Разработка Нечеткого OLAP-Куба в Хранилище Данных Системы Поддержки Принятия Решений

Гюльнара Набибекова
Институт Информационных Технологий НАНА
gulnarara58@mail.ru

***Аннотация*— В статье показана интеграция технологии OLAP (Online Analytical Processing) и нечеткой логики при разработке нечеткого OLAP-куба в поликубической OLAP-модели в системе поддержки принятия решений. С этой целью фаззифицированы измерения OLAP-куба. Показано формирование нечеткого среза в результате нечеткого запроса. Также представлено практическое применение данного подхода в системе поддержки принятия решений при решении задачи управления кадровыми ресурсами.**

Ключевые слова— система поддержки принятия решений, хранилище данных, OLAP, OLAP-кубЮ, нечеткое множество, функция принадлежности, терм-множество, лингвистическая переменная, нечеткий запрос, нечеткий срез, индекс соответствия срезу

I. ВВЕДЕНИЕ

В процессе принятия решений можно столкнуться с множеством нечетких задач, вследствие чего в запросах к Хранилищам данных (ХД), которые формулирует аналитик, часто имеют место неточности и неопределенности. Кроме того, для лиц, принимающих решения, важен и полезен бывает не сам результат запроса, а то, является ли этот результат хорошим, средним или плохим, то есть им важна качественная сторона результата [1,2]. Все это требует, чтобы система поддержки принятия решений (СППР) могла использовать нечеткие и неопределенные рассуждения и поддерживала соответствующую технологию представления знаний [2]. В этом случае ХД должно обеспечивать ответы на запросы, которые использую лингвистические термины, являющиеся терм-множествами соответствующей лингвистической переменной.

Для решения данной задачи необходимо выполнить фаззификацию измерений OLAP-куба и затем в результате нечеткого запроса сформировать нечеткий срез.

II. ФАЗЗИФИКАЦИЯ ИЗМЕРЕНИЙ OLAP-КУБА

В настоящее время для комплексного многомерного анализа больших объемов накопленной в ХД информации активно применяется технология OLAP [3]. Технология OLAP основана на представлении данных ХД в виде многомерной модели – гиперкуба, или OLAP-куба, содержащего одно или более измерений. Для фаззификации измерений OLAP-куба обратимся к получившему в настоящее время широкое распространение аппарату теории нечетких множеств в задачах поиска информации.

Нечеткое множество, включая функцию принадлежности, являющуюся его характеристикой, может быть задано экспертом. Но в некоторых случаях функцию принадлежности удобнее задать аналитической формулой и графически. Для задания функций принадлежности существуют различные типовые формы кривых, среди которых выберем треугольную и трапецеидальную кусочно-линейные формы. В общем случае треугольная функция принадлежности определяется тремя числами, а трапецеидальная – четырьмя, хотя их количество может и меняться в зависимости от условий задачи.

Для определения чисел, с помощью которых будет задана функция принадлежности, применим кластеризацию, на основании подхода, представленного в [1], согласно которому центры кластеров – медоиды, полученные в результате кластеризации соответствующих измерений, служат для определения функции принадлежности.

С целью выбора эффективного алгоритма кластеризации были рассмотрены неирархические алгоритмы, основанные на методе разбиения, а именно на методе k-medoids, так как медоид – это центр кластера, который принадлежит данному кластеру, что является главным условием задачи. Кроме того, методы k-medoids являются устойчивыми к наличию выбросов (outliers), т.е, точек, которые достаточно далеки от остальных точек, и могут работать достаточно эффективно с большими наборами данных.

Алгоритмы, основанные на методе разбиений, проходят два основных этапа [4]:

- начальный шаг, на котором k объектов выбраны в качестве медоидов;
- оценочный шаг, на котором происходит попытка минимизировать целевую функцию, обычно основанную на сумме общего расстояния между невыбранными объектами (значениями) и их медоидами, т.е.:

DOI: 10.25045/NCSoftEng.2017.21

$$D(r,s)=\sum_{j=1}^{n} d(r_i,s_j)$$

где $s_j \in S$ (S – множество объектов (значений) для кластеризации) и $d(r_i,s_i)<d(r_c,s_j)$, $\forall r_i,r_c \in R$ (R – множество объектов (значений) из S, выбранных в качестве медоидов), $r_i \neq r_c$. Чем меньше сумма расстояний между медоидом и всеми другими объектами соответствующего ему кластера, тем лучше кластеризация.

Были рассмотрены три известных алгоритма, основанных на методе k-medoids: PAM (Partitioning Around Medoids), CLARA (Clustering LARge Applications) и CLARANS (Clustering Large Applications based on RANdomized Search) [5, 6].

PAM является одним из первых алгоритмов, основанных на методе k-medoids. В работах [7, 8] представлен основной принцип процесса кластеризации PAM. заключающийся в переборе всех объектов, не являющихся медоидами, для вычисления расстояния от них до выбранных медоидов. Действие алгоритма PAM приводит к высокому качеству кластеров, но поскольку он пробует все возможные комбинации, он эффективен для небольших наборов данных. И в связи с его вычислительной сложностью его применение непрактично для кластеризации больших наборов данных.

Вычислительная сложность алгоритма PAM явилась мотивацией для разработки алгоритма CLARA – алгоритма кластеризации на основе выборки [7]. CLARA выделяет из множества данных несколько выборок данных, применяет PAM на каждой выборке и находит ее медоиды. Поскольку выборка произведена случайным образом, то медоиды выборки можно считать медоидами всего набора данных. Чтобы выбрать лучшие приближения, CLARA создает несколько выборок и на выходе выдает лучшую кластеризацию. В работе [7] также экспериментально показано, что пять выборок размером (40+2k), где k – количество кластеров, дают удовлетворительные результаты.

CLARANS был разработан в рамках пространственного анализа данных. При поиске лучшего медоида на шаге оценки CLARANS случайным образом выбирает объекты из ($n-k$) объектов (n – количество объектов в множестве, k – заданное количество кластеров, или количество медоидов). Количество объектов, перебираемых на этом этапе, ограничены параметром *maxneighbor*, заданным пользователем. Если после *maxneighbor* попыток не будет найдено лучшее решение, то локальный оптимум считается достигнутым. Процедура продолжается до тех пор, пока не найдено *numloc* локальных оптимумов.

Алгоритм CLARANS использует стратегию рандомизированного (случайного) поиска для облегчения кластеризации большого количества данных. Наличие большого количества данных позволяет гарантированно достичь *numloc* локальных оптимумов [8].

В результате исследования вышеупомянутых алгоритмов на предмет их достоинств и недостатков, для получения чисел, с помощью которых затем будет выполнена фаззификация, был выбран алгоритм CLARA, являющийся наиболее приемлемым для использования в данной задаче, с точки зрения сложности вычислений, размера множеств и затраченного времени.

Предположим, A является множеством измерений из всех измерений которые надо фаззифицировать.

Для каждого $j \epsilon A$ применим алгоритм CLARA и найдем в нем k медоидов $a_{1j}, a_{2j}, \ldots a_{kj}$ для k кластеров.

Пусть J - множество значений j-го измерения;

$F_{ii}(x)$ - функция принадлежности для получения значений принадлежности всех значений j-го измерения в i-ом нечетком множестве. Рассмотрим случай, когда количество терм-множеств равно 3, т.е. i=3. Отметим, что второе терм-множество считается средним, т.е. расположенным между первым и последним терм-множествами.

Для всех x є J выполняется:

1) для первого нечеткого множества (т.е. при i=1)

$F_{1j}(x)$=1, если $x \leq a_{1j}$;

$F_{1j}(x)=(a_{2j}-x)/(a_{2j}-a_{1j})$, если $a_{1j}<x<a_{2j}$;

$F_{1j}(x)$=0, если $x \geq a_{2j}$.

2) для второго нечеткого множества (т.е. при i=2)

$F_{2j}(x)$=0 если $x \leq a_{1j}$;

$F_{2j}(x)=(x-a_{1j})/(a_{2j}-a_{1j})$, если $a_{1j}<x<a_{2j}$;

$F_{2j}(x)$=1, если $x=a_{2j}$;

$F_{2j}(x)=(a_{3j}-x)/(a_{3j}-a_{2j})$, если $a_{2j}<x<a_{3j}$;

$F_{2j}(x)$=0, если $x \geq a_{3j}$.

3) для третьего нечеткого множества (т.е. при i=3)

$F_{3j}(x)$=0, если $x \leq a_{2j}$;

$F_{3j}(x)=(x-a_{2j})/(a_{3j}-a_{2j})$, если $a_{2j}<x<a_{3j}$;

$F_{3j}(x)$=1, если $x \geq a_{3j}$.

III. ФОРМИРОВАНИЕ НЕЧЕТКОГО СРЕЗА

В результате нечеткого запроса формируется нечеткий срез, состоящий из ячеек куба, соответствующих условиям запроса. Но для формирования итогового среза учитывают также и так называемый индекс соответствия срезу CI (Compliance Index), где $CI \in [0,1]$, который определяет аналитик [9].

Пусть имеется запрос:

DOI: 10.25045/NCSoftEng.2017.21

$$\{(L_1 = L_{1j}, (j = \overline{1, k_1})) \bigcup (L_2 = L_{2j}, (j = \overline{1, k_2})) \bigcup .... \bigcup (L_n = L_{nj}, (j = \overline{1, k_n}))\},$$

где $L_1, L_2, \ldots, L_n$ – лингвистические переменные, а L_{1j}, $(j = \overline{1, k_1})$, L_{2j}, $(j = \overline{1, k_2})$,…, L_{nj}, $(j = \overline{1, k_n})$ – соответствующие им терм-множества.

В каждой записи ХД, $\forall x \in L_i$, где $i = \overline{1, n}$, то есть для любого x лингвистической переменной, участвующей в запросе, найдем степень принадлежности $\mu_{L_{ij}}(x)$ этого x терм-множеству L_{ij}, участвующему в запросе.

Находим степень принадлежности каждой записи итоговому срезу, то есть ее индекс соответствия срезу, по формуле:

$$CI = min(\mu_{L_{mn}}(x)).$$

Для CI вводятся условия, например:

если $CI \leq 2$, то соответствие срезу «слабое»;

если $2 < CI < 5$, то соответствие срезу «среднее»;

если $CI \geq 5$, то принадлежность срезу «высокая».

IV. ПРАКТИЧЕСКОЕ ПРИМЕНЕНИЕ

В качестве примера создания нечеткого OLAP куба в поликубической OLAP-модели рассматривается подзадача управления кадровыми ресурсами в СППР, разработанной с целью определения степени интеграции стран. Данная СППР является многоцелевой, то есть помимо решения основной задачи, она может решать и побочные, в том числе и подзадачу управления кадровыми ресурсами, включающую решение таких вопросов, как анализ кадровых ресурсов, их размещение, обучение, планирование продвижения сотрудников и т.д. При решении такой подзадачи для лиц, принимающих решение, при формировании запроса к ХД часто бывает важен не конкретно возраст сотрудника, а то, является он *молодым, средним* или *пожилым*, или, например, не конкретно стаж работы, а является он *малым, средним* или *продолжительным*. Кроме того, его может интересовать не точная дата мероприятия, в котором сотрудник принимал участие, а то, когда произошло это событие - *давно* или *недавно*.

Пусть с целью создания нечеткого OLAP-куба на основании соответствующей витрины данных из всего множества измерений надо фаззифицировать 3 измерения – ВОЗРАСТ, СТАЖ РАБОТЫ и ДАТА

В результате фаззификации измерения СТАЖ РАБОТЫ и ВОЗРАСТ становятся лингвистическими переменными, а полученные 3 кластера станут тремя терм-множествами. Для измерения СТАЖ РАБОТЫ это {малый, средний, продолжительный}, а для измерения ВОЗРАСТ это {молодой, средний, пожилой}.

На рис.1 и рис.2 дано графическое изображение терм-множеств соответствующих измерений. Отметим, что номер j следует закреплять за соответствующим измерением. В данном случае j=1 – измерение “ВОЗРАСТ”, j =2 – измерение “СТАЖ РАБОТЫ”.

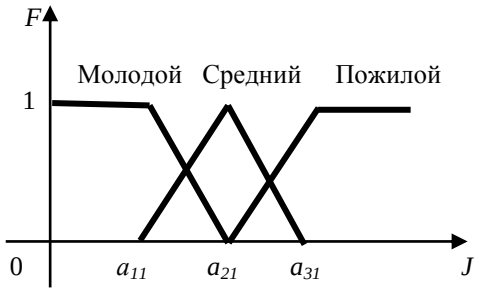

Рис. 1. Графическое изображение лингвистической переменной “ВОЗРАСТ

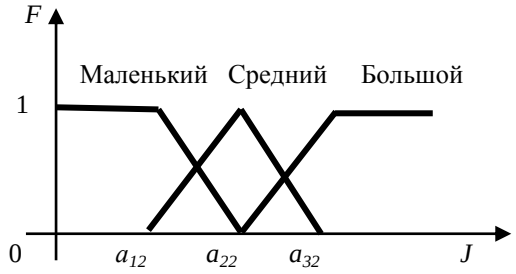

Рис. 2. Графическое изображение лингвистической переменной “СТАЖ РАБОТЫ”

Измерение ДАТА будет разбит на 2 терм-множества (*недавно и давно*), поэтому для его фаззификации применим алгоритм CLARA, задав количество кластеров k=2. В результате будут получены 2 кластера с центрами (медоидами) в точках a_{13} и a_{23}, с помощью которых будет задана функция принадлежности.

После выполнения фаззификации будут получены 2 терм-множества. На рис. 3 показано графическое изображение лингвистической переменной “ДАТА”.

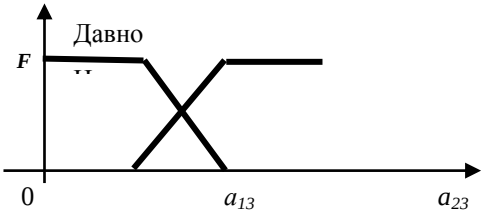

Рис. 3. Графическое изображение лингвистической переменной “ДАТА”.

В результате фаззификации измерений куба рассматриваемой СППР можно выполнять нечеткие запросы, используя термины *молодой, зрелый, пожилой* (касательно возраста), *малый, средний, продолжитель-ный* (касательно стажа работы), *недавно и давно* (касательно даты мероприятия).

DOI: 10.25045/NCSoftEng.2017.21

Приведены примеры типовых нечетких запросов, которые могут сыграть не последнюю роль в управления кадровыми ресурсами:

1. *Молодые* сотрудники, посетившие *недавно* Турцию;

2. Сотрудники, имеющие *продолжительный* стаж работы, которые посетили *недавно* Турцию в связи с мероприятиями в медицинской сфере;

3. Страны, где *молодые* сотрудники прошли *недавно* тренинги в военной сфере и т.д.

ЗАКЛЮЧЕНИЕ

Данная СППР реализована для персональных компьютеров, работающих в среде Windows XP, Windows 7, Windows 8 и т.д., допускающей сетевое многопользовательское использование технологии клиент-сервер. В качестве платформы используется система управления базами данных MS SQL сервер [10]. Среда реализации OLAP - Microsoft Visual Studio 2008 (Analysis Services). Клиентская часть реализована на языке программирования Object Pascal [11] с применением среды разработки приложений Delphi 2010 [12] и предназначена для обращения к серверу, обработки и представления полученных данных. Используемый язык запросов – T-SQL [13].

ЛИТЕРАТУРА

[1] K. Kumar et al. Fuzzy OLAP Cube for Qualitative Analysis / 3rd International Conference on Intelligent Sensing and Information Processing (ICISIP), 2005, p. 290 - 295. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1529464&tag=1

[2] Sh. Wang. Application of Decision Support System in E-government / International Conference “Management and Service Science”, 2009, (MASS'09). http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5302532

[3] E. F. Codd, S. B. Codd, C.T. Salley. Providing OLAP (Online Analytical Processing) to User-Analysts: An IT Mandate. San Jose: Codd & Date, Inc., 1993, p. 31. www.minet.uni-jena.de/dbis/lehre/ss2005/sem_dwh/lit/Cod93.pdf

[4] M.-C. N. Barioni et al. An efficient approach to scale up k-medoid based algorithms in large databases. XXI Brazilian Simposium on Databases. 2006. www.lbd.dcc.ufmg.br:8080/colecoes/sbbd/2006/018.pdf

[5] L. Kaufman, P. J. Rousseeuw Finding Groups in Data: an Introduction to Cluster Analysis. John Wiley & Sons, 2005,

[6] R.T. Ng, J. Han. Clarans: A method for clustering objects for spatial data mining // IEEE Transactions on Knowledge and Data Engineering (TKDE), 2002, 14(5), pp. 1003-1016.

[7] Г. Набибекова. Об одном методе фаззификации атрибутов хранилища данных в системах поддержки принятия решений в сфере внешней политики // Информационные технологии. №1, 2014, с.36-41.

[8] Г. Набибекова Об одном подходе к разработке нечеткого OLAP-куба в системах поддержки принятия решений в сфере внешней политики / Материалы IV Международной научно-практической конференции “Проблемы кибернетики и информатики” PCI2012, 2012, с.86-89.

[9] Н. Б. Паклин, В. И. Орешков. Бизнес-аналитика: от данных к знаниям. Учебное пособие, Питер, 2013, 703 с.

[10] R. Dyer Learning MySQL and MariaDB: Heading in the Right Direction with MySQL and MariaDB. O'Reilly Media, 2015, 408 p.

[11] M. Cantu Object Pascal Handbook. CreateSpace Independent Publishing Platform, 2015, 552 p.

[12] N. Hodges. More Coding in Delphi. Nepeta Enterprises, 2015, 246 p.

[13] I. Ben-Gan. T-SQL Fundamentals. Microsoft Press, 2016, 464 p.

DOI: 10.25045/NCSoftEng.2017.21