

## ДОПОЛНИТЕЛЬНЫЙ DATA MINING МОДУЛЬ ДЛЯ MICROSOFT SQL SERVER 2005 НА ОСНОВЕ СИСТЕМЫ DISCOVERY \*

Компания Microsoft осуществляет интеграцию методов Data Mining в Microsoft SQL Server 2005. Сейчас в SQL запросе к Microsoft SQL Server 2005 можно не только формировать выборку данных, но и вызывать Data Mining модуль, прилагаемый к Microsoft SQL Server 2005. Компания Microsoft предоставила интерфейс прикладного программирования компаниям занимающимся разработкой методов Data Mining для реализации и интеграции своих собственных методов в виде дополнительного модуля к SQL Server 2005. Целью данной работы является разработка дополнительного модуля для Microsoft SQL Server 2005, реализующего систему интеллектуальному анализу данных Discovery. Эта система реализует реляционный подход к обнаружению знаний (Relational Data Mining), основанный на логическом (relational) представлении информации в данных и обнаружении закономерностей путем логико-вероятностного семантического вероятностного вывода. Реляционный подход и реализующая его система Discovery были адаптированы для работы с источниками данных под управлением СУБД Microsoft SQL Server 2005. В результате реализованный программный модуль позволяет извлекать из данных знания, обладающие свойствами статистической значимости и непротиворечивости, а также демонстрирует высокий уровень точности построения прогнозов.

*Ключевые слова:* СУБД, SQL Server, модуль, Data Mining, scientific discovery, прогнозирование.

### Введение

На сегодняшний день в научной и производственной деятельности возникает все больше задач, связанных с обработкой больших массивов данных и извлечения знаний для поддержки принятия решений, выявления тенденций и взаимосвязей в них. В связи с этим все большее распространение получают системы хранения и управления базами данных, а также средства интеллектуального анализа данных Data Mining. В связи с постоянным совершенствованием и усложнением программных средств растет потребность в комплексных решениях для работы с данными. Поэтому все большую актуальность получают возможности самих СУБД по интеллектуальному анализу данных.

Microsoft SQL Server 2005 реализует концепцию интеграции СУБД и средств Data Mining и унификацию этих средств. SQL Server включает в себя Business Intelligence средства, позволяющие создавать проекты по анализу данных в среде Microsoft Visual Studio 2005. Таким образом, обеспечивается единый интерфейс работы с различными Data Mining моделями. Каждая Data Mining модель реализована в виде отдельного провайдера Data Mining функций, который может либо являться частью аналитической службы и входить в стандартную поставку SQL Server, либо представлять собой отдельный программный модуль. Среди Data Mining провайдеров, входящих в поставку SQL Server, есть модели, основанные на кластеризации, ассоциативных правилах, деревьях решений, нейронных сетях и т. д. Для анализа могут быть использованы данные реляционных таблиц или многомерных

---

\* Работа выполнена при финансовой поддержке РФФИ (проект № 08-07-00272а), в рамках интеграционного проекта СО РАН № 1 и 115, а также Государственного контракта 2007-4-1.4-00-04 и Совета по грантам Президента РФ и государственной поддержке ведущих научных школ (проект НШ-335.2008.1).

кубов. Также компания Microsoft активно развивает другие направления интеграции с Data Mining средствами SQL Server. Для этого, например, были созданы инструменты анализа для Excel и наборы шаблонов для Visio, входящих в состав Office 2007. Microsoft активно участвует в разработке и выступает за использование стандартов в области Data Mining, таких как XML for Analysis, PMML, OLE DB for Data Mining.

Идея расширения Data Mining возможностей СУБД имеет ряд преимуществ перед созданием независимого приложения. Одним из ключевых факторов здесь является экономия времени разработки системы и переиспользование функциональности СУБД. Вторым важным фактором является удобство использования дополнительного программного модуля. Это связано, прежде всего, с уменьшением объема системных ресурсов, которые требуются такому приложению, и наличием механизмов интеграции с СУБД. В случае, когда модификация данных, а также их структуры, и обработка данных происходят под управлением СУБД, это решает многие проблемы с синхронизацией и актуальностью данных. Также интегрированные средства предоставляют пользователю единый интерфейс и экономят время аналитика на подготовку данных.

Тем не менее существующие Data Mining провайдеры для SQL Server не обладают достаточным потенциалом для решения серьезных задач, требующих выявления сложных закономерностей в данных и обеспечения прозрачности построения решений и прогнозов, таких как финансовые или медицинские задачи. Алгоритмы, основанные на деревьях решений, линейной регрессии, кластеризации и т. п., подходят для решения достаточно простых задач, таких как выявление ассоциаций и корреляций между признаками [1], например задача анализа потребительской корзины. При этом сложные статистически значимые закономерности такими алгоритмами не обнаруживаются. Алгоритмы, основанные на нейронных сетях, не дают информации о том, на основе каких закономерностей сделан тот или иной вывод или просторен прогноз. Закономерности не могут быть подвергнуты критическому анализу эксперта, что ограничивает применимость таких алгоритмов для решения серьезных задач.

В данной работе описываются идеи и проблемы, связанные с разработкой дополнительного программного модуля (plug-in) для Microsoft SQL Server 2005 [2], который расширяет возможности SQL Server по интеллектуальному анализу данных. Основная функциональность такого модуля заключается в извлечении знаний из различных источников данных под управлением этой СУБД, построении прогнозов и предсказании поведения некоторых величин, также важна функциональность, относящаяся к интеграции с различными инструментами Business Intelligence, доступными пользователям Microsoft SQL Server 2005.

### Перечень терминов

*Признак* – анализируемая величина, имеющая в общем случае вещественный набор значений.

*Атомарный интервал* – множество значений признака, представляющее собой отрезок на прямой вещественных чисел. Атомарные интервалы признака не пересекаются. Множество атомарных интервалов некоторого признака конечно и покрывает весь интервал значений этого признака.

*Комбинированный интервал* – множество значений признака, представимое в виде объединения атомарных интервалов этого признака.

*Правило* – форма логической закономерности между значениями признаков, которой оперирует алгоритм Discovery.

$A_1 \& \dots \& A_n \rightarrow B$ , где  $A_1, \dots, A_n, B$  – это предикаты, утверждения о том, что значение некоторой наблюдаемой величины принадлежит некоторому комбинированному интервалу.  $B$  называется целевой (THEN) частью.  $A_1 \& \dots \& A_n$  называется условной (IF) частью.

*Подправило правила  $C$*  – правило  $H$  такое, что целевая часть правила  $H$  совпадает с целевой частью правила  $C$ , а условная часть правила  $H$  может быть получена путем удаления некоторых предикатов из конъюнкции, представляющей собой условную часть правила  $C$ .

*ОУВ (оценка условной вероятности) правила* – отношение количества экспериментов, на которых истинны условная и целевая части правила, к количеству экспериментов, на которых истинна только условная часть.

*ВЗ (вероятностный закон)* – правило, условная вероятность которого определена и строго больше условных вероятностей каждого из его подправил.

*СВЗ (сильнейший вероятностный закон)* – ВЗ, который не является подправилом никакого другого ВЗ.

*СВВ (семантический вероятностный вывод) некоторого СВЗ* – последовательность вероятностных законов  $C_1, C_2, \dots, C_n$  таких, что правило  $C_i$  является подправилом правила  $C_{i+1}$  и условная вероятность правила  $C_{i+1}$  больше, чем условная вероятность правила  $C_i$ .

*МСЗ (максимально специфический закон) вывода факта  $G$*  – СВЗ дерева семантического вероятностного вывода факта  $G$ , имеющий максимальное значение условной вероятности среди других СВЗ дерева семантического вероятностного вывода факта  $G$ .

### Применяемый подход

Работа выполняется в рамках разработки реляционного подхода к обнаружению знаний (Relational Data Mining [3; 4]) и реализующей его системе Discovery. Данный подход основан на логическом (relational) представлении информации в данных и обнаружении закономерностей в данных путем логико-вероятностного семантического вероятностного вывода (СВВ) [5]. Разрабатываемый программный модуль реализует некоторый вариант системы Discovery применительно к задаче интеллектуального анализа данных под управлением Microsoft SQL Server 2005 с использованием функций аналитических служб и в соответствии со стандартами, реализуемыми Data Mining провайдерами СУБД.

Предлагаемый реляционный подход и реализующая его система Discovery по многим свойствам превосходит другие Data Mining подходы [3; 5], предоставляемые пользователям SQL Server 2005. В данной работе реляционный подход реализуется в несколько упрощенной форме, когда признаки и данные могут быть представлены одноместными предикатами.

Наиболее близким к данной версии системы Discovery подходом можно считать поиск ассоциативных правил (Microsoft Association Rules) [2], ввиду того что закономерности представляются также в форме логических правил. Тем не менее между ними существует ряд принципиальных отличий:

- в общем случае реляционный подход оперирует произвольным набором предикатов, что позволяет находить более сложные закономерности в данных, такие как, например, монотонность;

- в детерминированном случае, когда нет шума в данных и нет объектов с одинаковыми признаками, принадлежащими разным классам, система Discovery может обнаружить одно правило  $A \& B \Rightarrow C$ , истинное на данных, тогда как алгоритм, обнаруживающий ассоциативные правила, должен обнаружить все правила вида  $A \& B \& \dots \& D \Rightarrow C$ , которые получаются из правила  $A \& B \Rightarrow C$  добавлением дополнительных условий  $D, F, \dots$ , например:  $A \& B \& D \Rightarrow C, A \& B \& F \Rightarrow C$ ;

- когда есть шум в данных и классы пересекаются, система Discovery может обнаружить одно правило  $A \& B \Rightarrow C$ , представляющее собой вероятностный закон с определенным уровнем статистической значимости, тогда как алгоритм, обнаруживающий ассоциативные правила, должен обнаружить все детерминированные правила вида  $A \& B \& D \Rightarrow C, A \& B \& F \Rightarrow C$ ;

- реляционный подход в большей степени применим к прогнозированию, ввиду того что он позволяет обнаруживать максимально специфические законы для каждого из анализируемых признаков.

### Описание алгоритмической части

*Предварительная обработка данных.* Разрабатываемый Data Mining модуль оперирует с данными, представленными в виде интервалов значений признаков [6]. В текущей

реализации модуль может работать с дискретными (discrete), непрерывными (continuous) и номинальными (key) типами данных. Для работы с непрерывными и дискретными типами применяется специальная схема кластеризации с учетом специфичности этих типов.

Для сравнения: стандартный Data Mining модуль, основанный на ассоциативных правилах (Microsoft Association Rules), не работает с непрерывными данными и не занимается дискретизацией данных. При создании модели столбец БД, содержащий, например, вещественные значения должен быть помечен, как дискретизируемый (descretized). Задача по разделению диапазона вещественных значений на интервалы, возложена на аналитические службы самой СУБД. Это означает, что данные разбиваются на группы без учета того, какой алгоритм далее будет с ними работать. Пользователь не может повлиять на количество этих групп или диапазонов для некоторого анализируемого признака. Также алгоритм Microsoft Association Rules не может использовать такое важное свойство вещественных данных, как их упорядоченность.

Предлагаемая реализация Data Mining модуля применяет схему кластеризации уже после получения метаданных о данных и может определять параметры разбиения диапазонов значений признаков на интервалы в зависимости от размера данных, количества анализируемых признаков, атрибутов признаков и т. д. Таким образом, алгоритм может адаптироваться к задачам разного размера и использовать свойство упорядоченности значений для данных как непрерывного, так и дискретного типа, что позволяет с большей точностью решать задачи прогнозирования.

*Форма представления закономерностей.* Предлагаемый нами подход Discovery оперирует закономерностями в виде логических IF – THEN формул, называемых *правилами*:

$$A_1 \& \dots \& A_n \rightarrow B,$$

где  $A_1, \dots, A_n, B$  – это предикаты, утверждения о том, что значение некоторой наблюдаемой величины принадлежит некоторому множеству значений (в дальнейшем просто предикат);  $B$  называется *целевой* (THEN) частью;  $A_1 \& \dots \& A_n$  называется *условной* (IF) частью.

Предполагается, что интервалы значений признаков, фигурирующие в закономерностях, будут определены заранее, на начальной стадии работы алгоритма и разбиты на характерные для этих величин подынтервалы с помощью алгоритма кластеризации. Подынтервалы не должны пересекаться и должны покрывать все допустимые значения соответствующего признака. Для этого могут применяться различные подходы. Таким образом, мы получим множество интервалов значений для каждого счетчика. Полученные подынтервалы назовем *атомарными интервалами*.

Очевидно, что для представления множества всех логических формул, выражающих зависимости между значениями наблюдаемых признаков, нужно ввести логическую операцию отрицания, чтобы все возможные формулы на языке логики первого порядка, не содержащие кванторов, могли быть представлены в виде комбинации правил. В нашем случае предлагается иное решение. Так как атомарные интервалы покрывают множество всех допустимых значений признака, то отрицание утверждения о принадлежности значения признака атомарному эквивалентно утверждению о принадлежности значения признака одному из оставшихся атомарных интервалов.

В нашем подходе для улучшения качества работы алгоритма используются предикаты, значение которых истинно тогда и только тогда, когда значение некоторого наблюдаемого признака принадлежит некоторому набору атомарных интервалов. Эти наборы атомарных интервалов назовем *комбинированными интервалами*. Очевидно, комбинированные интервалы могут пересекаться или даже целиком покрываться другими комбинированными интервалами. Это, в частности, позволяет использовать интервалы различной ширины в целевой части правил, что очень полезно в случаях, когда нельзя сделать однозначный прогноз, в какой из атомарных интервалов попадает значение некоторой наблюдаемой величины.

Приведенное выше правило также можно представить в виде

$$\neg A_1 \vee \dots \vee \neg A_n \vee B.$$

Очевидно, что при рассмотрении совокупности правил  $P_1, \dots, P_k$ , которым должна удовлетворять наблюдаемая система, ее можно представить конъюнкцией правил  $P_1 \& \dots \& P_k$  – конъюнктивной нормальной формой (КНФ). Также очевидно, что любая логическая закономерность между интервалами значений наблюдаемых величин может быть представлена в виде КНФ, а значит, может быть представлена в виде совокупности правил.

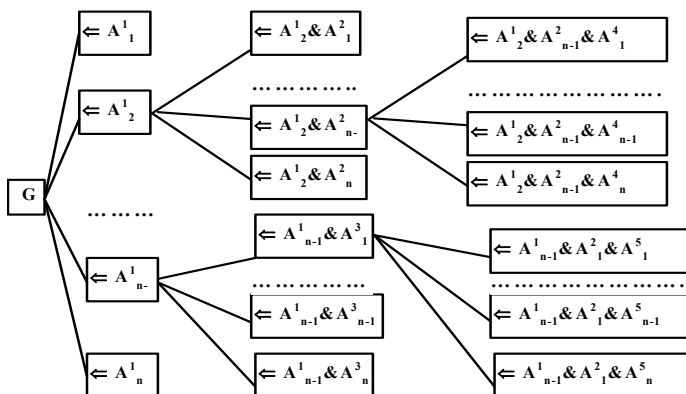
*Общая схема работы алгоритма поиска закономерностей.* В алгоритме поиска закономерностей используется методология семантического вероятностного вывода, позволяющего находить максимально специфические закономерности в данных. Тем не менее применять данную методологию в чистом виде не представляется возможным в первую очередь из-за требований к производительности, а также из-за ограниченности выборки, на которой производится обучение. Для этого применяются различные эвристики.

Под семантическим вероятностным выводом понимается такая последовательность правил  $C_1, C_2, \dots, C_n$ , что:

- 1)  $C_i = (A_1^i \& \dots \& A_{k_i}^i \Rightarrow G)$ ,  $i = 1, \dots, n$ ;
- 2)  $C_i$  – подправило правила  $C_{i+1}$ , т. е.  $\{A_1^i, \dots, A_{k_i}^i\} \subset \{A_1^{i+1}, \dots, A_{k_{i+1}}^{i+1}\}$ ;
- 3)  $\text{Prob}(C_i) < \text{Prob}(C_{i+1})$ ,  $i = 1, 2, \dots, n - 1$ , где *условная вероятность* (УВ) правила  $\text{Prob}(C_i) = \text{Prob}(G / A_1^i \& \dots \& A_{k_i}^i) = \text{Prob}(G \& A_1^i \& \dots \& A_{k_i}^i) / \text{Prob}(A_1^i \& \dots \& A_{k_i}^i)$ ;
- 4)  $C_i$  – *вероятностные законы* (ВЗ), т. е. для любого подправила  $C' = (A_1 \& \dots \& A_j \Rightarrow G)$  правила  $C_i$ ,  $\{A_1, \dots, A_j\} \subset \{A_1^i, \dots, A_{k_i}^i\}$  выполнено неравенство  $\text{Prob}(C') < \text{Prob}(C_i)$ ;
- 5)  $C_n$  – *сильнейший вероятностный закон* (СВЗ), т. е. правило  $C_n$  не является подправилом никакого другого вероятностного закона.

При применении этого определения к реальным данным необходимо проверять вероятностные неравенства некоторым статистическим критерием с определенной статистической значимостью. В качестве критериев статистической значимости применяются точный критерий Фишера и критерий Юла для уточняющего предиката [7; 8]. Другими словами, мы проверяем, насколько существенно этот признак в условной части влияет на целевую часть правила. При этом применение одного из критериев или обоих зависит от количества экспериментов, на которых истинна или ложна целевая часть правила при истинной или ложной условной части. Критерий Фишера показал хорошую эффективность на данных, когда соответствующие значения не велики, критерий Юла, наоборот, дает более адекватные оценки на больших объемах данных. Более того, вычисление значения точного критерия Фишера становится достаточно ресурсоемким процессом при увеличении числа экспериментов. Пороговые величины для применения того или иного критерия были установлены экспериментально.

В нашем алгоритме, при построении дерева семантического вероятностного вывода, для уточнения некоторого правила алгоритм всегда добавляет к его условной части только один предикат:



Таким образом, сильнейший вероятностный закон – это вероятностный закон, который не является подправилом никакого другого вероятностного закона, который можно получить добавлением одного предиката. Как показывают некоторые предварительные оценки и эксперименты, крайне редка ситуация, когда при добавлении в условную часть правила одновременно двух предикатов мы получаем вероятностный закон, удовлетворяющий критерию статистической значимости, но при этом любой из этих двух признаков сам по себе не уточняет исходное правило. Следовательно, данная эвристика почти не снижает количество и качество извлеченных из данных закономерностей при серьезном уменьшении пространства перебора.

Определим оценку условной вероятности (ОУВ) как частоту:

$$P(B / A_1 \& A_2 \& \dots \& A_{n-1}) = N(B / A_1 \& A_2 \& \dots \& A_{n-1}) / N(A_1 \& A_2 \& \dots \& A_{n-1}),$$

где  $N(B / A_1 \& A_2 \& \dots \& A_{n-1})$  есть число событий  $A_1 \& A_2 \& \dots \& A_{n-1} \& B$ ;

$N(A_1 \& A_2 \& \dots \& A_{n-1})$  есть число событий  $A_1 \& A_2 \& \dots \& A_{n-1}$ .

Параметры алгоритма, такие как минимальная оценка условной вероятности правила, пороговые значения критериев Фишера и Юла, максимальное число интервалов значений для признака и др., могут быть заданы пользователем перед началом процесса обучения экземпляра анализирующего модуля. Также будут учитываться атрибуты наблюдаемых признаков: входные (input), прогнозируемые (predict) и только прогнозируемые (predict only). Эти атрибуты указывают, в какой части правила (условной или целевой) может находиться предикат, относящийся к соответствующему столбцу: только в условной, в обеих или только в целевой соответственно.

Результатами работы алгоритма являются:

- дерево семантического вероятностного вывода для каждого целевого предиката;
- множество вероятностных и сильнейших вероятностных законов этих деревьев;
- *максимально специфический закон* (МСЗ) для каждого целевого предиката, определяемый как сильнейший вероятностный закон, обладающий наибольшей условной вероятностью среди других сильнейших вероятностных законов дерева вывода этого предиката.

Множество всех максимально специфических законов обладает таким важным свойством, как потенциальная непротиворечивость [5].

*Построение дерева семантического вероятностного вывода.* Процесс построения дерева семантического вероятностного вывода является достаточно ресурсоемким процессом. За счет правильного подбора пороговых значений критериев статистической значимости можно существенно уменьшить пространство перебора, а значит, и время работы алгоритма. Тем не менее выбор оптимальных значений сильно зависит от анализируемых данных, и чаще всего они подбираются экспериментально. Поэтому, кроме описанной в предыдущем параграфе идеи уточнения правила на один признак, использовались и другие методы повышения производительности на этапе поиска закономерностей.

В случае построения полного дерева семантического вероятностного вывода для некоторого целевого предиката (утверждения о том, что признак принадлежит некоторому комбинированному интервалу) на множестве предикатов мы можем ввести отношение порядка и осуществлять процесс вывода, добавляя в правила только предикаты с порядковым номером, большим, чем у последнего предиката в условной части правила. Очевидно, что последний предикат в условной части правила всегда будет иметь наибольший порядковый номер. В случае, когда семантический вероятностный вывод будет проводиться для сформулированного класса гипотез, мы можем разделить IF-часть каждого правила на два подмножества: предикаты, фигурирующие в исходной гипотезе, и предикаты, не фигурирующие в исходной гипотезе, и рассматривать порядковые номера только предикатов из второго подмножества.

Затем предлагается использовать поуровневую схему генерации правил, для того чтобы алгоритм мог быстро проверить, является ли новое правило вероятностным законом. Идея в следующем: сначала находим все возможные вероятностные законы с одним предикатом в условной части, затем с двумя, тремя и т. д. Значит, для проверки, является ли уточняющее

правило вероятностным законом, не нужно проверять условную вероятность всех подправил, а только подправила с предыдущего уровня в дереве семантического вероятностного вывода. Если какое-либо подправило отсутствует или его условная вероятность больше, чем условная вероятность уточняющего, то уточняющее правило – не вероятностный закон; иначе это вероятностный закон. Имея порядок на множестве предикатов, мы можем утверждать, что если для правила уровня  $N$  не найдено уточняющих правил уровня  $N + 1$ , то это правило – сильнейший вероятностный закон.

Другим преимуществом данной схемы является то, что она позволяет нам не хранить статистику для правил предыдущих уровней, что существенно сокращает затраты оперативной памяти на этапе генерации правил. Экономия памяти достигается благодаря тому, что вычисление условной вероятности и значения критериев статистической значимости Фишера и Юла происходит на основе статистики для уточняемого правила и статистики для добавляемого признака. Так как понятие вероятностного закона определяется относительно добавления одного предиката, то необходима проверка статистической значимости только каждого отдельного признака в соответствующем подправиле, а каждое такое подправило принадлежит предыдущему уровню дерева семантического вероятностного вывода. Вычисление условной вероятности также осуществляется на основе статистики для правил предыдущего уровня и статистики для добавляемого признака и реализуется с помощью операций над множествами: пересечение, объединение, дополнение и т. д. Таким образом, для вычислений необходима только статистика для правил, принадлежащих предыдущему уровню дерева и 1-му уровню, содержащему статистику для отдельных признаков. Помимо экономии памяти, применяемая схема позволяет производить вычисления значительно быстрее, чем прямое вычисление условной вероятности и значения критериев статистической значимости.

Как показало предварительное тестирование алгоритма, подсчет статистики для каждого правила является критическим для производительности программы местом на большинстве тестов. Поэтому для ускорения счета была применена следующая техническая оптимизация. Статистика для каждого предиката, условной и целевой частей правила представлена в виде массива чисел, разрядность которых будет совпадать с разрядностью процессора, операции над статистикой проводятся с помощью побитовых логических операций (bitwise operations). Это позволяет увеличить производительность операций по вычислению статистики для уточняющих правил, в то время как описанные выше эвристики позволяют снизить количество правил и признаков, для которых необходимы такие вычисления.

*Прогнозирование.* Основная функциональность OLAP – это построение прогнозов и статистических распределений для анализируемых данных. При этом пользователю SQL Server может быть нужна как статистика по уже проанализированным данным, так и определение наиболее вероятного распределения некоторого признака по набору значений других признаков.

Для прогнозирования используется специальный язык запросов, основанный на синтаксисе языка SQL. Формально строится select-запрос к Data Mining модели. Причем запрашиваться может как значение определенного признака, так и некоторая функция, специфичная для определенного типа данных и Data Mining модели. Результатом же функции может быть не только значение простого типа, но и таблица (например, для функций типа «гистограмма»). Результат прогнозирования строится по набору закономерностей, найденных на этапе анализа данных, и набору входных данных – значений некоторых признаков.

В самом простом случае пользователь на этапе анализа определяет набор признаков, для которых будет осуществляться предсказание, и набор признаков, по которым необходимо определять значения интересующих признаков. В этом случае для определения наиболее вероятных значений целевых признаков система должна применить найденные максимально специфические закономерности к входным данным. Наиболее вероятное значение каждого признака будет определено как середина интервала, который является значением целевого признака в соответствующем максимально специфическом правиле. В этом случае значения интересующих пользователя признаков определяются однозначно.

В более сложном случае входных данных может быть не достаточно для применения максимально специфических закономерностей. Тогда рассматриваются правила со всего

полученного дерева семантического вероятностного вывода, т. е. все правила, являющиеся вероятностными законами. Затем берутся все правила, где в целевой части стоит предикат, утверждающий, что нужный признак имеет некоторое атомарное значение, т. е. один интервал (а не блок подряд идущих атомарных интервалов или отрицание атомарного интервала). Далее среди них находится правило с наибольшей вероятностью, такое что:

- те признаки, значения которых заданы на вход, могут не встречаться вообще в условной части этого правила или встречаться со значением, соответствующим набору интервалов, в который должно попадать значение этого признака, заданное на вход;
- те признаки, значения которых не заданы на вход, не должны встречаться в условной части правила.

В случае, когда однозначный выбор правила сделать трудно, нужно брать правила, где целевой признак принимает не атомарное значение, а значение, соответствующее интервалу, состоящему из 2 подряд идущих атомов. Если и здесь ситуация не прояснится, то для 3 и т. д., т. е. необходимо расширять целевой интервал с целью увеличения условной вероятности. В данном случае используется специальная оценочная функция, которая по набору лучших правил для каждого значения признака и статистики для этих правил определяет, однозначно ли можно выбрать доминирующее правило, по которому и получается прогнозируемое значение.

В случае, когда результатом прогноза должно быть одно значение, оно получается как середина интервала, который является значением целевого признака в доминирующем правиле. В случае же, когда результатом прогноза должна быть гистограмма, алгоритм выделяет наиболее характерные интервалы значений по набору наиболее подходящих правил и возвращает статистическое распределение признака.

### **Описание технической части**

*Стандарты Data Mining. Knowledge Discovery in Data Bases and Data Mining (KDD&DM)* является достаточно новой и постоянно развивающейся областью информационных технологий. Однако уже сейчас мы можем наблюдать появление ряда стандартов, пытающихся упорядочить и согласовать достижение всей индустрии анализа данных за последнее десятилетие.

Стандарт OLE DB for Data Mining – первый индустриальный стандарт для Data Mining, разработанный компанией Microsoft. Он использует SQL-подобный язык запросов и специализирует структуры данных так, чтобы Data Mining потребители могли взаимодействовать с функциональностью Data Mining провайдера.

PMML – язык разметки моделей прогнозирования (Predictive Model Markup Language), разработан под руководством Data Mining Group. PMML построен на основе языка XML и служит общим, независимым от разработчика и конкретной реализации продукта DM, некоммерческим интерфейсом, который дает возможность приложениям различных производителей обмениваться полученными моделями и закономерностями, не раскрывая при этом коммерческую собственность каждого из них.

Среди приоритетных требований, предъявляемых к разрабатываемому модулю, были сформулированы требования соответствия перечисленным выше стандартам, ввиду того что именно благодаря им возможна интеграция с основными компонентами аналитических служб Microsoft SQL Server 2005. Такими компонентами, например, являются средства визуализации результатов работы алгоритма, конструкторы для создания и управления Data Mining моделями и т. д.

*Архитектура аналитических служб Microsoft SQL Server 2005.* Службы Microsoft SQL Server 2005 Analysis Services (SSAS) используют как серверные, так и клиентские компоненты для предоставления приложениям бизнес-аналитики функций оперативной аналитической обработки данных (OLAP) и Data Mining.

Серверная часть служб SSAS реализована в виде службы Microsoft Windows. Служба состоит из компонент безопасности, компоненты прослушивания XML for Analysis, компоненты обработчика запросов и множества других внутренних компонент,



выполняющих функции по анализу инструкций, получаемых от клиентов, управлению метаданными, обработке транзакций и вычислений, управлению ресурсами сервера и т. д.

Data Mining модуль напрямую взаимодействует со следующими сервисными компонентами службы: Algorithm Manager, Case Processor и DMX Query Processor. Algorithm Manager отвечает за управление жизненным циклом Data Mining модели, задание параметров модели и диспетчеризацию ресурсов. Case processor служит для получения данных из анализируемого источника, предварительной их обработки и передачи данных нашему модулю. DMX Query Processor отвечает за обработку запросов к Data Mining модели, в частности разбор инструкций SQL-подобного языка, на котором строятся запросы к модулю для получения прогнозов и статистических распределений.

Клиенты связываются с аналитическими службами, которые рассматриваются как веб-служба, с помощью SOAP по протоколу TCP/IP или HTTP через службы IIS. SOAP позволяет Data Mining провайдером предоставлять клиентам набор методов, которые могут быть задействованы с помощью XML сообщений. Такой способ взаимодействия, реализуемый аналитическими службами, описан в спецификации стандарта XML for Analysis (XMLA) – стандарта, предложенного группой ведущих разработчиков средств Business Intelligence для организации взаимодействия со средствами OLAP и Data Mining.

Также клиентские приложения Win32 могут подключаться к серверу аналитических служб с помощью интерфейсов OLE DB для OLAP объектной модели Microsoft ActiveX (ADO) для языков автоматизации модели COM, например Visual Basic. Приложения, написанные на языках платформы .NET, могут подключаться к серверу аналитических служб с помощью ADO MD.NET.

*Интеграция с аналитическими средствами Microsoft SQL Server 2005.* С технологической точки зрения дополнительный Data Mining модуль представляет собой набор COM компонент, реализующих стандарт OLE DB for Data Mining. Для реализации модуля корпорация Microsoft предоставляет специальный интерфейс прикладного программирования, использование которого позволяет организовать взаимодействие между реализуемым модулем и серверными компонентами аналитических служб, которые в свою очередь предоставляют OLAP и Data Mining функции другим приложениям.

Среди клиентских приложений стоит особо выделить Business Intelligence Development Studio. Это стандартное средство для решения OLAP и Data Mining задач. Являясь стандартным средством, входящим в поставку SQL Server и обеспечивающим доступ ко всем основным функциям Data Mining моделей, Business Intelligence Development Studio является тем приложением, механизмы работы которого, непременно должны быть учтены в ходе разработки дополнительного Data Mining модуля. С точки зрения программной архитектуры, Business Intelligence Development Studio – это набор дополнительных компонент для Microsoft Visual Studio 2005, позволяющий создавать проекты для решения задач бизнес-аналитики. Расширения включают в себя инструментарий для создания и управления Data Mining моделями: конструкторы, средства для подготовки и просмотра данных, средства визуализации результатов анализа и т. д.

Каждый дополнительный модуль обладает свойством самоописания. Модуль предоставляет службе метаданные, описывающие основные его свойства: возможность масштабирования модели, набор типов данных, с которыми он работает, набор поддерживаемых Data Mining функций, а также список конфигурируемых параметров алгоритма, например пороговые величины для критериев отбраковки закономерностей.

Различные Data Mining алгоритмы могут работать с разными типами данных. Среди них есть дискретные (discrete), дискретизируемые (descretized), непрерывные (continuous), номинальные (key), циклические (cyclical), время (key time) и др. Эти типы не связаны напрямую с типами, которые используются СУБД для хранения данных. При создании Data Mining модели пользователь сам указывает, как должен интерпретироваться тот или иной столбец БД. При работе с каждым из типов должна учитываться его специфика.

Другим важным свойством модуля является предоставление Data Mining функций аналитическим службам SQL Server. Для этого модулем реализуется набор COM интерфейсов, через которые он получает данные, необходимые параметры моделей, а также запросы на обработку данных, построение прогнозов и т. д.

Третье важное свойство модуля – это интеграция со средствами визуализации результатов анализа. Здесь возможна как реализация собственного модуля визуализации, так и использование стандартных инструментов визуализации [9]. Разрабатываемый Data Mining модуль ориентирован на работу с Microsoft Mining Content Viewer, инструментом, позволяющим просматривать закономерности в некотором унифицированном табличном виде. Каждая строка таблицы описывает закономерность в данных, в нашем случае правило, и содержит текстовое представление закономерности, XML-представление, данные статистики, важность с точки зрения алгоритма анализа и т. д.

### Анализ результатов

Проведено тестирование Data Mining модуля с целью сравнения качества анализа разрабатываемого алгоритма со стандартными средствами Data Mining SQL Server. Для сравнения был выбран Data Mining провайдер, основанный на поиске ассоциативных правил (Microsoft Association Rules), ввиду того что данный алгоритм также представляет найденные закономерности в виде IF-THEN правил и похожей схемы предварительной обработки данных.

Первый тест проводился на базе данных «Sample.mdb», входящей в поставку пакета для разработки Data Mining модулей и являющейся стандартным тестом для отладки и оценки качества Data Mining алгоритмов. В таблице БД находятся значения трех величин, связанных между собой линейной закономерностью с шумами. Размер таблицы достаточно мал – всего 29 строк. Для оценки качества работы алгоритма был проведен анализ данных таблицы обоими алгоритмами, и затем, на основе найденных закономерностей, выполнено предсказание значения одного из признаков по значениям двух других признаков на тех же данных. Нами сравнивалось среднеквадратичное отклонение (СО) предсказываемых значений признака от реальных значений этого признака на всей таблице. В результате точность предсказания нашего алгоритма получилась примерно на 7–8 % лучше.

Второй тест проводился на БД значительно большего размера. В качестве анализируемых данных использовались данные статистики, полученной с сервера, установленного в узле сети передачи данных СО РАН и предназначенного для раннего обнаружения вредоносных воздействий на сеть извне, проявлений аномального поведения компьютеров абонентов сети с целью обеспечения безопасности сети. Для анализа использовалась статистика за «чистый» период, когда сеть функционировала нормально, и за период, когда в сети наблюдалась аномальная активность. Полученные данные были преобразованы в одну таблицу реляционной базы данных. Каждая запись этой таблицы описывает процесс обмена данных между двумя хостами по определенному протоколу за некоторый интервал времени. Причем один из участников обмена должен принадлежать сети передачи данных СО РАН, а другой быть внешним по отношению к этой сети. Таким образом, рассматриваются только данные, прошедшие через внешние интерфейсы сети передачи данных СО РАН, транзитный сетевой трафик при этом также не учитывается. Адресное пространство сети СО РАН было поделено на 15 сетей в соответствии с их масками подсети. Признаки представленные в таблице:

- идентификатор протокола;
- идентификаторы сети отправителя;
- идентификаторы сети получателя;
- сетевой порт отправителя;
- сетевой порт получателя;
- суммарный размер пакетов за определенный временной интервал;
- период, в который получены данные (аномальный или «чистый»).

Размер таблицы составлял примерно 750 000 записей.

В данном тесте целевым признаком была принадлежность записи таблицы аномальному периоду. Результаты теста показали, что используемая реализация системы Discovery с вероятностью около 71 % верно определяет значение целевого признака. Этот результат лучше, чем результат алгоритма Microsoft Association Rules, который показал точность около 64 %.

## Список литературы

1. *Freitas A. A.* Understanding the Crucial Differences Between Classification and Discovery of Association Rules. Pontificia Universidade Catolica – Parana Dept. of Computer Science, 2000.
2. *Tang Z., MacLennan J.* Data Mining with SQL Server 2005. Wiley Publishing, Inc., 2005. 483 с.
3. *Kovalerchuk B., Vityaev E.* Correlation of Complex Evidences and Link Discovery // Proc. of the Fifth International Conference on Forensic Statistics. Venice, 2002.
4. *Kovalerchuk B., Vityaev E.* Data Mining in Finance: Advances in Relational and Hybrid Methods. Kluwer, 2000. 308 с.
5. *Витяев Е. Е.* Извлечение знаний из данных. Компьютерное познание. Модели когнитивных процессов. Новосибирск, 2006. 293 с.
6. *Kovalerchuk B.* Comments on the Microsoft Draft Standard (Specification) for Data Mining / Dept. of Computer Science. Central Washington University, 2000.
7. *Кендалл М. Дж., Стьюарт А.* Статистические выводы и связи. М.: Наука, 1973. 899 с.
8. *Закс Ш.* Теория статистических выводов. М.: Мир, 1975. 776 с.
7. *Halpern J. Y.* An analysis of first-order logic of probability. Artificial Intelligence. 1990. С. 311–350.
9. *Vityaev E., Kovalerchuk B.* Visual Data Mining with Simultaneous Rescaling // Visual and Spatial Analysis. Advances in Data Mining, Reasoning and Problem Solving. Springer, 2004. P. 371–385.

*Материал поступил в редколлегию 16.06.2008*

**D. S. Durdin, E. E. Vityaev**

### **Data Mining Plug-In for Microsoft SQL Server 2005 Based on the Discovery System**

Microsoft Company integrates Data Mining methods and Microsoft SQL Server 2005. Now SQL query to Microsoft SQL Server 2005 may include not only the sample formation but also run some Data Mining method on this sample that is incorporated in Microsoft SQL Server 2005. Microsoft Company produced an open source of API for Data Mining plug-ins development by Data Mining companies. These plug-ins may be automatically integrated in SQL Server 2005. The purpose of this work is plug-in development that realizes the Data Mining system Discovery. This system implements the Relational Data Mining approach for knowledge discovery. It uses logical (relational) representation of the data information and discovers regularities based on the special semantic probabilistic inference. Relational Data Mining approach and system Discovery were adapted for the Microsoft SQL Server 2005 conditions. In result, the realized plug-in may discover knowledge from data with statistical significance and predict with high accuracy.

*Keywords:* DBMS, SQL Server, plug-in, Data Mining, scientific discovery, prediction.