression Models (Analysis Services - Data Mining)	Linear Regression Model Query Examples
Naïve Bayes models	Mining Model Content for Naive Bayes Models (Analysis Services - Data Mining)	Naive Bayes Model Query Examples
Neural network	Mining Model Content for Neural Network Models (Analysis Services	Neural Network Model Query

models	- Data Mining)	Examples
Sequence clustering	Mining Model Content for Sequence Clustering Models (Analysis Services - Data Mining)	Sequence Clustering Model Query Examples
Time series models	Mining Model Content for Time Series Models (Analysis Services - Data Mining)	Time Series Model Query Examples

V. TOOLS FOR VIEWING MINING MODEL CONTENT

When you browse or explore a model in SQL Server Data Tools (SSDT), you can view the information in the **Microsoft Generic Content Tree Viewer**, which is available in both SQL Server Data Tools (SSDT) and SQL Server Management Studio.

The Microsoft Generic Content Viewer displays the columns, rules, properties, attributes, nodes, and other content from the model by using the same information that is available in the content schema rowset of the mining model. The content schema rowset is a generic framework for presenting detailed information about the content of a data mining model. You can view model content in any client that supports hierarchical rowsets. The viewer in SQL Server Data Tools (SSDT) presents this information in an HTML table viewer that represents all models in a consistent format, making it easier to understand the structure of the models that you create.

VI. TOOLS FOR QUERYING MINING MODEL CONTENT

To retrieve mining model content, you must create a query against the data mining model.

The easiest way to create a content query is to execute the following DMX statement in SQL Server Management Studio:

SELECT * FROM [<mining model name>].CONTENT

For more information, see Data Mining Queries.

You can also query the mining model content by using the data mining schema rowsets. A schema rowset is a standard structure that clients use to discover, browse, and query information about mining structures and models. You can query the schema rowsets by using XMLA, Transact-SQL, or DMX statements.

In SQL Server 2014, you can also access the information in the data mining schema rowsets by opening a connection to the Analysis Services instance and querying the system tables.

VII. CONCLUSION

Data mining is the process of analyzing data from different perspectives and summarizing it into useful information. mining model is complete after you have designed and processed a mining model using data from the underlying mining structure and contains mining model content. This paper highlights the mining model content used in data mining. The content for s mining model include key parameters such as the structure of the mining model content, the nodes involved as well as the mining model content by algorithm type. All these factors, including the tools for viewing and querying mining model content have been discussed in this paper, thereby highlighting the process and importance of mining model content in data mining.

REFERENCES

[1] http://msdn.microsoft.com/

[2] http://technet.microsoft.com

[3] http://link.springer.com/article/10.1007%2Fs10586-013-0308-1

[4] http://www.dmg.org/v4-1/GeneralStructure.html