Д.А. Кушнарев (Минск, БГУ)

КЛАССИФИКАЦИЯ АЛГОРИТМОВ КЛАСТЕРИЗАЦИИ ТЕКСТОВЫХ ДОКУМЕНТОВ

В связи с постоянно возрастающим объемом информации, доступной в электронном виде, все большую актуальность приобретают методы автомати–ческой ее обработки, к которым, в частности, относится и кластеризация текстов.

Предметом данной работы является автоматическая кластеризация текстов. *Объектом* является классификация различных подходов, применяемых при кластеризации текстов.

1. Понятие кластеризации текста

Кластеризацией, или неконтролируемой классификацией текстовых документов (англ. clustering, unsupervised document classification) называется процесс распределения множества текстовых документов по группам на основании схожести их содержания, причем группы также должны определяться автоматически, этим кластеризация отличается от контролируемой классификации.

При рассмотрении процесса кластеризации текстовых документов можно выделить две практически независимые задачи:

• предварительное преобразование текста в промежуточное представление — модель документа, и расчет схожести (близости) документов между собой на основании этой модели;

• кластеризация текстовых документов на основании их схожести.

Традиционно в качестве модели документа использовалось линейное векторное пространство. Каждый документ представлялся как вектор, то есть массив всех и наиболее часто встречающихся слов, а затем для расчета использовалось, например, евклидово расстояние [1].

1.1 Вспомогательные задачи

Кроме описанных выше задач, надо также отметить необходимость преобразования документа в доступный для обработки вид, что обычно представляет собой *фильтрацию* документа с целью удалить информацию о его разметке (например, HTML-тегов).

Кроме того, возможны дополнительные преобразования, которые позволяют увеличить точность кластеризации:

• *Приведение слов к основе (stemming).* Позволяет привести лексемы к какой-либо единой форме, например, путем отбрасывания окончаний, что позволяет повысить точность кластеризации.

• *Удаление частотных слов (stopword removal).* Самые частотные слова, встречающиеся в документах всех типов, не представляют интереса для кластеризации, так как они зачастую зависят не столько от темы документа, сколько от стиля, в котором он написан, поэтому их удаление позволяет повысить точность кластеризации.

• *Латентная семантическая индексация.* Предполагает объединение слов-синонимов в одно измерение вектора. Для данного алгоритма требуется лексическая база данных (например, в этой роли часто используется WordNet, которая позволяет повысить точность результатов, но могут возникнуть трудности из-за омонимии [4].

• *Выделение ключевых слов.* Предполагает выделение основных слов в тексте и дальнейшую работу именно с ними. Данный подход применяется для кластеризации текстов большого объема, когда выполнять кластеризацию по всему тексту неэффективно из-за чрезмерного расхода вычислительной мощности. На практике применяются различные алгоритмы данного метода, в частности, предполагающие выделение ключевых словосочетаний [5].

2. Модели текста и способы расчета расстояния между документами

Для расчета расстояния между двумя текстами, необходимо преобразовать их в промежуточное представление, которое позволяет вычислить степень схожести или различий.

Можно выделить два основных вида моделей: учитывающие частоту словоупотреблений и учитывающие порядок слов. Можно использовать два варианта этих моделей одновременно, чтобы учитывать оба данных фактора.

2.1. Векторное представление

Векторное представление представляет собой массив всех или наиболее часто используемых слов документа или n-граммов, то есть последовательностей нескольких слов.

Каждое слово представляет собой измерение вектора.

Часто после преобразования всех документов производится удаление слов, которые встречаются в корпусе лишь несколько раз. В основе данного шага лежит предположение, что такие слова, даже если они и значимы, привели бы в формированию кластеров малого размера.

Особенностью векторной модели является то, что в ней никаким образом не указывается порядок слов, несмотря на то, что он может быть важен, n-граммные векторы позволяют несколько уменьшить эту проблему.

По векторному представлению никак невозможно восстановить первоначальный текст, чтобы вывести его пользователю, — можно отобразить только самые частые слова в определенном кластере [1; 3].

2.2. Индексный граф документа

Данное представление позволяет рассчитать расстояние между документами, исходя из количества совпадающих слов, расположенных подряд.

В данной модели каждое слово представляется в виде вершины графа, и каждый документ представляется в виде соединения этих точек в определенном порядке [1].

2.3. Суффиксное дерево

Алгоритм кластеризации с использованием суффиксного дерева не предполагает отдельного этапа по вычислению близости документов, а

использование специально разработанного алгоритма, который позволяет снизить скорость обработки данных.

Как показывает практика, данный алгоритм даёт хорошие результаты при небольших объёмах текстов, но при их увеличении он уступает другим подходам [2].

3. Алгоритмы кластеризации

При рассмотрении алгоритмов кластеризации имеет смысл выделить следующие основания для характеристики:

• *Интерактивность.* Интерактивные алгоритмы кластеризации (online clustering) не требуют повторной обработки при добавлении новых документов, а позволяют производить обработку пошагово.

• *Представляется ли результат в виде иерархии,* т.е. системы вложенных разбиений, или в виде непересекающихся разбиений. Иерархическое представление позволяет облегчить навигацию для пользователя.

• *Нечеткий ли метод,* то есть позволяет ли он включение одного документа в несколько кластеров одновременно, или же только в один определенный кластер.

• *Автоматическое определение количества кластеров.* Некоторые алгоритмы требуют задания числа кластеров заранее, что во многих ситуациях нежелательно.

• *Сложность алгоритма* (О-большое) позволяет определить, как будет увеличиваться время на выполнение алгоритма при увеличении обрабатываемых документов.

Классификацию алгоритмов на основе выделенных характеристик можно представить в виде следующей таблицы:

Таблица

Алгоритм	Инт.	Иерархия	Нечетк.	Нахождение кол-ва кл.	Сложность
Метод суффиксного дерева	+	–	+	+	O(n * logn)
k- средних	±	–	–	–	O(knl)
c-средних	±		+		O(knl)
По одной связи		+		+	$O(n^2)$
По групповому среднему		+		+	$0(n^2)$
По всем связям	–	+	–	+	$0(n^3)$
Метод минимального покрывающего дерева	+	–	±	+	O(n * logn)
Метод выделения связанных компонент	+	–	±	+	

В данной таблице n — число документов, k — число кластеров, l — число итераций, которые необходимы для нахождения решения.

3.2. Рассмотрение отдельных алгоритмов

3.2.1. Метод k -средних и его разновидности

Метод к-средних (k-means) или метод динамических ядер — один из самых простых в реализации методов. Данный метод предполагает предварительное задание числа кластеров.

В начале работы алгоритма выбираются центральные точки кластеров. На каждой итерации такая точка приписывается к тому кластеру, к центральной точке которого он ближе всего расположен. После этого рассчитывается среднее значение координат всех точек, которые вошли в определенный кластер, и это среднее значение и становится центральной точкой данного кластера.

Итерации повторяются до тех пор, пока алгоритм не сойдется.

Разновидности данного алгоритма можно классифицировать по следующим признакам:

• в зависимости от того, как вычисляются центры кластеров: k-means (средние значения), k-median (медианы);

• в зависимости от того, как вычисляютсяначальные значения: случайно, алгоритмом k-means++;

• в зависимости от того, может ли документ входить в несколько кластеров одновременно: четкие и нечеткие.

3.2.2. Алгоритм EM

Метод k-средних является частным случаем алгоритма EM (expectancy–maximisation), который основывается на предположении, что множество можно описать как линейную комбинацию нормальных распределений. Задача алгоритма состоит в нахождении таких параметров (математического ожидания и дисперсии) [2].

3.2.3. Иерархические алгоритмы

Алгоритмы данного типа предполагают на каждой итерации производить объединение двух документов, расположенных ближе всего друг к другу, в одну группу. После этого производится перерасчет расстояний между документами, причем группа уже считается неделимым объектом.

3.2.4. Алгоритмы на основе графов

Документы можно представить как вершины графа, и в таком случае задача кластеризации документов сводится к удалению определенных ребер графа. По такому принципу работают следующие алгоритмы:

- Минимального покрывающего дерева
- Выделения связных компонент (в частности MajorClust).

ЛИТЕРАТУРА

1. Michael Steinbach, George Kapyris, Vipin Kumar A Comparison of Document Clustering Techniqes. University of Minnesota, 2000.

2. Nicholas O. Andrews and Edward A. Fox, Recent Developments in Document Clustering. Department of Computer Science, Virginia Tech., Blacksburg, 2007.

3. Zamir O., Etzioni O. «Web Document Clustering: A Feasibility Demonstration», в Proceedings of ACM SIGIR 98, 1998. С. 46-54.

4. Dan Munteanu, Severum Bumbaru, A Survey of Text Clustering Techniqes used for Web Mining.

5. Henry Anaya-Sánchez, Aurora Pons-Porrata, and Rafael Berlanga-Llavori A document clustering algorithm for discovering topics, Pattern Recognition Letters/ Vol: 31, No: 6, pp: 502-510, April 2010.