УДК 004.01:004.89

О. В. Пескова

Методы автоматической классификации электронных текстовых документов без обучения

Рассматривается задача автоматической кластеризации коллекций текстовых электронных документов. Описываются кластеризационные методы разбиения данных, затрагивается проблема оценки эффективности методов кластеризации, основной акцент при этом делается на сравнении различных аспектов их программной реализации, важных при выборе метода для конкретных условий задачи. Рассматриваются методы проверки обоснованности кластерного решения

ВВЕДЕНИЕ

В последние годы стремительно развиваются электронные документные фонды, дающие большие преимущества в обслуживании читателей, в первую очередь - поисковые возможности, реализуемые различными механизмами, основным из которых является классификация документов. Однако быстрое развитие информационных массивов делает чрезмерно трудоёмкими процессы формирования классификационной схемы и классификации по ней документов вручную. Стала очевидной необходимость разработки и внедрения автоматических средств, выполняющих классификационные действия над электронными текстовыми документами.

Автоматические методы классификации текстовых документов можно разделить на две группы: с обучением (для исследования доступны обучающие выборки) и без обучения (обучающие выборки недоступны). Автоматические методы классификации текстов с обучением называют также методами текстовой категоризации, информацию о них можно найти, например, в [1]. В настоящей статье будут обсуждаться методы классификации без обучения -- методы кластеризации текстов.

ЗАДАЧА АВТОМАТИЧЕСКОЙ КЛАСТЕРИЗАЦИИ ДАННЫХ

Рассмотрим задачу классификации текстовых документов, в которой необходимо классифицировать исходную коллекцию документов без наличия априорной информации о ней, т. е. исходя только из анализа и выявления внутренней тематической структуры коллекции. В таком случае следует применять методы автоматической классификации текстовых документов без обучения. В основе этих методов лежит *кластерный анализ*, главным назначением которого является разбиение множества исследуемых объектов и признаков на однородные (в соответствующем понимании) группы, или кластеры.

Кластерный анализ текстовых документов основывается на **кластерной гипотезе** [2]: тесно связанные по смыслу документы стремятся быть релевантными одним и тем же запросам, т. е. документы, релевантные запросу, отделимы от тех, которые нерелевантны этому запросу.

В общем случае процесс кластеризации текстовых документов состоит из следующих этапов:

1) Предобработка и индексация текстов на естественном языке (в результате формируются векторы признаков документов как представление данных документов).

2) Выбор алгоритма кластеризации, в частности, выбор меры сходства документов и кластерного критерия таким образом, чтобы ожидать максимально эффективного разбиения конкретных исходных данных.

3) Проверка обоснованности кластерного решения выбранными для данного случая критериями и техниками оценки обоснованности.

В данной статье мы будем рассматривать второй и третий этапы и начнём с одного из ключевых понятий, лежащих в основе каждого алгоритма кластеризации, - сходства/различия между документами.

Рассмотрим наиболее часто используемые **меры сходства** текстовых документов $Sim(x_i, x_j)$.

Косинусная мера сходства (*cosine similarity*) вычисляет значение косинуса между двумя векторами документов:

$$Sim(x_i, x_j) = \cos(\angle(x_i, x_j)) = \frac{\sum_{k=1}^{|T|} x_{ki} x_{kj}}{\sqrt{\sum_{k=1}^{|T|} x_{ki}^2} \cdot \sqrt{\sum_{k=1}^{|T|} x_{kj}^2}}, \quad (0)$$

где T - множество признаков коллекции документов.

Часто мера сходства документов вычисляется на основе измерения расстояния между векторами документов в многомерном пространстве признаков документов, тогда:

$$Sim(x_i, x_j) = 1 - d(x_i, x_j).$$

Степенное расстояние позволяет прогрессивно увеличить или уменьшить вес, относящийся к размерности, для которой соответствующие документы сильно отличаются:

$$d(x_i, x_j) = \left(\sum_{k=1}^{|T|} |x_{ik} - x_{jk}|^p \right)^{1/r}, \qquad (1)$$

где r и p — параметры, определяемые пользователем.

Параметр p ответственен за постепенное взвешивание разностей по отдельным координатам, параметр r — за прогрессивное взвешивание больших расстояний между объектами.

Самыми популярными типами расстояний для текстовой кластеризации являются частные случаи степенного расстояния: евклидово расстояние ($p = 2,\ r = 1/2$), квадрат эвклидова расстояния ($p = 2, r = 1$), расстояние городских кварталов, или манхэттенское расстояние ($p = 1, r = 1$).

МЕТОДЫ КЛАСТЕРИЗАЦИИ ЭЛЕКТРОННЫХ ТЕКСТОВЫХ ДОКУМЕНТОВ

По типу разбиения данных существующие методы кластеризации текстов могут быть разделены на следующие основные группы:

1) иерархическая или плоская (неиерархическая) кластеризация;

2) мягкая (нечёткая) или жёсткая (чёткая) кластеризация.

При *иерархической кластеризации* получается древовидная структура кластеров, каждый узел этого дерева представляет кластер, который содержит все объекты его кластеров-потомков. *Плоская кластеризация* состоит из заданного числа кластеров, отношения между которыми часто не определены. При *жёсткой кластеризации* каждый объект связан с одним и только с одним кластером. *Мягкая кластеризация* позволяет объекту принадлежать многим кластерам с различной степенью принадлежности.

1. Иерархические методы

Алгоритмы иерархической кластеризации бывают двух типов: агломеративный (agglomerative clustering) и разделяющий (divisive clustering).

Агломеративный алгоритм производит дерево документов снизу вверх, начиная с отдельных кластеров для каждого документа и группируя наиболее сходные кластеры в один новый кластер. Алгоритм завершается, когда сформирован один огромный кластер.

Для объединения кластеров документов необходим выбор правила вычисления расстояния между кластерами. Рассмотрим базовые правила.

Одиночная связь, или метод ближайшего соседа (Single-link)

В этом методе расстояние между двумя кластерами определяется расстоянием между двумя наиболее близкими объектами (ближайшими соседями) в различных кластерах. Это правило строит кластеры, "сцепленные вместе" только отдельными элементами, случайно оказавшимися ближе остальных друг к другу, поэтому результирующие кластеры имеют тенденцию быть представленными длинными "цепочками". Считается, что данный метод даёт низкое глобальное качество, не учитывает глобальный контекст. Но полученные таким образом кластеры отличаются хорошей локальной связанностью.

Полная связь, или метод наиболее удаленных соседей (Complete-link)

В качестве альтернативы одиночной связи можно рассматривать сходство между кластерами как сходство между их наиболее различными членами (т. е. "наиболее удаленными соседями"). Данный метод сфокусирован на глобальном качестве кластеров. Обычно он работает очень хорошо, когда объекты на самом деле происходят из реально различных "рощ". Если же кластеры имеют в некотором роде удлиненную форму или их естественный тип является "цепочечным", то этот метод непригоден. Тем не менее для большинства задач обработки естественного языка (ОЕЯ) сферообразные кластеры, полученные методом кластеризации с полной связью, признаны более предпочтительными, чем вытянутые кластеры, полученные методом кластеризации с одиночной связью [3].

Попарное среднее (Group-average)

Данный способ является компромиссом между Single-link и Complete-link и вычисляет расстояние между двумя различными кластерами как среднее расстояние между всеми парами объектов в них. Метод эффективен, когда объекты в действительности формируют различные "рощи", однако он работает хорошо и в случаях протяженных ("цепочного" типа) кластеров.

Поскольку большинство систем иерархической текстовой кластеризации основаны на применении агломеративного алгоритма в комбинации с одним из описанных выше правил объединения кластеров, то в литературных источниках можно встретить название правила объединения кластеров в качестве названия применяемого иерархического метода кластеризации.

Разделяющий алгоритм производит дерево документов сверху вниз, начиная с одного кластера, содержащего все документы и разделяя наименее связные кластеры на группы с целью максимизации сходства внутри группы. Кластеры со сходными объектами считаются более связными, чем кластеры с различными объектами. Алгоритм завершается, когда все полученные кластеры содержат по одному документу.

Разбиение одного кластера на два также является кластеризационной задачей — задачей поиска двух подкластеров одного кластера. Для операции разделения может использоваться любой алгоритм кластеризации, включая агломеративные алгоритмы и неиерархические методы. Возможно, из-за необходимости рекурсивного применения второго алгоритма кластеризации разделяющая кластеризация не так часто используется по сравнению с методом "снизу вверх".

Результатом работы этих иерархических алгоритмов является дендрограмма (бинарное дерево), связывающая все тексты. При заданном вручную

числе кластеров (или предельной величине близости) делается соответствующее сечение бинарного дерева, дающее разбиение текстов на кластеры.

Метод кластеризации с использованием суффиксных деревьев

Кроме рассмотренной выше общей схемы иерархической кластеризации существуют и другие методы кластеризации, которые можно отнести к группе иерархических. К таким методам принадлежит, в частности, метод суффиксных деревьев (Suffix Trees) [4].

Изначально суффиксные деревья были разработаны и применялись для быстрого поиска подстрок. Позднее в диссертации [5] было предложено использовать идею суффиксных деревьев для кластеризации результатов запросов поисковой системы. Построение дерева документов осуществляется следующим образом: для набора документов, получаемых в ответ на запрос поискового сервера, строится дерево, единицей, находящейся на рёбрах дерева, является слово или словосочетание. Каждой вершине дерева соответствует фраза. Её можно получить, объединив все слова/словосочетания, находящиеся на рёбрах на пути от корня дерева к данной вершине дерева. В вершине дерева имеются ссылки на документы, в которых встречается фраза, соответствующая вершине. Множества документов, на которые указывают эти ссылки, образуют базовые кластеры. После этого производится комбинирование базовых кластеров и получение набора кластеров. Комбинируются кластеры по простой методике:

$$\text{если } \frac{|C_i \cap C_j|}{|C_i|} > 0.5 \text{ и } \frac{|C_i \cap C_j|}{|C_j|} > 0.5, \text{ где } |C_i|, |C_j| \text{ и } |C_i \cap C_j| - \text{ размеры} \qquad (3)$$

соответствующих кластеров, то базовые кластеры объединяются в один общий.

Основным достоинством метода является линейная скорость построения дерева, достигаемая посредством использования специальных указателей.

2. Метод квадратичной ошибки

Одним из наиболее часто используемых критериев в плоской кластеризации является критерий квадратичной ошибки, который имеет тенденцию хорошо работать с изолированными и компактными кластерами [6].

Квадратичная ошибка кластеризации (разбиения) C множества объектов X, содержащая k кластеров, вычисляется по следующей формуле:

$$e^2(X, C) = \sum_{j=1}^{k} \sum_{i=1}^{n_j} \left\| x_i^{(j)} - c_j \right\|^2, \qquad (4)$$

где $x_i^{(j)}$ — это i-й объект, принадлежащий j-у кластеру, c_j — центроид j-го кластера.

Самым распространённым алгоритмом, применяющим идею критерия квадратичной ошибки, является алгоритм по методу k-средних.

Метод k-средних (k-means) — это жёсткий кластеризационный алгоритм, который определяет кластеры по центру тяжести их членов и представляет собой частный случай общего метода EM [3]. Метод k-средних строит k различных кластеров, расположенных на возможно больших расстояниях друг от друга, действуя по следующей обобщённой схеме:

1) Выбор k начальных центров кластеров.

2) Каждый документ присваивается тому кластеру, чей центр является наиболее близким документу.

3) Перевычисляются центры каждого кластера как центроиды или средние своих членов по формуле:

$$\vec{\mu} = \frac{1}{M} \sum_{\vec{x} \in c} \vec{x} \qquad (5)$$

4) Если достигнуто условие остановки, то алгоритм завершается, иначе — п. 2.

Функцией расстояния между кластером и документом является евклидово расстояние.

Исходные центры кластеров обычно выбираются случайным образом. Важен ли способ выбора начальных центров или нет, зависит от набора данных: многие наборы доброкачественны, и большинство способов инициализации приведёт к результатам кластеризации со сравнимым качеством, а для непредсказуемых наборов данных необходимо сначала вычислить хорошие центры кластеров, как, например, в алгоритме Бакшота (Buckshot), в котором для инициализации центров кластеров применяется попарно средний агломеративный алгоритм кластеризации к случайно выбранному подмножеству данных размером равным квадратному корню размера полного множества.

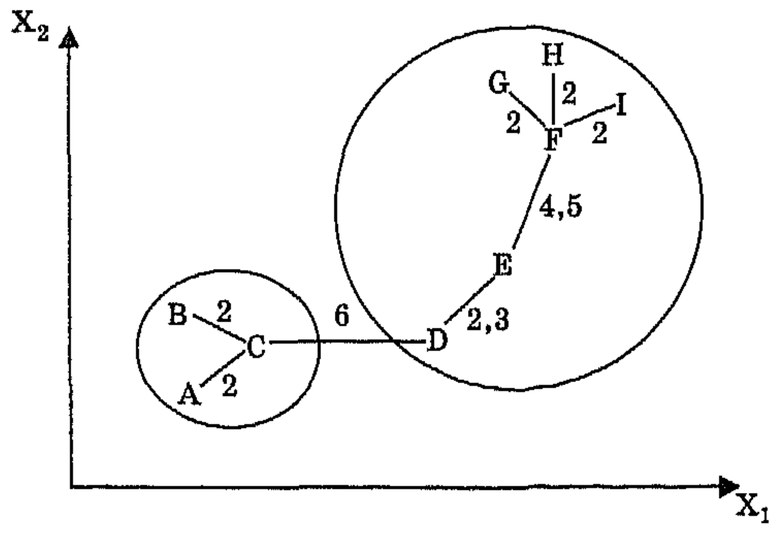
Метод k-средних является одним из самых простых алгоритмов кластеризаций и несмотря на его ограничения работает весьма удовлетворительно для решения многих проблем ОЕЯ [3].

Однако метод k-средних весьма чувствителен к выбросам, которые могут искажать среднее. Возможным решением этой проблемы является модификация алгоритма — алгоритм k-медиан (k-medoids), который менее чувствителен к шумам и выбросам данных, поскольку медиана меньше подвержена влияниям выбросов.

3. Методы теории графов

Самым известным алгоритмом плоской кластеризации по методу теории графов является **алгоритм, основанный на построении минимального остовного дерева** (minimal spanning tree MST) с последующим удалением рёбер MST с наибольшими длинами для генерации кластеров [6]. Для пояснения идеи алгоритма рассмотрим пример (рисунок).

На рисунке изображено минимальное остовное дерево, полученное для 9-ти двуразмерных объектов. Путём удаления связи, помеченной CD, с длиной равной 6-ти единицам (ребро с максимальным евклидовым расстоянием) получаем два кластера: {A, B, C} и {D, E, F, G, H, I}. Второй кластер в дальнейшем может быть разделён ещё на два кластера путём удаления ребра EF, которое имеет длину равную 4,5 единицам.



Использование минимального остовного дерева для получения кластеров

Важно заметить, что между кластеризацией методом теории графов и иерархической кластеризацией методами Single-link и Complete-link существует связь, которая заключается в возможной интерпретации Single-link и Complete-link как максимально связанного и максимально полного графов соответственно, полученных из MST путём удаления рёбер [3]. Таким образом, MST содержит больше информации, чем, например, иерархия, полученная Single-link, однако при методе MST существенно более сложный процесс обновления кластеризации при добавлении нового объекта, чем при использовании метода Single-link [2].

4. Методы разрешения смесей

Разрешение смесей (Mixture-Resolving) как подход к кластеризации основан на допущении, что кластеризуемые объекты разбросаны по одному из нескольких распределений; тогда целью является идентификация количества этих распределений и их параметров. Традиционный подход к решению данной проблемы применяет итеративное получение оценки максимальной вероятности векторов параметров распределений. На этом подходе основан **метод максимизации ожидания** (expectation maximization – EM).

Метод EM оперирует вероятностной моделью отнесения документа к определенному кластеру (плоскому). Предполагается, что существует k (по числу кластеров) скрытых независимых "генераторов", подчиняющихся многомерному закону нормального распределения. Векторы документов рассматриваются в качестве реализации многомерной случайной величины. Если параметры распределений известны, можно вычислить условную вероятность принадлежности вектора документа к одному из "генераторов". Таким образом, EM-алгоритм, начиная с исходной оценки вектора параметров, итеративно подбирает параметры распределений (средние и стандартные отклонения) так, чтобы правдоподобие наблюдаемых данных (распределения) было максимально.

EM-алгоритм является весьма полезным и популярным в настоящее время, тем не менее он имеет ряд существенных недостатков. Основным недостатком EM-алгоритма считается его высокая чувствительность к инициализации параметров. Но даже если параметры неплохо инициализированы, алгоритм обычно попадает в одну из многочисленных локальных точек минимума в рассматриваемом пространстве. Одно из средств устранения этих проблем — использовать для инициализации параметров результаты работы другого алгоритма кластеризации. Например, алгоритм k-средних является очень эффективным способом поиска начальных оценок координат центров кластеров EM-алгоритма для гауссовых смесей. Скорость схождения данного алгоритма может быть очень медленной. Наконец, стоит отметить, что данный алгоритм в действительности используется только тогда, когда не существует более прямого способа решения условно-оптимизационной задачи [3].

5. Методы, основанные на концепции плотности

Известным алгоритмом, использующим для формирования кластеров концепцию плотности (density), является **DBSCAN (Density Based Spatial Clustering of Applications with Noise)** [7], разработанный для обнаружения кластеров и шума в пространственных базах данных. Применение этого алгоритма к задаче кластеризации текстов встречается в работах [8, 9].

Суть алгоритма DBSCAN заключается в обнаружении кластеров на основе предположения о том, что внутри каждого кластера наблюдается типичная плотность объектов, которая значительно выше плотности объектов за пределами кластера. Более того, плотность в областях шума ниже, чем плотность в любом из кластеров. Эта интуитивная нотация "кластеров" и "шума" формализуется для данных в некотором многомерном пространстве, следуя ключевой идее: для каждого объекта кластера соседство заданного радиуса должно содержать, по крайней мере, минимальное количество объектов, т. е. плотность в соседстве должна превышать заданное пороговое значение. Таким образом (анализируя плотность соседства каждого объекта) исследуется всё пространство объектов, и те объекты, которые не вошли ни в один из кластеров, объявляются шумом.

Основными преимуществами данного алгоритма являются способность обнаруживать кластеры произвольной формы (в отличие, например, от сферических кластеров метода k-средних) и попытка одновременно обнаруживать шумовые объекты. Главный недостаток DBSCAN - необходимость "вручную" подбирать приемлемые значения параметров плотности объектов кластеризации. Кроме того, в работе [8] сделан вывод о неэффективности применения данного метода к большим текстовым коллекциям без дополнительных модификаций алгоритма, поскольку алгоритм классифицировал большинство документов текстовых коллекций как шум или поместил их в один огромный кластер.

6. Методы, основанные на нейронных технологиях

Искусственные нейронные сети интенсивно применяются для решения кластеризационных задач. Наиболее известными являются алгоритмы ART и SOM.

ART: методы теории адаптивного резонанса

В семье алгоритмов теории адаптивного резонанса (Adaptive Resonance Theory) — ART1, ART2

и ARTMAP — первым считается алгоритм ART1. Это очень простой алгоритм с обучением, основанный на свойствах человеческого мозга изучать новые понятия, сравнивая их с уже существующими знаниями [10].

Обучающий алгоритм ART1 корректирует имеющийся прототип категории (кластера), только если входной вектор в достаточной степени ему подобен. В этом случае они резонируют. Степень подобия контролируется пороговым значением сходства, которое связано также с числом категорий. Когда входной вектор не подобен ни одному существующему прототипу сети, создается новая категория и с ней связывается нераспределенный (неиспользуемый) элемент со входным вектором в качестве начального значения прототипа. ART1 отбрасываст входные примеры, когда сеть исчерпала свою емкость. Число обнаруженных сетью категорий чувствительно к параметру сходства.

Алгоритм ART1 концептуально прост и лёгок в реализации. Недостаток его заключается в том, что конечный набор кластеров (и векторов-прототипов) может изменяться в зависимости от порядка, в котором проводилось обучение.

SOM: самоорганизующиеся карты

Самоорганизующиеся карты (Self Organizing Maps — SOM), или карты Кохонена [11, 12], представляют собой один из вариантов кластеризации многомерных данных. Самоорганизующаяся карта является разновидностью нейронной сети. В алгоритме SOM все нейроны (узлы, центры классов) упорядочены в некоторую структуру, обычно двумерную сетку (прямоугольной или шестиугольной конфигурации). Обучение сети состоит из последовательности коррекций векторов, представляющих собой нейроны. На каждом шаге обучения из исходного набора данных случайно выбирается один из векторов, а затем производится поиск наиболее похожего на него вектора коэффициентов нейронов. При этом выбирается нейрон-победитель, который наиболее похож на вектор входов по оценке расстояния между векторами, обычно вычисляемого в евклидовом пространстве. Затем производится корректировка весов как нейрона-победителя, так и векторов, описывающих его соседей, которые в сетке перемещаются в направлении входного вектора. Для модификации весовых коэффициентов используется формула:

$$w_i(t+1) = w_i(t) + h_{ci}(t) * [x(t) - w(t)], \qquad (6)$$

где t — номер эпохи, $x(t)$ вектор, случайно выбранный из обучающей выборки на итерации t, $h(t)$ — функция соседства нейронов, зависящая от собственно функции расстояния между нейроном-победителем и соседними нейронами и функции скорости обучения от времени.

На карте кластером будет являться группа векторов, расстояние между которыми внутри этой группы меньше, чем расстояние до соседних групп. Структура кластеров при использовании алгоритма SOM может быть отображена путем визуализации расстояния между опорными векторами (весовыми коэффициентами нейронов).

Поскольку в ходе обучения модифицируется не только нейрон-победитель, но и его соседи (хотя и в меньшей степени), SOM можно считать одним из методов проецирования многомерного пространства в пространство с более низкой размерностью. При использовании этого алгоритма векторы, близкие на полученной карте, оказываются близки и в исходном пространстве.

7. Эволюционной подход к кластеризации

Эволюционный подход мотивирован естественной эволюцией, использует эволюционные операторы и популяцию решений для получения глобально оптимального разделения данных [6, 13, 14]. Решения-кандидаты кластеризационной задачи рассматриваются как хромосомы. Наиболее широко применяемыми эволюционными операторами являются селекция, рекомбинация и мутация. Каждый из операторов трансформирует одну или более входных хромосом в одну или более выходных хромосом. На основе анализа значения функции здоровья выясняется вероятность выживания хромосомы в следующем поколении.

Опишем эволюционный алгоритм для кластеризации:

1) Случайным образом выбрать популяцию решений. Каждое решение соответствует обоснованному k-разделению данных. Связать значения функции здоровья с каждым решением. Обычно значения здоровья обратно пропорциональны значению квадратичной ошибки.

2) Использовать эволюционные операторы для генерации следующего поколения решений. Оценить значения здоровья этих решений.

3) Повторять шаг 2 до тех пор, пока не будет удовлетворено условие остановки.

Самыми часто используемыми в кластеризации эволюционными техниками являются генетические алгоритмы. В отличие от чёткого и нечёткого методов k-средних, методов на основе нейронных сетей, генетические алгоритмы выполняют глобальный поиск решения задачи, а не локальный. Однако основной проблемой генетических алгоритмов является их высокая чувствительность к выбору различных параметров алгоритма, например, размера популяции, вероятностей кроссовера и мутации и т. п.

8. Методы понижения размерности пространства как кластеризаторы документов

Методы понижения размерности пространства признаков документов призваны решать проблемы формирования эффективных индексов документов. Однако результат работы данных методов уже сам по себе может рассматриваться и как кластеризация данных. В основе этих методов лежат идеи факторного анализа о том, что признаки документов каким-то семантическим образом связаны между собой (а значит, коррелированны), следовательно, документы, содержащие семантически близкие термины, сгущаются в определённых местах пространства терминов. После определения числа и параметров латентных факторов (латентно-семантический анализ) или факторов как комбинаций явных признаков (главных компонент) происходит отображение множества документов на пространство факторов.

Выделенные главные факторы являются координатными осями нового — редуцированного — пространства факторов. В полученном факторном пространстве документы и термины концентрируются областями, имеющими общий семантический, латентный, смысл. Применительно к кластеризации получаемые области и есть кластеры.

Рассмотрим основные методы, реализующие данные идеи.

Латентно-семантический анализ

Латентно-семантический анализ, или латентно-семантическое индексирование (LSA/LSI) [4, 15, 16] — метод обнаружения латентных связей, основанный на идее, что совокупность всех контекстов, в которых встречается и не встречается данное слово, задает множество обоюдных ограничений, которые в значительной степени позволяют определить похожесть лексических значений слов между собой. В качестве исходной информации LSA использует матрицу "термы-на-документы". Элементы этой матрицы содержат частоты использования каждого терма в каждом документе.

Наиболее распространенный вариант LSA основан на использовании разложения матрицы по сингулярным значениям (singular value decomposition — SVD). Используя SVD, огромная исходная матрица разлагается во множество из p ортогональных матриц, линейная комбинация которых является неплохим приближением исходной матрицы, т. е. сокращённая матрица "термы-на-документы", содержащая только p первых линейно независимых компонент, отражает основную структуру ассоциативных зависимостей, присутствующих в исходной матрице, и в то же время не содержит шума. Таким образом, каждый терм и документ представляются при помощи векторов в общем пространстве размерности p (пространстве факторов).

Главным достоинством метода LSA/LSI является то, что он пытается преодолевать проблемы синонимии и омонимии, присущие текстовой коллекции, и при этом опирается только на статистическую информацию о множестве документов/признаков, тогда как из-за этих двух проблем простые средства обработки текстов могут принимать неадекватные решения. Однако в этом кроется и основной недостаток — высокие вычислительные затраты, что становится критичным на огромных массивах данных, какими и являются текстовые коллекции.

Метод главных компонент

Метод главных компонент (МГК) основан на предположениях, что, с одной стороны, наиболее интересными для процесса классификации признаками документов являются те, которые обнаруживают наибольшую изменчивость (наибольший разброс) при переходе от одного документа к другому, а с другой стороны, к вполне удовлетворительной классификации объектов может привести система, использующая малое количество признаков, каждый из которых является некоторой комбинацией от большого числа непосредственно замеряемых на объекте признаков [16, 17].

МГК вычисляет собственные векторы и собственные значения ковариационной матрицы признаков документов с целью построения на данных векторах нового пространства. Размерность нового пространства заметно меньше размерности исходного, поскольку для его построения выбираются собственные векторы, соответствующие нескольким наибольшим собственным значениям. А главными компонентами исследуемой системы признаков считаются линейные комбинации признаков, обладающие наибольшей дисперсией. Геометрически первая главная компонента проходит в направлении наибольшего разброса исходных данных, т. е. в направлении вытянутости анализируемого "облака" документов. Каждая последующая главная компонента перпендикулярна предыдущим и так же проходит в направлении "наибольшей вытянутости" (среди возможных перпендикулярных направлений).

Нужно отметить, что корректное применение метода главных компонент возможно лишь при предположении о нормальном распределении векторов исходного набора. Что касается кластеризации текстовых документов, то метод главных компонент может оказаться недостаточно эффективным ввиду, во-первых, его линейности, а во-вторых, невысокой скорости работы на больших коллекциях текстов.

9. Нечёткая кластеризация

Многие плоские чёткие алгоритмы имеют нечеткие вариации: нечёткий метод c-средних (fuzzy c-means — FCM), нечёткие кластеризационные сети Кохонена (fuzzy Kohonen clustering networks — FKCN), алгоритмы на основе нейронной сети по нечёткой теории адаптивного резонанса (unsupervised Fuzzy Adaptive Resonance Theory — Fuzzy-ART) и т. д.

В литературных источниках о нечёткой кластеризации наиболее популярными и эффективными считаются **нечёткий метод c-средних** (вариация чёткого метода k-средних) и его модификации [6, 18, 19].

Целевой функцией алгоритма c-средних является следующая функция:

$$J_m = \sum_{i=1}^{c} \sum_{j=1}^{n} \mu_{ij}^m ||x_j - c_i||^2, \qquad (7)$$

где $\mu_1, \ldots, \mu_c$ — нечёткое c-разделение,
$\mu_{ij} = \mu_i(x_j)$,
$\mu_i \in [0, 1]$,
$\sum_{i=1}^{c} \mu_i(x) = 1$ для всех x из X,
m — степень нечёткости ($m > 1$).

Что касается степени нечёткости, то чем она больше, тем более "размазанной" становится конечная матрица μ_{ij}, т. е. все объекты будут принадлежать всем кластерам с одной и той же степенью. Кроме того, степень нечёткости позволяет при формировании координат центров кластеров усилить влияние объектов с большими значениями степеней принадлежности и уменьшить влияние объектов с малыми значениями степеней принадлежности [20].

Нахождение матрицы нечёткого разбиения с минимальным значением целевой функции представляет собой задачу нелинейной оптимизации.

В основу метода c-средних для решения этой задачи положен метод неопределенных множителей Лагранжа. Он позволяет найти локальный оптимум, поэтому выполнение алгоритма из различных начальных точек может привести к разным результатам.

СРАВНИТЕЛЬНАЯ ХАРАКТЕРИСТИКА МЕТОДОВ ТЕКСТОВОЙ КЛАСТЕРИЗАЦИИ

В реальных задачах автоматической кластеризации выбор алгоритма целесообразно производить исходя из известных особенностей кластеризуемых данных и целей конкретной кластеризации. Следовательно, для качественного подбора алгоритма необходимо иметь представление о разнообразных характеристиках алгоритмов. Попытаемся выделить наиболее существенные.

Временная сложность

Оценки вычислительной сложности основных методов кластеризации приведены в следующей таблице (где n — количество документов, m — размерность пространства документов, p — размерность редуцированного пространства документов):

Метод кластеризации	Вычислительная сложность
Single-Llink	$O(n^2)$
Complete-Link	$O(n^2\log(n))$
Group-Average	$O(n^2)$
Суффиксное дерево	$O(n)$
Метод k-средних	$O(n)$
MST (минимальное остовное дерево)	$O(n^2\log(n))$
EM-алгоритм	$O(n)$
DBSCAN	$O(n\log(n))$
SOM (самоорганизующиеся карты)	$O(mn^2)$
LSA (*латентно-семантический анализ*)	$O(p^3n^2)$
FCM (нечёткий метод c-средних)	$O(n)$

Очевидно, что если время выполнения алгоритма является первостепенной задачей, то более предпочтительны плоские алгоритмы. Высокая временная сложность является основным недостатком методов, анализирующих семантику документов.

Задание числа кластеров

В отличие от иерархических алгоритмов для корректного выполнения плоских алгоритмов необходимы *априорные догадки относительно числа* кластеров. И может показаться, что отсутствие необходимости определения количества кластеров при иерархических алгоритмах является достоинством последних. Однако для получения удобного в использовании набора иерархических кластеров полное дерево часто усекают. А для выполнения этого необходимо задать желаемое количество кластеров или значение меры сходства, при котором связи дерева отсекаются. Поэтому нет существенной разницы между иерархическим и плоским алгоритмом в данном вопросе. Пожалуй, можно выделить только алгоритм DBSCAN, который самостоятельно определяет число кластеров.

Сходимость алгоритма

В методах k-средних и c-средних, если центроиды кластеров выбираются случайным образом, результаты, получаемые на одной и той же выборке документов, будут отличаться. Это может происходить в большей степени по причине локальной сходимости положенного в основу математического аппарата и в меньшей степени по причине неудовлетворительной работы генератора случайных чисел. То же характерно и для EM-алгоритма, который, даже если параметры неплохо инициализированы, обычно попадает в одну из локальных точек минимума, кроме того, скорость схождения данного алгоритма может быть очень медленной.

Для нейросетевых алгоритмов характерны проблемы, связанные со сложностью построения и длительностью обучения, и отсутствие гарантии, что процесс обучения сети определенной структуры не остановится, не достигнув допустимого порога ошибки, или не попадет в локальный минимум.

Для генетических алгоритмов актуальна проблема преждевременного схождения.

Важность порядка поступления документов

Для некоторых алгоритмов характерна зависимость конечного набора кластеров от порядка, в котором анализировались документы. Яркими примерами таких алгоритмов являются ART1 и SOM.

Чувствительность к инициализации параметров

Результат выполнения большинства алгоритмов кластеризации зависит от инициализации входных параметров (помимо задания количества кластеров). Особенно чувствительными являются EM-алгоритм (инициализация вектора параметров распределений), генетические алгоритмы (размер популяции, вероятности кроссовера и мутации и т. п.), DBSCAN (параметры оценки плотности) и нейросетевые алгоритмы (параметры обучения).

Чувствительность к шумам и выбросам

Важной характеристикой алгоритма кластеризации является его способность аккуратно обрабатывать документы, представляющие собой выбросы по отношению к данной конкретной коллекции. Например, метод k-средних слишком чувствителен к выбросам, поскольку они могут искажать среднее и сильное влиять на распределение кластеров. И в EM-алгоритме присутствие шума оказывает сильное влияние на распределение кластеров. Степень влияния для данных алгоритмов зависит от начального распределения кластеров. Алгоритм MST так же подвержен сильному влиянию выбросов, поскольку возможно порождение множества кластеров, состоящих из одного вектора, появляющихся на начальных этапах разбиения дерева. Чувствительность к выбросам является одним из основных недостатков метода главных компонент, так как из-за больших незамеченных выбросов в данных может быть искажена оценка ковариационной матрицы.

Наглядность результата и возможность получить детальное представление о структуре данных

Для некоторых исследований коллекций документов бывают важны удобство интерпретации и наглядность представления результатов кластеризации. В таких ситуациях наиболее подходящими считаются иерархические методы, в частности суффиксное дерево, которое кроме древовидной структуры предоставляет понятные человеку названия кластеров (фрагменты текстов и фраз), а также SOM как один из методов визуализации данных.

Ограничения для набора данных

Не все алгоритмы кластеризации способны работать с большими наборами данных. В первую очередь это связано со временем выполнения таких алгоритмов. Так, например, методы Single-link, Group-average, Complete-link, LSA и МГК могут очень медленно работать на больших базах данных. Качественное обучение нейронной сети так же возможно только для ограниченного и относительно малого количества объектов. Вариантом решения этой проблемы является использование выборки данных.

Пересекаемость кластеров

При кластеризации текстовых документов бывают ситуации, когда один и тот же документ отражает несколько тематик, лежит на границе между кластерами, тогда может появиться необходимость указать этот факт в результатах разбиения. Одним из способов отразить такую ситуацию могут быть пересекающиеся кластеры. Как правило, пересекающиеся кластеры допустимы в результатах выполнения только нечётких методов, например нечёткого метода c-средних. Считается, что для большинства проблем ОЕЯ нечёткая кластеризация является более подходящей, чем чёткая.

Анализ скрытого родства документов

Для качественной кластеризации текстов необходимо учитывать проблемы обработки естественного языка, например, синонимию и омонимию. В классическом варианте почти все описанные методы кластеризации неспособны выявлять скрытую семантику документов, что приводит к некорректному разбиению данных. Эту задачу пытаются решать только методы понижения размерности — путём анализа пространства признаков документов.

ОЦЕНКА КАЧЕСТВА РАЗБИЕНИЯ ДОКУМЕНТОВ МЕТОДАМИ КЛАСТЕРИЗАЦИИ

Все описанные выше алгоритмы кластеризации теоретически могут быть применены для задачи кластеризации текстов. Однако разные алгоритмы имеют тенденцию показывать различные кластеризационные результаты, и даже один и тот же алгоритм может показать различные варианты разбиения данных в зависимости от настройки входных параметров, например, числа кластеров. Естественно попытаться определить качество различных способов разбиения заданной совокупности элементов на классы, т. е. найти тот количественный критерий, следуя которому можно было бы предпочесть одно разбиение другому. В задаче кластеризации процедура оценки результатов известна под названием *обоснованность кластерного решения* (cluster validity), а критерии оценки качества разбиения можно разделить на две группы, отражающие два подхода к исследованию обоснованности кластеров в зависимости от используемых данных [21, 22]:

1) Внешние критерии: сравнение полученных результатов классификации с привлечёнными извне "эталонными" результатами классификации этих же данных, полученными, например, на основе мнения экспертов.

2) Внутренние критерии: анализ внутренних свойств, присущих конкретному набору данных, для оценки качества классификации.

Внешние критерии качества разбиения коллекции документов

Рассмотрим две основные группы внешних критериев оценки качества кластеризации.

Классические критерии оценки систем информационного поиска

Традиционными мерами оценки производительности систем информационного поиска являются точность и полнота, а также их объединённая мера F_γ-меры. В классической трактовке точность и полнота отвечают на следующие вопросы: сколько в ответ на запрос система выдала релевантных документов среди всех выданных документов и сколько выдано документов среди всех возможных релевантных документов.

При условии, что для исследования доступно "эталонное" разбиение текстовой коллекции, полнота и точность также могут быть применимы для оценки алгоритмов кластеризации. Тогда для полученного системой кластера α и эталонного кластера β определим точность ($P_{\alpha\beta}$), полноту ($R_{\alpha\beta}$) и F_γ-меру следующим образом [23]:

$$P_{\alpha\beta} = \frac{\text{çîëč÷ íñâî äîęóěíňîâ čç çëŕíńäŕ } \beta \text{ â çëŕíńäŕ } \alpha}{\text{îáůĺĺ çîëč÷ íñâî äîęóěíňîâ â çëŕíńäŕ } \alpha},$$

$$R_{\alpha\beta} = \frac{\text{çîëč÷ íñâî äîęóěíňîâ čç çëŕíńäŕ } \beta \text{ â çëŕíńäŕ } \alpha}{\text{îáůĺĺ çîëč÷ íñâî äîęóěíňîâ â çëŕíńäŕ } \beta},$$

$$F^{\gamma}_{\alpha\beta} = \frac{(\gamma^2+1)\cdot P_{\alpha\beta}\cdot R_{\alpha\beta}}{\gamma^2\cdot P_{\alpha\beta} + R_{\alpha\beta}},$$

где γ — параметр, контролирующий относительный вес точности и полноты (для равного участия γ принимают равной 1). Индивидуальные меры $P_{\alpha\beta}$, $R_{\alpha\beta}$ и $F_{\alpha\beta}$ усредняют для получения общих оценок меры P, R и F. Заметим, что значения этих мер лежат в диапазоне от 0 до 1, и чем ближе значения мер к единице, тем точнее отражено "эталонное", "ручное" разбиение данных в разбиении, полученном программно.

Другие критерии оценки сходства двух разбиений набора данных

Составим для каждой пары (x_i, x_j) в полученном разбиении C и "эталонном" разбиении P таблицу типа:

Разбиение данных $\{(x_i, x_j)\}$	Принадлежат одному кластеру в P	Принадлежат разным кластерам в P
Принадлежат одному кластеру в C	a	c
Принадлежат разным кластерам в C	b	d

Теперь на основе данной таблицы вычислим значения некоторых индексов; чем больше будут значения этих индексов, тем выше сходства разбиений C и P. Такие индексы широко описаны в литературных источниках, например в [22, 24]. Поэтому здесь мы приведём лишь два примера индексов:

Rand-статистика:

$R = \dfrac{a+d}{a+b+c+d}$ — вероятность принятия одинаковых решений относительно принадлежности документа кластеру в разных разбиениях.

Индекс Folkes и Mallows:

$$FM = \sqrt{\frac{a}{a+b} \cdot \frac{a}{a+c}}, \qquad (8)$$

где $a/(a+b)$ - вероятность, что два документа принадлежат одному кластеру в C, если они принадлежат также одному кластеру в P, $a/(a+c)$ -- вероятность, что два документа принадлежат одному кластеру в P, если они принадлежат одному и тому же кластеру в C.

Главными недостатками подхода с использованием внешних критериев являются субъективность "эталонного" разбиения, полученного на основе экспертного мнения специалистов, а также необходимость затрат на получение и обработку этого мнения.

Внутренние критерии качества разбиения коллекции документов

Как только разделение данных получено путём выполнения алгоритма кластеризации, анализ внутренних показателей качества может помочь выяснить, аккуратно ли алгоритм представил структуру данных. Анализ внутренних критериев качества разбиения основан на предположении, что оптимальная кластеризационная схема обладает следующими свойствами [22]:

а) компактность: члены одного кластера должны быть настолько близкими друг другу, насколько это возможно; обычной мерой компактности является дисперсия, которая должна быть минимальной;

б) отделимость: сами кластеры должны далеко отстоять друг от друга.

Поэтому при определении многих индексов обоснованности вычисляют меж- и внутрикластерные расстояния.

Внутренние показатели качества часто используются для поиска оптимального количества кластеров для настройки алгоритмов, требующих задания числа кластеров в качестве входного параметра, если исследователю недоступна априорная информация об их числе.

Допустим, что коллекция текстовых документов по своей природе разбиваема на k кластеров ($k > 1$), тогда рассмотрим распространённые внутренние показатели качества.

Внутренние показатели качества жёсткого разбиения

Традиционными мерами обоснованности жёстких кластеров текстовых документов являются следующие [24–27]:

Индекс Dunn, модифицированный в [26] для снижения чувствительности к шуму:

$$DI(C) = \frac{\min\limits_{i \neq j}\{\delta(C_i, C_j)\}}{\max\limits_{1 \le l \le k}\{\Delta(C_l)\}}, \qquad (9)$$

где $C = \{C_1, \ldots, C_k\}$ — результат кластеризации множества документов D, c_i — центр кластера C_i,

$$\delta(C_i, C_j) = \frac{1}{|C_i||C_j|} \sum_{x \in C_i, y \in C_j} d(x, y) \qquad (10)$$

— мера расстояния между кластерами,

$$\Delta(C_i) = 2\left(\frac{\sum_{x \in C_i} d(x, c_i)}{|C_i|}\right)$$

— мера диаметра кластера.

Целью анализа является максимизация индекса Dunn.

Мера Davies-Bouldin, являющаяся функцией отношения суммы внутрикластерного разброса к межкластерному разделению:

$$DB = \frac{1}{k} \cdot \sum_{i=1}^{k} R_i, \qquad (11)$$

$$R_i = \max_{j=1,\ldots,n,\ i \neq j} R_{ij} \text{ и } R_{ij} = \frac{(s(C_i) + s(C_j))}{\delta(C_i, C_j)},$$

где c_i — центроид кластера C_i,

$s(C_i) = \frac{1}{|C_i|} \sum\limits_{x \in C_i} \|x - c_i\|$ — мера разброса внутри кластера,

$\delta(C_i, C_j) = \|c_i - c_j\|$ — межкластерное расстояние как расстояние между центроидами.

Поскольку малый разброс объектов и высокое расстояние между кластерами приводит к низким значениям R_{ij}, целью является минимизация меры DB.

Индекс Calinski Harabasz (CH-индекс):

$$CH = \frac{traceB/(k-1)}{traceW/(n-k)}, \qquad (12)$$

где B и W являются матрицами меж- и внутрикластерного разброса.

След матрицы межкластерного разброса B можно записать следующим образом:

$$traceB = \sum_{i=1}^{k} n_i \|c_i - c\|^2, \qquad (13)$$

где n_i — число точек в i-м кластере, c - центроид всего набора данных.

След матрицы внутрикластерного разброса W можно записать следующим образом:

$$traceW = \sum_{j=1}^{k} \sum_{i=1}^{n_i} \|x_i - c_j\|^2. \qquad (14)$$

В формуле (12) числитель измеряет, насколько центроиды кластеров отличаются от среднего документа, а знаменатель показывает, насколько документы отличаются от центроидов их кластеров. Следовательно, оптимальное разбиение будет найдено при максимизации *CH*-индекса.

I-**индекс** вычисляется по формуле:

$$I(K) = \left(\frac{1}{k} \times \frac{E_1}{E_k} \times D_k\right)^p, \qquad (15)$$

где

$$E_k = \sum_{i=1}^{k}\sum_{j=1}^{n} u_{ij}\|x_j - c_i\|,$$

$$D_k = \max_{i,j=1}^{k} \|c_i - c_j\|,$$

$\lfloor u_{ij} \rfloor_{k\times n}$ — матрица разбиения данных.

Из (15) следует, что фактор $\frac{1}{k}$ предназначен для сокращения I-индекса при росте k, фактор $\frac{E_1}{E_k}$ – для увеличения I-индекса при росте k, а фактор D_k измеряет максимальную отделимость между двумя кластерами среди всех возможных пар кластеров и увеличивается вместе со значением k. Таким образом, эти три фактора сбалансированы друг другом. Степень p используется для контроля контраста между различными конфигурациями кластеров, например в [27] $p = 2$. Оптимальное разделение данных достигается при максимизации I-индекса.

Внутренние показатели качества нечёткого разбиения

Распространёнными мерами обоснованности кластеров, применяемыми для оценки нечёткой кластеризации текстовых документов, являются [19], где c — это количество кластеров, a_i — центроид i-го кластера.

Индексы, использующие только нечёткую принадлежность:

a) коэффициент разделения (PC):

$$PC(c) = \frac{1}{n}\sum_{i=1}^{c}\sum_{j=1}^{n}\mu_{ij}^2, \qquad (16)$$

где c — количество кластеров; $\frac{1}{c} \le PC(c) \le 1$.

Оптимальное качество разбиения данных достигается посредством решения $\max\limits_{2\le c\le n-1} PC(c)$.

b) энтропия разделения *(PE)*:

$$PE(c) = -\frac{1}{n}\sum_{i=1}^{c}\sum_{j=1}^{n}\mu_{ij}\log_2\mu_{ij}, \qquad (17)$$

где $0 \le PE(c) \le \log_2 c$.

Оптимальное качество разбиения данных достигается посредством решения $\min\limits_{2\le c\le n-1} PE(c)$.

c) модифицированный индекс коэффициента разделения (MPC):

$$MPC(c) = 1 - \frac{c}{c-1}(1 - PC(c)), \qquad (18)$$

где $0 \le MPC(c) \le 1$.

Индексы PC и PE имеют монотонную тенденцию развития с изменением c. MPC сокращает эту тенденцию. Оптимальное качество разбиения данных достигается посредством решения $\max\limits_{2\le c\le n-1} MPC(c)$.

Индексы, учитывающие и нечёткую принадлежность, и структуру данных:

a) критерий обоснованности, предложенный Fukuyama и Sugeno (FS):

$$FS(c) = \sum_{i=1}^{c}\sum_{j=1}^{n}\mu_{ij}^m\|x_j - a_i\|^2 - \sum_{i=1}^{c}\sum_{j=1}^{n}\mu_{ij}^m\|a_i - \overline{a}\|^2 =$$

$$= J_m(\mu, a) + K_m(\mu, a), \qquad (19)$$

где $\overline{a} = \sum\limits_{i=1}^{c}\frac{a_i}{c}$,

$J_m(\mu, a)$ — мера компактности,

$K_m(\mu, a)$ — мера отделимости.

Оптимальное качество разбиения данных достигается посредством решения $\max\limits_{2\le c\le n-1} FS(c)$.

b) критерий обоснованности, предложенный Xie и Beni (1991) при $m = 2$ и модифицированный Pal и Bezdek (1995) (XB):

$$XB(c) = \frac{\sum\limits_{i=1}^{c}\sum\limits_{j=1}^{n}\mu_{ij}^m\|x_j - a_i\|^2}{n\min_{i,j}\|a_i - a_j\|^2} = \frac{J_m(\mu, a)/n}{Scp(a)}, \qquad (20)$$

где $J_m(\mu, a)$ — мера компактности,

$Sep(a)$ — мера отделимости.

Оптимальное качество разбиения данных достигается посредством решения $\max\limits_{2\le c\le n-1} XB(c)$. В работе [20] именно этот критерий рекомендуется использовать для оценки результата алгоритма нечётких c-средних.

c) критерий обоснованности, предложенный Zahid (1999) (SC):

$$SC(c) = SC_1(c) - SC_2(c), \qquad (21)$$

$$SC_1(c) = \frac{\sum\limits_{i=1}^{c}\|a_i - \overline{a}_i\|^2/c}{\sum\limits_{i=1}^{c}\left(\sum\limits_{j=1}^{n}\mu_{ij}^m\|x_j - a_i\|^2 \Big/ \sum\limits_{j=1}^{n}\mu_{ij}\right)}, \qquad (22)$$

$$SC_2(c) =$$

$$= \frac{\sum\limits_{i=1}^{c-1}\sum\limits_{l=i+1}^{c}\left(\sum\limits_{j=1}^{n}(\min(\mu_{ij}, \mu_{lj}))^2\right) \Big/ \sum\limits_{j=1}^{n}\min(\mu_{ij}, \mu_{lj})}{\sum\limits_{j=1}^{n}\left(\max\limits_{1\le i\le c}\mu_{ij}\right)^2 \Big/ \sum\limits_{j=1}^{n}\max\limits_{1\le i\le c}\mu_{ij}}, \qquad (23)$$

где оба коэффициента (SC_1 и SC_2) измеряют компактность и отделимость.

SC_1 рассматривает геометрические свойства структуры данных, а SC_2 - нечёткую принадлежность. Оптимальное качество разбиения данных достигается посредством решения $\min\limits_{2\le c\le n-1} SC(c)$.

d) критерий обоснованности нечёткого гипер-объёма, предложенный Gath и Geva (1989) (FHV):

$$FHV(c) = \sum_{i=1}^{c} [\det(F_i)]^{1/2}, \tag{24}$$

где

$$F_i(c) = \frac{\sum\limits_{j=1}^{n} (\mu_{ij})^m (x_j - a_j)(x_j - a_i)^T}{\sum\limits_{j=1}^{n} \mu_{ij}^m}, \tag{25}$$

F_i — нечёткая ковариационная матрица кластера i. (26)

Оптимальное качество разбиения данных достигается посредством решения $\min\limits_{2 \le c \le n-1} FHV(c)$.

*

В ряде исследований, например в [27, 28], наиболее эффективным критерием обоснованности был признан I-индекс. Однако важно отметить, что какими бы надёжными ни были меры обоснованности на одних данных, на других данных их поведение может существенно измениться. Такой подход к оценке обоснованности разбиения данных хорошо работает при условии, что достоверные кластеры являются компактными, плотными и хорошо разделимыми. Однако существует множество областей применения, где данные разбиваются на кластеры произвольной формы. В таких ситуациях традиционные основы критериев обоснованности (дисперсия, плотность, отделимость) не являются достаточными. В работе [29] показано, что в общем случае к таким ситуациям относятся и коллекции текстовых документов, поскольку текстовые документы представляются высокоразмерными разреженными векторами. В таком пространстве объектов сходство между документом и центроидом в общем случае является низким и поэтому трудно ожидать компактных кластеров. А следовательно, необходим поиск иных критериев, более пригодных, чем традиционные внутренние критерии качества разбиения данных.

ЗАКЛЮЧЕНИЕ

Разнообразие традиционных кластеризационных подходов позволяет выбрать оптимальный метод для применения в конкретных условиях поставленной задачи. При необходимости данный метод возможно модифицировать с учётом специфики кластеризуемых данных или дополнить систему специфической предобработкой данных, поступающих на вход метода кластеризации. Последнее особенно важно для обработки текстовых документов. Связано это с тем, что для получения качественного результата кластеризации необходим предварительный анализ пространства признаков документов. Данный анализ позволил бы, во-первых, выявить скрытые родственные связи между документами и их признаками, во-вторых, сократить количество шумовых признаков. В результате итоговым пространством документов стало бы пространство сокращённой размерности с выявленными семантическими проблемами анализируемых текстов, что повысило бы и качество, и скорость выполнения традиционных методов кластеризации текстовых документов.

СПИСОК ЛИТЕРАТУРЫ

1. Пескова О. В. Методы автоматической классификации текстовых электронных документов // НТИ. Сер. 2. – 2006.— № 3.— С. 13-20.

2. Van Rijsbergen C. J. Information retrieval / Department of Computing Science, University of Glasgow.— London: Butterworths, 1979; См. также: http://www.dcs.gla.ac.uk/Keith/Preface.html

3. Manning D., Schutze H. Foundations of statistical natural language processing.-- The MIT Press, 2003.

4. Кириченко К. М., Герасимов М. Б. Обзор методов кластеризации текстовой информации // Материалы международной конференции "Диалог'2001" [Электрон. ресурс]. 2001.— Режим доступа: http://www.dialog-21.ru/materials/archive.asp?id=6912&y=2001&vol=6078

5. Eli Zamir O. Clustering Web Documents: A Phrase-Based Method for Grouping Search Engine Results [Электрон. ресурс] / University of Washington, Department of Science & Engineering. - 1999.-- Режим доступа: http://www.cs.washington.edu/research/projects/WebWarel/www/metacrawler/

6. Jain A. K., Murty M. N., Flynn P. J. Data Clustering: A Review // ACM Computing Surveys. 1999.— Vol. 31, № 3.

7. Ester M., Kriegel H.-P., Sander J., Xu X. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise // Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96). – Portland: AAAI Press, 1996; См. также: http://ifsc.ualr.edu/xwxu/

8. Nurminen M., Honkaranta A., Karkkainen T. ExtMiner: combining multiple ranking and clustering algorithms for structured document retrieval // Proceedings of Database and Expert Systems Applications, 2005. Sixteenth International Workshop on.— 2005. – P. 1036-1040.

9. Lampos C., Eirinaki M., Jevtuchova D., Vazirgianni M. Archiving the Greek Web // Proceedings of 4th International Web Archiving Workshop (IWAW04) [Электрон. ресурс].- 2004. Режим доступа: http://www.iwaw.net/04/

10. Джонс М. Т. Программирование искусственного интеллекта в приложениях / Пер. с англ. А. И. Осипова.— М.: ДМК Пресс, 2004.

11. Kohonen T., Kaski S., Lagus K., SalojArvi J., Honkela J., Paatero V., Saarela A. Self organization of a massive document collection // IEEE Transactions of neural networks. 2000.— Vol. 11, № 3.

12. Стариков А. Самоорганизующиеся карты математический аппарат [Электрон. ресурс]. - 2000. - Режим доступа: www.basegroup.ru/neural/som.htm

13. Kanade P. M., Hall L. O. Fuzzy Ants as a Clustering Concept // 22nd international conference of the North American fuzzy information processing society (NAFIPS-2003).-- 2003. - P. 227 232; См. также: http://morden.csee.usf.edu/ailab/hall.html

14. Pakhira M. K., Bandyopadhyay S., Maulik U. A study of some fuzzy cluster validity indices, genetic clustering and application to pixel classification // Fuzzy Sets and Systems. - 2005. - Vol. 155. P. 191 214.

15. Landauer T. K., Foltz P. W., Laham D. Introduction to Latent Semantic Analysis // Discourse Processes.— 1998.- Vol. 25. P. 259 -284.

16. Singular value decomposition and principal component analysis // A Practical Approach to

Microarray Data Analysis / Eds. D. P. Berrar, W. Dubitzky, M. Granzow.— Kluwer, 2003.— P. 91–109.

17. Прикладная статистика: Классификация и снижение размерности· Справ. изд. / С. А. Айвазян, В. М. Бухштабер, И. С. Енюков, Л. Д. Мешалкин; Под ред. С. А. Айвазяна.— М.: Финансы и статистика, 1989.

18. Tsekouras G. E. On the use of the weighted fuzzy c-means in fuzzy modeling // Advances in Engineering Software. – 2005.-- Vol. 36.— P. 287–300.

19. Kuo-Lung Wu, Miin-Shen Yang. A cluster validity index for fuzzy clustering // Pattern Recognition Letters.— 2005. – Vol. 26.— P. 1275 1291.

20. Штовба С. Д. Введение в теорию нечетких множеств и нечеткую логику [Электрон. ресурс].— Режим доступа: http://matlab.exponenta.ru/fuzzylogic/book1/index.php

21 Гусарова Л., Яцкив И. Проверка обоснованности кластерного решения // Proceedings in "RELIABILITY and STATISTICS in TRANSPORTATION and COMMUNICATION (RelStat'03)".— 2003.— Vol. 2.— P. 49–56.

22. Halkidi M., Batistakis V. S., Vazirgiannis M. On Clustering Validation Techniques // Journal of Intelligent Information Systems.— 2001.— Vol. 17, № 2–3.— P. 107–145.

23. Mendes M. E. S., Sacks L. Dynamic Knowledge Representation for e-Learning Applications // Proc. of the 2001 BISC International Workshop on Fuzzy Logic and the Internet (FLINT'2001), Memorandum No. UCB/ERL M01/28.— University of California Berkeley, 2001.— P. 176–181; См. также: http://www.ee.ucl.ac.uk/~mmendes/index_ucl.htm

24. Boutin F., Hascoët M. Cluster Validity Indices for Graph Partitioning // Proceedings of the Eight International Conference on Information Visualization (IV'04). IEEE.- 2004.

25. Stein B., zu Eissen S. M., Wißbrock F. On Cluster Validity and the Information Need of Users // 3rd IASTED Int. Conference on Artificial Intelligence and Applications (AIA 03). Spain.— ACTA Press, 2003.— P. 216–221.

26. Bezdek J. sC., Pal N. R. Some New Indexes of Cluster Validity // IEEE Transactions on Systems, Man and Cybernetics. Part B. Cybernetics.— 1998. – Vol. 28, № 3.

27. Maulik U., Bandyopadhyay S. Performance Evaluation of Some Clustering Algorithms and Validity Indices // IEEE Transactions on Pattern Analysis and Machine Intelligence.— 2002.– Vol. 24, № 12.

28. Lam B. S. Y., Yan H. A new cluster validity index for data with merged clusters and different densities // Systems, Man and Cybernetics, 2005. IEEE International Conference on.-- 2005.— Vol. 1. – P. 798 803.

29. Пескова О. В. Автоматическое формирование тематической схемы коллекции документов // Технологии Microsoft в теории и практике программирования: Труды Всероссийской конференции студентов, аспирантов и молодых учёных. – М.: МГТУ им. Н. Э. Баумана, 2006.

30. Roth V., Lange T., Braun M., Buhmann J. A Resampling Approach to Cluster Validation / Eds. Wolfgang Härdle, Bernd Rönz // Proceedings in Computational Statistics: 15th Symposium Held in Berlin (COMPSTAT2002). Germany. Heidelberg: Physica-Verlag, 2002.— P. 123–128.

31. Halkidi M., Batistakis Y., Vazirgiannis M. Cluster validity methods. Part 1 // Source ACM SIGMOD Record.-- New York: ACM Press, 2002.— Vol. 31, Issue 2.-- P. 40–45.

32. Halkidi M., Batistakis Y., Vazirgiannis M. Clustering validity checking methods. Part 2 // Source ACM SIGMOD Record. New York: ACM Press, 2002.-- Vol. 31, Issue 3. - P. 19 27.

Материал поступил в редакцию 22.09.06.

А. Б. Вольфтруб, А. К. Поликарпов

Методы управления рисками многофакторного ущерба в автоматизированных информационных системах

Статья описывает один из методов, используемых для оценки рисков, а также определяет направление дальнейших исследований в этой области.

Проблема управления рисками информационной безопасности не нова. В современном мире информация является одной из важнейших основ любого бизнес-процесса. Теория защиты информации вводит понятия риска нарушения информационной безопасности и управления рисками, т. е. прогнозирования и оценки влияния того или иного риска на систему, а также определение мер, позволяющих минимизировать его.

Для автоматизации решения подобных задач разрабатываются различные программные комплексы. Математическая модель, описываемая в этой статье, реализована в системе автоматизации управления рисками "АванГард", разработанной в Институте системного анализа РАН [1–6]. Этот программный комплекс успешно используется различными предприятиями.

ИСПОЛЬЗУЕМЫЕ ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

Описываемые в статье методы применяются для управления рисками в компьютерных автоматизированных информационных системах, поэтому термины "система", "риск", "технология", "компонент", "объект" следует рассматривать в применении к данной области.

Преимущества компьютерных технологий очевидны. Вместе с тем, нужно четко понимать, что их использование порождает и новые риски, которые необходимо знать и оценивать, добиваясь их снижения. Риски могут возникать в связи с тем, что технологии недостаточно надежны, уязвимы и/или применяются с нарушением правил (требований) их безопасного использования. Особенно