УДК 004.9

С. И. Богучарский,
С. В. Машталир, канд. техн. наук

ИЕРАРХИЧЕСКАЯ АГЛОМЕРАТИВНАЯ КЛАСТЕРИЗАЦИЯ В БАЗАХ ДАННЫХ МУЛЬТИМЕДИА

Аннотация. *В работе рассматриваются методы иерархической агломеративной кластеризации больших баз данных. Вводится их матричный аналог, что позволяет проводить обработку не просто больших объемов данных, но и больших баз многомерных данных, а соответственно позволяет работать с мультимедиа контентом таким как изображения или видео информация.*

Ключевые слова: *Иерархическая агломеративная кластеризация, мультимедиа, кластер, субкластер, дендрограмма, матричный алгоритм*

S. Bogucharskiy,
S. Mashtalir, PhD.

HIERARCHICAL AGGLOMERATIVE CLUSTERING IN MULTIMEDIA DATABASE

Abstract. *Agglomerative hierarchical clustering methods on large databases are discussed. We introduce their matrix analog that allows not only processing huge data amounts, but also multidimensional data large database, and therefore allows you to work with multimedia content such as images or video information.*

Keywords: *Hierarchical agglomerative clustering, multimedia, cluster, subcluster, dendrogram, matrix algorithm*

С. І. Богучарський,
С. В. Машталір, канд. техн. наук

ІЄРАРХІЧНА АГЛОМЕРАТИВНА КЛАСТЕРИЗАЦІЯ В БАЗАХ ДАНИХ МУЛЬТИМЕДІА

Анотація. *В роботі розглянуті методи ієрархічної агломеративної кластеризації великих баз даних. Введений їх матричний аналог, який дозволяє проводити обробку не тільки великих об'ємів даних, але й великих баз багатовимірних даних, а тому дозволяє працювати з мультимедіа контентом таким як зображення чи відео інформація.*

Ключові слова: *Ієрархічна агломеративна кластеризація, мультимедіа, кластер, субкластер, дендрограма, матричний алгоритм*

1. Введение

Развитие информационных технологий приводит к необходимости обработки больших объемов информации в том числе и визуальной. При этом задача кластеризации (автоматической классификации) массивов многомерных наблюдений, основной целью которой является нахождение в обрабатываемых выборках данных однородных в принятом смысле групп (классов, сегментов, кластеров), является важной частью активно развивающегося в настоящее время научного направления [1 – 4].

Несмотря на то, что на сегодня известны десятки, если не сотни подходов, различных методов и алгоритмов кластеризации, говорить о некотором универсальном подходе к решению всех возможных задач разбиения данных на множества не приходится, что определяется в значительной мере спецификой объектов.

Одной из сложных, требующих решения задач кластеризации является обработка видеоинформации, включая и ее поиск в сверхбольших базах данных (VLDB) [5 – 7]. Основными факторами, определяющими сложность этой задачи являются огромные объемы анализируемой информации, описание изображений, как правило, в матричной форме, наличие сегментов изображений сложной формы, их искаженность различного рода возмущениями и шумами. В таких задачах, классические традиционные методы кластеризации оказываются неэффективными либо вообще неработоспособными.

© Богучарский С.И., Машталир С.В., 2015

Следуя классификации, введенной в [6], методы кластеризации подразделяются на два больших класса: основанные на разбиении и иерархические.

На сегодня наибольшее распространение получении процедуры, основанные на разбиении, которые делят массив информации, содержащий N многомерных наблюдений, описываемых n-мерными векторами признаков $x(k) \in R^n$, $k = 1, 2, \ldots$, N, на p классов (сегментов), где p – основной параметр, задаваемый, как правило, априорно из эмпирических соображений.

Следует отметить, что подобные процедуры в качестве начальных данных получают некоторое произвольное разбиение, которое в процессе оптимизации, некоторой, также априори заданной, целевой функции, основанной на той или иной метрике (обычно евклидовой или манхэттенской), непрерывно корректируется с помощью той или иной итерационной процедуры. При этом каждое наблюдение, содержащееся в исходном массиве, неоднократно просматривается. Основной характеристикой каждого формируемого кластера является его центроид, вокруг которого группируются наблюдения конкретного класса. Наиболее характерными представителями этого подхода являются алгоритмы k-средних и т.п. Несмотря на популярность и достаточно строгую формализа-

цию алгоритмов разбиения, им присущи и существенные недостатки. К основному из них следует отнести то, что они формируют выпуклые сегменты, которые в реальных изображениях присутствуют далеко не всегда. Конечно, всякую невыпуклую фигуру можно покрыть множеством кругов достаточно малого радиуса. Однако при этом, возрастает вычислительная сложность алгоритма. Необходимость неоднократного просмотра каждого вектора-образа делает использование подобных алгоритмов в VLDB крайне проблематичным.

В отличие от этого подхода иерархические алгоритмы, которые, в свою очередь, делятся на агломеративные и дивизимные, автоматически определяют число классов путем слияния отдельных наблюдений в кластеры либо дробления исходного массива данных на подвыборки – кластеры. Понятно, что в этом случае число формируемых кластеров может лежать в интервале $2 \le p \le N-1$. Иерархический подход может формировать кластеры произвольной формы, крайне прост с алгоритмической точки зрения, однако в силу необходимости многократного просмотра всех наблюдений хранящихся в базе данных, крайне неудобен для обработки информации, содержащейся в VLDB. Кроме того, получаемые с помощью этого подхода результаты, весьма чувствительны к различного рода возмущениям и шумам, всегда присутствующим в реальных данных.

Нельзя не отметить также интенсивно развивающиеся последние годы методы кластеризации, основанные на плотности распределению данных [1]. Эти методы позволяют формировать кластеры произвольной формы в условиях, когда данные «зашумлены» возмущениями различной природы. При этом в рамках данного подхода под кластерами понимаются области в пространстве признаков с наиболее высокой плотностью распределения данных. Эти области чередуются с областями с низкой плотностью, где и концентрируются возмущения и помехи. Таким образом, алгоритмы, основанные на плотности, в процессе вычислений «выращивают» области с высокой плотностью распределения и формируют кластеры произвольной формы, отделяя при этом возмущения и шумы. Также нельзя не отметить, что данный подход может быть адаптирован для работы с VLDB. Вместе с тем алгоритмы, основанные на плотности, требуют предварительного задания ряда параметров, определяющих в конечном итоге качество получаемого результата.

Интуитивно понятно, что наиболее простым, понятным, хорошо интерпретируемым и наглядным является иерархический подход для кластеризации многомерных данных, и если бы удалось существенно сократить объем «просматриваемых» данных для формирования устойчивых кластеров, его можно было бы рекомендовать и для работы с VLDB. Понятно также, что эта идея не могла не привлечь внимание исследователей [7 – 9].

При этом стандартные подходы к кластеризации VLDB мультимедиа данных усложняются необходимостью представления многомерной (в данном случае двумерной) информации в качестве некоторого характеристического вектора, формируемого на основании некоторого множества характеристик/признаков визуальной информации. Что в общем случае является нетривиальной задачей. Таким образом, целью данной работы является разработка матричного аналога иерархического агломеративного метода кластеризации многомерных данных.

2. Матричный итеративный иерархический балансовый метод кластеризации

Исторически первым иерархическим агломеративным алгоритмом кластеризации, ориентированным на работу с VLDB, является BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies) [7], вычислительная сложность которого линейно зависит от объема обрабатываемой выборки, а возможность обработки данных в последовательном режиме делает его особенно привлекательным.

В рамках BIRCH вводится два основных понятия этого метода: «признак кластеризации» (Clustering Feature – CF) и «дерево признаков кластеризации» (CF – Tree, CF – дендрограмма), при этом здесь под дендрограммой понимается вложенная группировка объектов, образов, векторов признаков и т.п., изменяющаяся по определенным правилам на различных уровнях иерархии. Использование введенных понятий позволяет упростить вычисления, повысить скорость обработки и организовать динамическую кластеризацию вновь поступающих данных.

Чтобы ввести эти понятия допустим, что l-й кластер CL_l образован N_l n-мерными объектами-образами $x(\mathrm{k}) \in \mathrm{CL}_l$. Для этого кластера вводится его

центроид $C_l = \dfrac{1}{N_l} \underset{x(k)\in CL_l}{\sum_{k=1}^{N_l}} x(k)$,

радиус $R(l) = \left(\dfrac{1}{N_l} \underset{x(k)\in CL_l}{\sum_{k=1}^{N_l}} \| x(k) - \mathrm{C}(l) \|^2 \right)^{\frac{1}{2}}$

и диаметр

$$\mathrm{Dm}(l) = \left(\frac{1}{N_l(N_l-1)} \underset{x(k)\in CL_l}{\sum_{k=1}^{N_l}} \underset{x(q)\in CL_l}{\sum_{q=1}^{N_l}} \| x(k) - x(q) \|^2 \right)^{\frac{1}{2}}.$$

Несложно заметить, эти признаки порождены евклидовой метрикой, хотя несложно заметить, что и метрикой Минковского.

Собственно же кластеризация на верхних уровнях иерархии реализуется на основе CF-векторов определяемых тройкой $CF(l) = (\mathrm{N}_l, LS^T(l), \mathrm{SS}(l))^T$,

где $LS = \underset{x(k)\in CL_l}{\sum_{k=1}^{N_l}} x(k)$, $SS = \underset{x(k)\in CL_l}{\sum_{k=1}^{V_l}} \| x(k) \|^2$.

Таким образом размерность вектора $CF(l)$ есть $(\mathrm{n}+2)\mathrm{x}1$. Важно, что CF-векторы обладают свойством аддитивности, т.е.

$$CF(l,\ l+1) = CF(l) + CF(l+1) =$$
$$= (\mathrm{N}_l + N_{l+1},\ LS^T(l) + \mathrm{LS}^T(l+1),\ \mathrm{SS}(l) + \mathrm{SS}(l+1))^T$$

и могут уточняться по мере поступления новых данных.

В дальнейшем BIRCH оперирует только с *CF*, которые накапливаются и анализируются с помощью *CF*-дендрограммы. Сама же *CF*-дендрограмма характеризуется двумя признаками: фактором ветвления *B*, который определяет максимальное число элементов в каждом кластере (субкластере) на нижнем уровне иерархии, и максимальным диаметром субкластера *T* в каждом узле дерева (уровня иерархии). Понятно, что чем больше значение *B*, тем меньше количество кластеров будет сформировано, а чем больше *T*–тем меньше уровней иерархии будет иметь дендрограмма.

Кластеризация на основе BIRCH происходит в два этапа.

На первом этапе в результате однократного просмотра базы данных формируется исходная дендрограмма, при этом на нижнем уровне обрабатываются исходные данные $x(k)$, а на верхних-векторах признаков $CF(l)$.

На втором этапе производится кластеризация сформированных на первом этапе субкластеров, при этом субкластеры, содержащие малое число объектов, удаляются как шумы и выбросы. Сливаются в один субкластеры, чьи центроиды расположены достаточно близко, т.е. $D(\mathrm{C}l_l, \mathrm{C}l_r) = \| C(l) - \mathrm{C}(r) \| < \Delta$.

Таким образом, достигается работоспособность процесса кластеризация, а последовательная обработка сводится к тому, что вновь поступивший образ $x(k)$ просто «вставляется» в субкластер с наиболее близко расположенным центроидом.

Реализация BIRCH предполагает, что каждый обрабатываемый объект описывается n-мерным вектором $x(k)$, а сам процесс обработки информации связан с векторными операциями. В ситуации, когда необходимо обрабатывать двумерные изображения, они должны быть подвергнуты векторизации, что ведет к резкому возрастанию размерности CF-векторов, после чего собственно решается задача кластеризации, результат решения которого далее должен быть девекторизован. Процесс кластеризации массивов изображений можно упростить, используя вместо векторных операций соответствующие матричные операции, при этом исходная информация задается не в форме n-мерных векторов, а в виде матриц $x(k) = \{x_{i1,\ i2}(k)\}$, $i_1 = 1, 2, ..., m$; $i_2 = 1, 2, ..., n$; $k = 1, 2, ..., N$; $x(k) \in R^{m \times n}$.

Далее можно ввести в рассмотрение центроид l-го кластера $C_l = \frac{1}{N_l} \sum\limits_{\substack{k=1 \\ x(k) \in Cl_l}}^{N_l} x(k)$, матричный радиус

$$R_l = \left(\frac{1}{N_l} \sum_{\substack{k=1 \\ x(k) \in Cl_l}}^{N_l} Sp(x(k) - \mathrm{C}(l))(x(k) - \mathrm{C}(l))^T \right)^{\frac{1}{2}},$$

матричный диаметр

$$Dm(l) =$$
$$= \left(\frac{1}{N_l N_{l-1}} \sum_{\substack{k=1 \\ x(k) \in Cl_l}}^{N_l} \sum_{\substack{q=1 \\ x(\mathrm{q}) \in Cl_l}}^{N_l} Sp(x(k) - (x(q))(x(k) - (x(q))^T) \right)^{\frac{1}{2}}$$

и CF-матрицу $CF(l) = \begin{pmatrix} N_l & \vdots \\ SS(l) & \vdots \\ \cdots\cdots & LS(l) \\ \bar{O} & \vdots \end{pmatrix}$ размерности $(\mathrm{m} \times (\mathrm{n}+1))$.

Здесь $LS(l) = \sum\limits_{\substack{k=1 \\ x(k) \in CL_l}}^{N_l} x(k)$,

$$SS(l) = \sum_{\substack{k=1 \\ x(k) \in Cl_l}}^{N_l} \mathrm{Sp}(x(k)x^T(k)),$$

$\bar{O}$ – вектор, образованный нулями.

При этом в качестве меры расстояния используется сферическая норма

$$D_S(x(k), x(r)) = (\mathrm{Sp}(x(k) - x(r))((x(k) - x(r))^T)^{\frac{1}{2}}.$$

Далее может быть реализована BIRCH-процедура, где вместо CF-векторов используются введенные нами CF-матрицы.

Таким образом, можно говорить о том, что матричная модификация метода BIRCH проста в численной реализации, достаточно быстра, робастна различного рода выбросам, допускает последовательную обработку данных.

В тоже время следует отметить и что главный недостаток такой процедуры состоит в том, что в результате ее применения формируются только выпуклые кластеры, что, естественно ограничивает ее применимость и заставляет искать альтернативные подходы.

3. Заключение

Рассмотрена задача кластеризации больших массивов мультимедиа данных на основе иерархического агломеративного подхода. Введена матричная модификация метода BIRCH, позволяющая обрабатывать многомерные данные без использования операций векторизации-девекторизации, что позволяет сэкономить, во-первых время, необходимое на построение векторов характеристик изображений или кадров видеопоследовательности, во-вторых, избежать проблем с выбором самих таких характеристик и определения их важности для решения задачи кластеризации, что

учитывая VLDB является достаточно важным аспектом в виду большого разнообразия входных данных.

Предлагаемая процедура кластеризации проста в численной реализации, не требует многократного просмотра обрабатываемого массива, обеспечивают последовательную обработку поступающей информации, формируя кластеры выпуклой формы в условиях воздействия интенсивных возмущений. В свою очередь предполагается развитие введения матричных модификаций методов кластеризации, для того, чтобы можно было получать кластеры произвольной формы.

Список использованной литературы

1. Han J., and Kamber M., (2006), Data Mining: Concepts and Techniques, 2-nd ed., *San Francisco: Morgan Kaufmann,* 800 p.

2. Бодянский Е. В. Самообучающаяся каскадная спайк-нейронная сеть для нечеткой кластеризации на основе метода группового учета аргументов / Е. В. Бодянский, Е. А. Винокурова, А. И. Долотов // Проблемы управления и информатики. – 2013. – К. : – № 2. – С. 25 – 34.

3. Колчигин Б. В. Адаптивная нечеткая кластеризация с переменным фаззификатором / Б. В. Колчигин, Е. В. Бодянский // Кибернетика и системный анализ. – 2013. – К. : – Т. 49. – № 3. – С. 47 – 55.

4. Бодянский Е. В. Адаптивная нечеткая кластеризация данных на основе метода Густафсона–Кесселя / Е. В. Бодянский, Б. В. Колчигин, В. В. Волкова, И. П. Плисс // Управляющие системы и машины. – 2013. – К. : – № 2**.** – С. 40 – 46.

5. Ng R.T., and Han J., (1994), Efficient and Effective Clustering Methods for Spatial Data Mining, *Proc. 20th Int. Conf. on Very Large Data Bases.* Santiago, Chile, pp. 144 – 145.

6. Kaufman L., and Rousseeuw P.J. Finding Groups in Data: An Introduction to Cluster Analysis, (1990). *John Wiley&Sons,* N.Y**.,** 342 p.

7. Zhang T., Ramakrishnan R., and Livny M., (1996), BIRCH: An Efficient Data Clustering Method for very Large Databases, *Proc. Of the ACM SIGMOD Conf. On management of Data. ACM Press*, Montreal pp. 103 – 114

8. Guha S., Rastogi R., and Shim K., (2001), CURE: an Efficient Clustering Algorithm for Large Databases, *Elsevier, Information Systems,* New Jersey, 26, No. 1, pp. 35 – 58.

9. Jain A.K., and Dubes R.C., (1988), Algorithms for Clustering Data, *Englewood Cliffs, N.J.: Prentice Hall,* New Jersey, 318 p.

Получено 15.05.2015

References

1. Han J., Kamber M., (2006) Data Mining: Concepts and Techniques. 2-nd ed., *San Francisco: Morgan Kaufmann*, 800 p. (In English)

2. Bodyanskiy Ye.V., Vynokurova E.A., and Dolotov A.I., (2013), Self-Learning Cascade Spiking Neural Network for Fuzzy Clustering Based on Group Method of Data Handling, *Journal of Automation and Information Sciences,* 45, No. 3, pp.23 – 33 (In English).

3. Kolchygin B.V., and Bodyanskiy Ye.V., (2013), Adaptive Fuzzy Clustering with a Variable Fuzzifier, *Cybernetics and Systems Analysis*, 49, Issue 3, pp. 366 – 374 (In English).

4. Bodyanskiy Ye.V., Kolchygin B.V., Volkova V.V., and Pliss I.P., Adaptivnaya nechetkaya klasterizatsiya dannykh na osnove metoda Gustafsona–Kesselya [Adaptive Fuzzy Clustering via Gustafson-Kessel Methods], (2013), *Upravliaushie Sistemi i Mashini,* No. 2, pp. 40 – 46 (In Russian).

5. Ng R.T., and Han J., (1994), Efficient and Effective Clustering Methods for Spatial Data Mining, *Proc. 20th Int. Conf. on Very Large Data Bases.* Santiago, Chile, pp. 144 – 145 (In English).

6. Kaufman L., and Rousseeuw P.J., (1990), Finding Groups in Data: An Introduction to Cluster Analysis. *N.Y.: John Wiley&Sons,* 342 p. (In English).

7. Zhang T., Ramakrishnan R., and Livny M., (1996), BIRCH: An Efficient Data Clustering Method for Very Large Databases, *Proc. Of the ACM SIGMOD Conf. On management of Data, Montreal: ACM Press*, pp. 103 – 114 (In English).

8. Guha S., Rastogi R., and Shim K., (2001), CURE: am Efficient Clustering Algorithm for Large Databases, *Information Systems,* 26, No. 1, pp. 35 – 58 (In English).

9. Jain A.K., and Dubes R.C., (1988), Algorithms for Clustering Data, *Englewood Cliffs, N.J.: Prentice Hall*, 318 p. (In English).

Богучарский
Сергей Игоревич, аспирант каф. информатики Харьковского нац. ун-та радиоэлектроники.
Украина, Харьков, пр. Ленина, 14, ауд. 288.
E-mail: sbogucharskiy@rambler.ru

Машталир
Сергей Владимирович, к.т.н., доц. каф. информатики Харьковского нац. ун-та радиоэлектроники.
Украина, Харьков, пр. Ленина, 14, ауд. 288.
E-mail: mashtalir_s@kture.kharkov.ua