КОМПЬЮТЕРНАЯ ЛИНГВИСТИКА И ОБРАБОТКА ЕСТЕСТВЕННОГО ЯЗЫКА

УДК 81’322;004 ISSN 1995-5499
DOI: https://doi.org/10.17308/sait.2020.3/3045
Поступила в редакцию 07.09.2020
Подписана в печать 30.09.2020

КЛАСТЕРИЗАЦИЯ ТЕКСТОВОЙ ВЫБОРКИ, ПАРАМЕТРИЗОВАННОЙ КЛЮЧЕВЫМИ СЛОВАМИ СВОИХ ЭЛЕМЕНТОВ

© 2020 Э. А. Головастова✉[1], Д. Н. Красотин[2]

[1]Московский государственный университет им. М. В. Ломоносова
Ленинские горы, 1, Москва 119991, Российская Федерация
[2]ЗАО «Московский научно-исследовательский телевизионный институт»
ул. Гольяновская, 7а, стр. 1, Москва 105094, Российская Федерация

Аннотация. В данной работе рассмотрено решение задачи кластеризации больших объемов текстовых выборок фиксированной длины с помощью компьютерных средств обработки информации. Автоматическое разделение на группы близких по смыслу текстов является одной из важнейших задач анализа данных, так как имеет очень широкую область применения. Основное внимание в статье уделено скорости выполнения алгоритма. Для этого используется способ представления выборки, использующий в качестве набора признаков документов их ключевые слова, которые есть наиболее важные слова в тексте, набор которых может дать для читателя достаточно полное представление о его содержании. Ключевые слова определяются с помощью предварительно вычисленных значений статистической меры tf-idf, характеризующей важность каждого слова текста именно для рассматриваемого текста. Следующим этапом является непосредственно кластеризация корпуса документов. В данной работе используется модификация метода Dbscan, который является плотностным алгоритм пространственной кластеризации с присутствием шума, но здесь он интерпретируется как разновидность обхода в ширину с некоторыми ограничениями графа выборки документов. Поэтому в данной работе после определения ключевых слов элементов выборки строится инвертированный индекс для словаря корпуса текстов. Далее с помощью найденного инвертированного индекса определяется объект связей документов корпуса, который впоследствии передаётся в качестве аргумента в алгоритм Dbscan. Подобный подход к реализации поставленной задачи выбран из-за предположения о его быстродействии. Для проверки этого предположения проводится замер времени выполнения ключевых операций, значения которого приводятся в качестве иллюстрации результата тестирования предложенного метода кластеризации.
Ключевые слова: кластеризация, текстовая выборка, мера tf-idf, ключевые слова, индексная структура данных, алгоритм Dbscan, скорость выполнения.

ВВЕДЕНИЕ

Кластеризация текстовых документов, или автоматическое разделение на группы близких по смыслу текстов из заданной неструктурированной выборки фиксированной длины, является одной из важнейших задач анализа данных. Область применения кластеризации очень широка: её используют в филологии, психологии, маркетинге, в различных социологических исследованиях и других дисциплинах. На сегодняшний день существует много различных методов для решения задачи кластеризации [1]. При общем рассмотрении эти методы можно раз-

✉ Головастова Элеонора Александровна
e-mail: golovastova.elina@yandex.ru

делить на два типа: первые основываются на представлении текстов в виде векторов в многомерном пространстве признаков и используют некоторую заданную метрику близости между векторами, и другие, которые основываются на иных представлениях кластеризуемых текстов. К последним относятся методы, использующие в качестве признаков документов часто встречающиеся в тексте наборы слов и словосочетаний. Формирование кластеров происходит с помощью графовых методов кластеризации или на основе алгоритмов частичного обучения, использующих предварительно натренированную модель [2]. Также сюда можно отнести методы семантической кластеризации — алгоритмы, основанные на графах содержательных выражений или именованных сущностей с семантическими отношениями между ними [3].

Для первой группы методов процесс кластеризации текстов, как правило, разделяется на два этапа. На первом этапе происходит определение количественных признаков текста и создание вектора признаков, задающего этот текст. На втором этапе на основании значений выбранной метрики на паре векторов происходит формирование кластеров текстов с помощью одного из алгоритмов кластеризации: k-means [4], Dbscan [5], метод иерархической кластеризации и других. Существует несколько различных способов отображения документов в векторное пространство. Наиболее распространённым представлением документов в векторном виде является метод, основанный на так называемым мешке слов — общем словаре, построенном согласно некоторым заданным правилам из терм, встречающихся в выборке текстов. Также выделить признаки из документов можно с помощью методов тематического моделирования. К основным таким методам относятся, например: LSA, pLSA, LDA [6]. Здесь каждый найденный признак в отдельном тексте относится к определенной теме, представленной в выборке. Другой способ параметризации текстов основан на использовании «word embeddings» — векторов слов, полученных с помощью искусственной нейронной сети, натренированной таким образом, что расстояние между словами тем меньше, чем ближе значения этих слов. Наиболее известными моделями являются, например: word2vec [7], GloVe [8], BERT [9].

Особенностью подхода к векторизации документов при помощи «word embeddings» является необходимость предобучения. Нейронная сеть обучается на текстах, содержащих термины, которые будут встречаться в кластеризируемой выборке, и таким образом строит векторную модель словаря. Отметим, что этот процесс имеет некоторую продолжительность и предполагает значительное задействование вычислительной мощности ЭВМ. Однако в условиях практического применения может иметь место требование в максимальном сокращении времени выполнения кластеризации и использовании компьютерных ресурсов, что делает данный подход неэффективным. Также в некоторых случаях этот подход может оказаться вовсе неприменимым в силу отсутствия или малого объёма обучающей текстовой выборки.

Большинство алгоритмов кластеризации принимает в качестве входных данных прямоугольную матрицу, представляющую собой векторную модель выборки в многомерном пространстве признаков, или симметричную квадратную матрицу расстояний, где каждый элемент есть значение метрики на паре соответствующих векторов текстов. Основной проблемой методов, оперирующих такими объектами, является большая размерность пространства признаков, большая часть которых является избыточными для отдельного документа или даже может поспособствовать неверному определению кластера для этого документа. Поэтому зачастую текстовая выборка предварительно подвергается некоторой обработке в целях уменьшения размерности пространства признаков её элементов. Наиболее используемым приёмом является фильтрация так называемых несмысловых слов. Также могут применяться лингвистические методы, выделяющие из текстов с помощью словарей и тезаурусов словоформы и объединяющие их в синонимические группы; лингво-статистические методы, использующие некоторые вероятностные характеристи-

ки о встречаемости терма для включения его в статистически значимые слова всей выборки; а также методы, объединяющие алгебраические и вероятностные приёмы: с помощью алгебраических преобразований размер матрицы векторов текстов уменьшается до некоторого значения с учетом корреляционных связей между терминами.

На начальном этапе обработки текста можно использовать морфологические библиотеки: Lemmatizer, FreeLing, NLTK [10], MCR, tokenizer; а также инструменты синтаксического анализа: Link Grammar Parser, Solarix; как, например, делается в работе [11]. В статье [12] для семантико-синтаксического анализа использовался Stanford CoreNLP Parser, позволяющий получать разметку синтаксических зависимостей в формате «Universal Dependencies».

Недостатком такого подхода является то, что зачастую подобные программные средства поддерживают небольшое число языков, а также они достаточно медленные при обработке большого корпуса текстов.

В работе [13] принцип векторизации документов основан на определении объектов признакового пространства, под которым понимается множество значимых слов и словосочетаний отдельного текста в выборке и на последующем вычислении меры смысловой значимости для каждого понятия. Базовой величиной в формуле расчёта этой меры является статистическая мера tf-idf, принимающая большее значение для понятий с высокой частотой встречаемости в отдельном документе и с относительно низкой частотой во всём текстовом корпусе, и которая далее домножается на так называемые синтаксические и семантические коэффициенты. Эти коэффициенты учитывают вхождение слов или словосочетаний в общий тезаурус, синтаксическую роль слова или словосочетания в предложении, принадлежность терма к фамильно-именной группе и др.

Отметим, что статистическая мера tf-idf даёт оценку важности слова в контексте документа, являющегося элементом корпуса. Получение с её помощью наиболее значимых слов текста не использует инструментов языкового анализа, как-то: программных библиотек или словарей и тезаурусов, замедляющих процесс кластеризации. При подобной параметризации теста не задействована информация о семантических классах выборки, но при требовании быстрого выполнения кластеризации данный способ выделения признаков является вполне приемлемым. Это показано, например, в работе [14], где при векторизации корпуса на основе tf-idf были показаны достаточно высокие показатели скорости и качества различных алгоритмов кластеризации.

Представление текстов в виде векторов в многомерном пространстве даёт матрицу признаков текстовой выборки, которая зачастую довольно громоздкая. Для последующих операций эту матрицу требуется хранить, однако хранение матриц подобного размера потребует много памяти. Также для такого рода параметризации приходится совершать целый ряд действий с многомерными числовыми массивами, что делает кластеризацию вычислительно затратной. Например, в работе [15] алгоритм кластеризации представляет собой разделение гиперсферы, содержащей все вектора текстов выборки, на отдельные области в соответствии со значениями метрики расстояния, которая не должна превышать некоторого значения, зависящего от заданной величины. В основе метода лежит предположение, что при плотном заполнении кластера векторами, его форма стремиться к гипершару, а оболочки кластеров выборки должны образовать так называемый субслой внутри гиперсферы. С помощью методов аналитической геометрии в работе сначала находится расстояние, на котором будут находиться оболочки всех кластеров соответствующего субслоя от центра системы координат, далее определяются координаты узлов сети — центров оболочек кластеров. Затем для каждого вектора слоя попарно вычисляются расстояние от него до узлов сети. Если это расстояние удовлетворяет заданной точности, то вектор относится к соответствующему кластеру.

Зачастую в случае, когда обработка векторов, согласно алгоритму, производится независимо, то вычисления распараллеливают, что позволяет значительно ускорить выпол-

нение, получив тем самым выигрыш в производительности.

Параметризация документа с помощью наборов слов и словосочетаний не вызывает, как правило, проблем многомерности признаков текста и проблемы большого расхода памяти при их хранении. Также если задать документ набором терминов, то можно с помощью индексации обеспечить быстрое выполнение операций с этими объектами. Подобный подход используется в области информационного поиска. Он также может быть задействован и для решения задачи кластеризации. Например, в работе [16] для каждого документа выборки строится так называемый спектральный индекс, элементами которого являются пары — идентификатор лексического дескриптора и его вес в документе, где под лексическим дескриптором понимается заранее выделенная из текста лексема или словосочетание в канонической форме. Спектральный индекс документа далее помещается в базу данных, где в качестве ключа выступает идентификатор документа.

С учётом преимуществ и недостатков подходов, рассмотренных выше, сформулируем цель данного исследования, а именно: реализация быстрой кластеризации неразмеченной текстовой выборки фиксированной длины. Качество метода будет оцениваться по значениям показателей быстродействия, точности и полноты финального отображения кластеров для различного объема текстовых выборок. При этом время выполнения измеряется для всех этапов алгоритма.

1. МЕТОДЫ И МАТЕРИАЛЫ

Для тестирования метода кластеризации в исследовании использовалась неразмеченная выборка из нескольких тысяч документов, каждый из которых является русскоязычным новостным текстом, к которому прибавлено предложение — его заголовок. Алгоритм решения поставленной задачи реализован на языке Python 3.

В качестве предварительной обработки выступает стемминг слов документов с помощью библиотеки NLTK [10]. Также из каждого текста исключаются как часто встречающиеся (например, союзы и предлоги), так и редко встречающиеся слова (например, опечатки), в целях уменьшения влияния слов, не определяющих текст, на результат параметризации.

Исходя из результатов рассмотренных исследований, можно сделать вывод, что одним из наиболее эффективных способов параметризации является подход, где в качестве признаков документа используются наборы слов или словосочетаний его текста. Также отметим, что текст новости вполне характеризуют его ключевые слова, которые есть особо важные, краткие, общепонятные слова в тексте, набор которых может дать для читателя высокоуровневое описание его содержания, обеспечив при этом компактное представление и хранение смысла текста на запоминающем устройстве. Учитывая это, зададим тексты выборки их ключевыми словами в качестве набора признаков.

Получение наиболее значимых слов текста производилось с помощью статистической меры tf-idf, которая предназначена для оценки важности слова в документе, являющегося элементом коллекции или корпуса. Для каждого слова t текста d формула для вычисления его меры tf-idf имеет вид:

$$tf - idf(t, d) = tf(t, d) \cdot idf(t), \qquad (1)$$

Здесь $idf(t) = \log\{(1+N)/(1+df(t))\}+1$ — инверсия частоты, с которой слово t встречается в текстах выборки; $df(t)$ — число текстов из выборки, где встречается слово t; $tf(t,d)$ — частотность слова t в тексте d; N — общее число текстов.

Как следует из формулы (1), значение меры tf-idf растет пропорционально частоте появления слова в документе, но это компенсируется количеством документов, содержащих это слово. Потому данная величина характеризует важность слова t в тексте d именно для текста d. Число ключевых слов документа было эмпирически подобрано и положено равным 20.

Следующим этапом после выделения признаков из корпуса текстов является непосредственно его кластеризация, и наиболее популярным методом при подобной параме-

тризации является графовый метод. В данной работе мы используем некоторую модификацию метода Dbscan [5]. Несмотря на то, что Dbscan — плотностной алгоритм пространственной кластеризации с присутствием шума, в то же время он представляет собой разновидность обхода в ширину графа выборки документов с некоторыми ограничениями.

В этом можно убедиться на примере общего описания алгоритма его работы. Сначала введём следующие определения. Пусть задана метрика $\rho(x, y)$ на элементах некоторого множества и константы ε и m. Тогда: область $E(x)$, в которой для любого y: $\rho(x, y) \le \varepsilon$, — ε-окрестность элемента x; корневым элементом называется объект, ε-окрестность которого содержит не менее m других элементов; элемент p непосредственно достижим из элемента q, если $p \in E(q)$ и q — корневой элемент; элемент q достижим из p, если имеется путь $p_1, \ldots, p_n$ с $p_1 = p$ и $p_n = q$, и каждый элемент p_{i+1} непосредственно достижим из p_i, $1 \le i < n$; все элементы, не достижимые из корневых элементов, считаются шумом.

Тогда в общем виде алгоритм Dbscan можно представить так:

1. Выбираем любой необработанный элемент p и отмечается как обработанный.

2. Находим все непосредственно достижимые элементы объекта p.

3. Определяем, является ли p корневым элементом.

• Если элемент p — корневой, то создаём новый кластер и находим среди других необработанных объектов все достижимые из p элементы, добавляя их в кластер.

• Если элемент p — некорневой, то отмечаем его как шум.

4. Если присутствуют необработанные элементы, то возвращаемся к пункту 1.

Как видно из последнего, алгоритм Dbscan может быть интерпретирован, как выделение компонент связности в графе текстовой выборки, где пара точек соединяются ребром в случае, если один из них непосредственно достижим из другого.

Алгоритм Dbscan применим для больших корпусов документов. Время выполнения в таком случае уменьшается с помощью использования структур для поиска достижимых элементов, таких как R-деревья [17], представляющих собой индексацию многомерной информации. Также, например, в работе [18] для подобных целей используется основанная на графе групп элементов выборки индексная структура данных.

В силу наличия требования в быстродействии алгоритма в поставленной задаче и в силу эффективности применения индексных структур для увеличения скорости выполнения кластеризации, мы используем аналогичный подход. А именно, после выделения признаков документов мы строим инвертированный индекс для словаря корпуса текстов, то есть структуру данных, в которой для каждого слова коллекции документов в соответствующем списке перечислены все документы в коллекции, в которых оно встретилось.

Исходя из результатов работы [14], Dbscan имеет значительное увеличение скорости выполнения, если в качестве аргумента ему передаётся матрица расстояний между элементами выборки. Имея в виду этот факт, мы далее определим объект связей документов в корпусе, то есть для каждого текста с помощью построенного ранее инвертированного индекса найдём непосредственно достижимые элементы выборки, если таковые имеются. Критерием непосредственной достижимости элемента p из элемента q в нашем случае будет являться наличие некоторого числа общих терм среди ключевых слов соответствующей пары текстов. В ходе эксперимента это число было эмпирически подобрано и положено равным 6.

Данный объект имеет смысл матрицы расстояний для векторизированной выборки. Так как здесь вместо вычислений значений метрики на паре векторов большой размерности, используются операции над множествами с небольшим числом элементов, то возникает предположение о более быстром создании подобного объекта связей. Таким образом мы получаем все корневые элементы вместе с элементами, непосредственно достижимыми из них. Алгоритм Dbscan не меняется, кроме действий, связанных с определением связанности пары элементов, кото-

рые отсутствуют по причине наличия ранее вычисленного объекта связей.

Отметим, что зачастую алгоритм Dbscan применяется для кластеризации выборки, параметризованной числовым набором признаков (как, например, в работах [5], [14], [18]). В нашем же случае Dbscan применяется к индексной структуре — объекту связей документов корпуса. Подобный подход имел место, например, в работе [18], однако в этом исследовании индексная структура строилась для векторизированных текстов, в нашем же случае она создаётся на основе наборов ключевых слов документов. Подобный подход к реализации поставленной цели исследования выбран из-за предположения о его быстродействии, сформированном на анализе результатов различных работ по данной задаче. Для проверки этого предположения мы проводим замер времени выполнения ключевых операций.

Также мы следим за числом элементов, отмеченных как шум. Малое количество текстов при итоговом отображении кластеров свидетельствует о том, что много информации из выборки потеряно. Точность соответствия новостей теме кластера оценивалась как по значению формального показателя, так и по непосредственному просмотру их содержимого.

Формально показатель точности кластеризации определяется с помощью силуэтного анализа [19], который чаще всего применяется, если истинные метки элементов выборки неизвестны.

Коэффициент силуэта рассчитывается для каждого кластера и зависит от двух показателей: a — среднего расстояния между документом и всеми другими элементами кластера, в котором он находится, b — среднего расстояния между документом и всеми элементами во всех других кластерах, которым этот документ не принадлежит. Коэффициент силуэта s для одного кластера задается следующим образом:

$$s = \frac{b - a}{\max(a, b)}. \qquad (2)$$

Здесь под расстоянием понимается евклидово расстояние:

$$dist(\mathbf{p}, \mathbf{q}) = \sqrt{(p_1 - q_1)^2 + \ldots + (p_n - q_n)^2},$$

где $\mathbf{p} = (p_1, \ldots, p_n)$, $\mathbf{q} = (q_1, \ldots, q_n)$ — некоторые n-мерные вектора. Таким образом, коэффициент силуэта показывает, насколько среднее расстояние до элементов своего кластера отличается от среднего расстояния до элементов других кластеров.

Для группы кластеров коэффициент силуэта рассчитывается как среднее значение от всех коэффициентов для каждого кластера. Более высокий показатель коэффициента силуэт относится к случаю, кода создаются плотно заполненные, четко разделённые кластеры. Он соответствует 1, а в случае, когда формируются кластеры сильно разрозненной структуры, он равен –1. Значения вблизи 0 указывают на то, что получившиеся кластеры пересекаются, то есть накладываются друг на друга.

Вектора текстов определяются на этапе получения ключевых слов текста с помощью меры tf-idf (1). Максимальный размер словаря корпуса документов составлял 15 000 слов. То есть для каждого текста выборки сначала вычислялся вектор в 15 000 координат, впоследствии из которых выбирались 20 с наибольшим значением, соответствующие наиболее важным ключевым словам документа. Далее эти вектора используются только для вычисления формального показателя качества кластеризации (2), то есть не нуждаются в последующем хранении и могут быть удалены из памяти ЭВМ.

РЕЗУЛЬТАТЫ ИССЛЕДОВАНИЯ И ИХ ОБСУЖДЕНИЕ

Временная сложность предложенного алгоритма в худшем случае составляет $O(n^2)$, в лучшем — $O(n)$. Отметим, что для классического алгоритма Dbscan временная сложность находится в границах от $O(n\log(n))$ до $O(n^2)$. Показатель скорости выполнения, близкий к наилучшему уровню временной сложности, достигается, например, на данных, где каждое слово из словаря выборки встречается в одинаковом числе текстов и каждый текст связан также с одинаковым числом других текстов выборки, и оба этих числа постоянны и не зависят от её размера. (см. рис. 1, рис. 2)

Рис. 1. Число операций
[Fig. 1. Number of operations]

Рис. 2. Время выполнения кластеризации
[Fig. 2. Clustering time]

Появились фото Ту-160М в небе над Казанью

Модернизированный в Казани «Белый лебедь» совершил свой первый полет
модернизирова, жуковск, отработа, 1500, лебед, глубок, оборудован, аэродром, высот, соверш, модернизац, бел, полет, длил, ту-160м, штатн, казан, экипаж, замечан, анр

Первый полет модернизированного Ту-160М продлился более получаса
модернизирова, ракетоносец-бомбардировщик, жуковск, ту-160, па, подня, строев, сист, баз, горбунов, неб, радиолокацион, оборудован, бомбардировщик, полет, продл, ту-160м, аэродром, анр, тупол

Модернизированный Ту-160М впервые поднялся в воздух
модернизирова, вперв, грац, строев, ракетоносец-бомбардировщик, подня, самолет, тупол, конструкторск, глубок, оборудован, лебед, стратегическ, изменя, полет, воздух, ту-160м, аэродром, sakhanews, ту-160

ПАО «Туполев» показал фото первого полета ракетоносца Ту-160М над Казанью
значительн, назначен, над, показа, фот, глубок, радиолокацион, радиоэлектрон, модернизац, противодейств, казан, полет, бортов, применен, ту-160м, эффективн, па, станц, комплекс, тупол

Появились фото Ту-160М в небе над Казанью
модернизирова, над, строев, фот, горбунов, неб, авиацион, радиолокацион, модернизац, появ, оборудован, казан, радиоэлектрон, полет, бортов, ту-160м, па, аэродром, комплекс, тупол

Обновленный бомбардировщик Ту-160М совершил первый полет
пишет, ракетоносец-бомбардировщик, обновлен, тупол, сист, оборон, радиолокацион, навигацион, оборудован, соверш, радиоэлектрон, бомбардировщик, полет, бортов, штатн, ту-160м, горбунов, аэродром, режим, стратегическ

«Россети» запустят подстанцию «Спутник» в Воронеже в 2020 году

«Россети» цифруют сетевое хозяйство
спутник, цифров, диспетчер, подстанц, маяк, затрат, воронежск, стилизова, электросетев, россет, опор, электроэнерг, маковск, потребител, ливинск, диспетчерск, технологическ, воронеж, электроснабжен, трансформац

Глава “Россетей” проконтролировал строительство подстанции «Спутник» в Воронеже
спутник, цифров, глав, подстанц, видеонаблюден, несанкционирова, перевест, россет, электросет, оснащ, посторон, корректив, проконтролирова, строительств, проникновен, воронеж, энергообъект, узл, 2030, трансформац

Новая подстанция «Россетей» появится в Воронеже
спутник, трансформац, цифров, подстанц, 110, электросетев, россет, нов, воплощен, появ, леж, контекст, постройк, концепц, производ, воронеж, имеющ, 2030, запуст, консолидац

Строительство подстанции «Спутник», запуск цифровых объектов и другие итоги визита Павла Ливинского в Воронеж
спутник, запуск, цифров, гусев, подстанц, павел, визит, опор, россет, павл, электроэнерг, электросетев, стилизова, ливинск, костром, объект, костромск, эксплуатац, воронеж, гус

«Россети» запустят подстанцию «Спутник» в Воронеже в 2020 году
спутник, подстанц, видеоконференц, видеонаблюден, несанкционирова, объект, костромск, россет, энергообъект, высокотехнологичн, акцентирова, охраня, ливинск, холдинг, проникновен, воронеж, белгородск, 2030, запуст, трансформац

Рис. 3. Пример отображения содержимого кластеров
[Fig. 3. An example of the clusters content]

При тестировании алгоритма были получены следующие результаты (см. табл. 1, табл. 2, рис. 3). При выполнении все операции алгоритма выполнялись последовательно, то есть действия не распараллеливались там, где это было возможно. Также после получения меток кластеров для документов выборки, тексты кластеров, содержащих менее четырёх элементов, отмечались как шум. Это учитывалось при расчёте числа текстов, не вошедших в кластеры, и при расчёте коэффициента силуэт (2).

Отметим, что суммарное время получения объекта связей и выполнения непосред-

Таблица 1. Время работы алгоритма
[Table 1. Running time of the algorithm]

Размер выборки	Время токенизации и стемминга (в с.)	Время расчёта меры tf-idf (в с.)	Размер вектора признаков	Время расчета объекта связей (в с.)	Время кластеризации (в с.)
1000	13.88	0.15	10464	0.04	1.06E-04
5000	62.83	0.60	15000	0.68	0.0011
10000	129.00	1.31	15000	2.54	0.0033
20000	260.51	2.32	15000	8.68	0.0070
30000	395.98	3.50	15000	18.85	0.0117
40000	519.69	4.51	15000	33.11	0.0172
44000	568.08	5.10	15000	41.09	0.0200

Таблица 2. Результаты алгоритма
[Table 2. Algorithm results]

Размер выборки	Число кластеров	Число кластеров после фильтрации	Число отброшенных текстов	Силуэт
1000	17	9	949	0.25
5000	129	90	4385	0.27
10000	265	186	8717	0.25
20000	515	373	17284	0.18
30000	776	566	25672	0.14
40000	988	723	34602	0.13
44000	1078	768	38285	0.14

ственно кластеризации (см. табл. 1) сравнимо, например, с результатами в работе [18], где тестирование предложенного в статье метода проводилось для 10000 векторов размерности от 5 до 65. Также данные показатели скорости значительно выше, чем скорость любого алгоритма, рассмотренного в работе [14], в которой для тестирования была задействована аналогичная текстовая выборка и метод проверки эффективности алгоритма, но применялась векторное представление элементов корпуса.

Исходя из результатов табл. 2, отметим, что данный алгоритм по числу оставленных для итогового отображения кластеров уступает, например, результатам, продемонстрированным в работе [16]. В ней для тестирования была выбрана выборка, состоящая из 1555 текстов новостей, и алгоритм, представленный в статье, имеет показатель полноты в среднем равный 700 текстам, сгруппированным по кластерам. Однако в тоже время как коэффициент силуэт, так и показатель полноты в данном исследовании сравнимы со значениями этих величин, полученных в работе [14].

В результате проведённого тестирования, можно отметить высокую скорость выполнения алгоритма кластеризации. Заметим, что этому значительно способствует применение индексной структуры, что было отмечено, например, в работах [16], [18]. Однако мы параметризируем документы выборки их ключевыми словами, а числовым набором. Наибольшую часть времени при этом занимает извлечение признаков, а именно морфологическая обработка текстов. Также алгоритм группирует в кластеры сравнительно небольшое число документов из корпуса, но этот показатель здесь не хуже, чем, например, в работе [14]. В проведённом исследовании наблюдается сравнительно невысокий показатель коэффициента силуэт, однако при просмотре содержимого ошибок метода не наблюдается: в каждом кластере находятся документы близкой тематики (рис. 3).

ЗАКЛЮЧЕНИЕ

В данной работе был предложен метод кластеризации текстовой выборки фиксированной длины, позволяющий ускорить выполнение задачи. Достижение поставленной цели осуществляется за счёт параметризации корпуса документов их ключевыми словами и введения индексной структуры данных, имеющей смысл объекта связей элементов корпуса. В результате проведённого исследования была проверена эффективность метода по показателю скорости выполнения. В реализации все этапы алгоритма выполнялись последовательно. Однако, например, получение стеммингованных слов текста или создание индекса для отдельного документа в объекте связей происходит независимо от других элементов выборки. Возможно, скорость реализации можно увеличить ещё больше, распараллелив эти процессы. На показатели качества кластеризации может повлиять использование синонимических групп словаря выборки, а также содержательных выражений и именованных сущностей текстов при параметризации документов. Возможно, подобное определение позволит получать более чёткое разбиение на кластеры корпуса текстовых документов, однако следует учитывать, что это потребует дополнительной лингвистической обработки выборки.

КОНФЛИКТ ИНТЕРЕСОВ

Авторы декларируют отсутствие явных и потенциальных конфликтов интересов, связанных с публикацией настоящей статьи.

СПИСОК ЛИТЕРАТУРЫ

1. *Пархоменко, П. А.* Обзор и экспериментальное сравнение методов кластеризации текстов / П. А. Пархоменко, А. А. Григорьев, Н. А. Астраханцев // Труды ИСП РАН. – 2017. – Т. 29, № 2. – С. 161–200. DOI: 10.15514/ispras-2017-29(2)-6

2. *Aggarwal Charu C.* Mining text data. / Aggarwal Charu C., Zhai Cheng Xiang. – New York:Springer, 2012. – 522 p.

3. *Богатырев, М. Ю.* Анализ текстов естественного языка с применением многомерной кластеризации / М. Ю. Богатырев , Н. Л. Коржук // Известия ТулГУ. Технические науки. – 2019. – № 9. – С. 142–150.

4. *Bejar, J.* K-means vs Mini Batch K-means: a comparison. – Режим доступа: http://hdl.handle.net/2117/23414 – (дата обращения 10.06.2020).

5. *Ester, M.* Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise / M. Ester, H. P. Kriegel, J. Sander, A. XiaoweiXu // In Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining. AAAI Press. – 1996. – P. 264–323.

6. *Shi, M.* WE-LDA: A Word Embeddings Augmented LDA Model for Web Services Clustering / M. Shi, J. Liu, D. Zhou, M. Tang, B. Cao // IEEE International Conference on Web Services (ICWS). – 2017. – P. 9–16. DOI: https://doi.org/10.1109/icws.2017.9

7. *Eissa, M.* Alshari Improvement of Sentiment Analysis Based on Clustering of Word-2Vec Features / Eissa M. Alshari, Azreen Azman, Shyamala Doraisamy, Norwati Mustapha and Mustafa Alkeshr // 28th International Workshop on Database and Expert Systems Applications (DEXA). – 2017. – P. 123–126. DOI: https://doi.org/10.1109/dexa.2017.41

8. *Pennington, J.* Glove: Global Vectors for Word Representation / Jeffrey Pennington, Richard Socher, Christopher D. Manning // Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP) – 2014. – P. 1532–1543. DOI: https://doi.org/10.3115/v1/d14-1162

9. *Stankevičius L., Lukoševičius M.* Testing pre-trained Transformer models for Lithuanian news clustering. – Режим доступа: https://arxiv.org/pdf/2004.03461.pdf (дата обращения 10.06.2020).

10. *Bird, S.* NLTK: the natural language toolkit / Bird S. // In Proceedings of the COLING/ACL on Interactive presentation sessions. Association for Computational Linguistics. – 2006. – P. 69–72. DOI: https://doi.org/10.3115/1225403.1225421

11. *Солошенко, А. Н.* Методы тематической кластеризации применительно к анализу новостных статей / А. Н. Солошенко, Ю. А. Ор-

лова, А. В. Заболеева-Зотова // Открытые семантические технологии проектирования интеллектуальных систем. – 2015. – № 5. – С. 555–560.

12. *Лапшин, С. В.* Кластеризация текстов с использованием семантико-синтаксических связей слов / С. В. Лапшин, И. С. Лебедев, А. И. Спивак // Научно-технический вестник информационных технологий, механики и оптики. – 2019. – № 6. – С. 1058–1063. DOI: 10.17586/2226-1494-2019-19-6-1058-1063

13. *Захаров, В. Н.* Метод кластеризации новостных сообщений средств массовой информации на основе их концептуального анализа / В. Н. Захаров, Р. Р. Мусабаев, А. М. Красовицкий, Я. Д. Козловская, А. А. Хорошилов, А. А. Хорошилов // Системы и средства информ. – 2019. – Т. 29, № 3. – С. 52–65. DOI: https://doi.org/10.14357/08696527190305

14. *Головастова, Э. А.* Задача эффективной кластеризации текстовой выборки в зависимости от различной параметризации этой выборки / Э. А. Головастова, Д. Н. Красотин // Информационные технологии и вычислительные системы. – 2019. – № 4. – С. 60–69. DOI: 10.14357/20718632190406

15. *Отраднов, К. К.* Модель кластеризации слабоструктурированных текстовых данных / К. К. Отраднов, Д. О. Жуков, О. А. Новикова // Современные информационные технологии и ИТ-образование. – 2017. – Т. 13, № 3. – С. 100–115. DOI: 10.25559/SITITO.2017.3.439

16. *Девяткин, Д. А.* Метод тематической кластеризации масштабных коллекций научно-технических документов / Д. А. Девяткин, Р. Е. Суворов, И. В. Соченков // Информационные технологии и вычислительные системы. – 2013. – № 1. – С. 33–42.

17. *Guttman, A.* R-trees: A dynamic index structure for spatial searching / A. Guttman // In: Proc. of 13th Int. Conf. on Mang. of Data ACM SIGMOD. – 1984. – V. 2. – P. 47–57. DOI: https://doi.org/10.1007/springerreference_62807

18. *Mahesh, K. Kumar.* A fast DBSCAN clustering algorithm by accelerating neighbor searching using Groups method / K. Mahesh Kumar, A. Rama Mohan Reddy // Pattern Recognit. – 2016. – № 58. – P. 39–48. DOI: https://doi.org/10.1016/j.patcog.2016.03.008

19. *Peter, J.* Rousseeuw Silhouettes: a Graphical Aid to the Interpretation and Validation of Cluster Analysis / Peter J. Rousseeuw // Computational and Applied Mathematics. – 1987. – V. 20. – P. 53–65. DOI: https://doi.org/10.1016/0377-0427(87)90125-7

Головастова Элеонора Александровна — аспирант кафедры теории вероятностей механико-математического факультета Московского государственного университета им. М. В. Ломоносова.
E-mail: golovastova.elina@yandex.ru
ORCID iD: https://orcid.org/0000-0003-2802-0882

Красотин Дмитрий Николаевич — ведущий инженер в ЗАО «Московский научно-исследовательский телевизионный институт».
E-mail: dima88_kr@mail.ru
ORCID iD: https://orcid.org/0000-0001-6258-5030

DOI: https://doi.org/10.17308/sait.2020.3/3045 ISSN 1995-5499
Received 07.09.2020
Accepted 30.09.2020

CLUSTERING TEXT SAMPLES PARAMETERISED BY THE KEYWORDS OF THEIR ELEMENTS

© 2020 E. A. Golovastova✉[1], D. N. Krasotin[2]

[1]Lomonosov Moscow State University
GSP-1, Leninskie Gory, 119991 Moscow, Russian Federation
[2]CJSC "MNITI"
7A, str.1, Golyanovskaya Street, 105094 Moscow, Russian Federation

Annotation. This paper describes the solution to the problem of automated clustering of large text samples of a fixed length. Automatic grouping of texts of similar meaning is one of the most important tasks of data analysis since it has a wide scope of applications. The study focuses on the execution speed of the algorithm. That is why we consider a method to present a text sample by using its keywords as a set of document characteristics. Keywords are defined by the pre-computed values of the statistical TF-IDF measures. The next step involves text sample clustering. The study uses a modification of the Dbscan method, which is a density-based spatial clustering algorithm with the presence of noise. However, here it is interpreted as a form of breadth-first traversal with some restrictions of the document selection graph. Dbscan takes an index structure as an argument. This index structure is an object of links between documents in the corpus. Such an approach to the solution of the problem was chosen due to its presumed speed. To test this assumption, we measured the execution time of main operations, whose values are given to illustrate the test result of the proposed clustering method.
Keywords: clustering, text samples, TF-IDF measures, keywords, index data structure, Dbscan algorithm, execution speed.

CONFLICT OF INTEREST

The authors declare the absence of obvious and potential conflicts of interest related to the publication of this article.

REFERENCES

1. *Parhomenko P. A., Grigorev A. A. & Astrakhantsev N. A.* A survey and an experimental comparison of methods for text clustering: application to scientific articles. Proceedings of the Institute for System Programming of the RAS. 2017. 29(2). P. 161–200. Available at: http://dx.doi.org/10.15514/ispras-2017-29(2)-6.

2. *Aggarwal Charu C., Zhai Cheng Xiang.* Mining text data. New York, Springer. 2012

3. *Bogatyrev, M. Yu., Korzhuk, N. L.* Application of multidimensional formal contexts in natural language text analysis. Izvestiya Tul'skogo gosudarstvennogo universiteta. Tekhnicheskie nauki. 2019. 9. P. 42–150.

4. *Bejar J.* K-means vs Mini Batch K-means: a comparison. 2013. Available at: http://hdl.handle.net/2117/23414 (accessed 10.06.2020).

5. *Ester M., Kriegel H. P., Sander J., Xiaowei Xu A.* Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise In Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining. 1996. P. 264–323.

6. *Shi M. et al.* WE-LDA: A Word Embeddings Augmented LDA Model for Web Services Clustering. 2017 IEEE International Conference on Web Services (ICWS). 2017. P. 9–16. Available at: http://dx.doi.org/10.1109/icws.2017.9.

7. *Alshari E. M., Azman A., Doraisamy S., Mustapha N., & Alkeshr M.* Improvement of Sentiment Analysis Based on Clustering of Word2Vec Features. 2017 28th International Workshop on Database and Expert Systems Ap-

✉ Golovastova Eleonora A.
e-mail: golovastova.elina@yandex.ru

plications (DEXA). 2017. P. 123–126. Available at: doi:10.1109/dexa.2017.41

8. *Pennington J., Socher R. & Manning C.* Glove: Global Vectors for Word Representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2014. P. 1532–1543. Available at: doi:10.3115/v1/d14-1162

9. *Stankevičius L., Lukoševičius M.* Testing pre-trained Transformer models for Lithuanian news clustering. Available at: https://arxiv.org/pdf/2004.03461.pdf [Accessed 10.06.2020].

10. *Bird S.* NLTK.: the natural language toolkit. In proceedings of the COLING/ACL on Interactive presentation sessions. 2006. P. 69–72. Available at: http://dx.doi.org/10.3115/1225403.1225421.

11. *Soloshenko A. N., Orlova Yu. A., Zaboleeva-Zotova A. V.* Thematic clustering methods applied to news articles analysis. OSTIS. 2015. 5. P. 555–560.

12. *Lapshin S. V., Lebedev I. S., Spivak A. I.* Text clustering powered by semantico-syntactic features Scientific and Technical Journal of Information Technologies, Mechanics and Optics. 2019. 6. P. 1058–1063. Available at: 10.17586/2226-1494-2019-19-6-1058-1063

13. *Zakharov V. N., Mussabayev R. R., Krassovitsky A. M., Kozlovskaya Ya.D., Khoroshilov Aleksandr A., Khoroshilov Alexey A.* Clustering method of news media reports based on conceptual analysis Systems and Means of Informatics. 2019. 29(3). P. 52–65. Available at: https://doi.org/10.14357/08696527190305

14. *Golovastova E. A., Krasotin D. N.* Effective clustering of a text sample depending on the different parameterization of this sample Informacionnye tekhnologii I I vichslitel'nye sistemy. 2019. 4. P. 60–69. Available at: 10.14357/20718632190406

15. *Otradnov K. K., Zhukov D. O., Novikova O. A.* Clustering model of low-structured text data Modern Information Technologies and IT-education. 2017. 13(3). P. 100–115. Available at: 10.25559/SITITO.2017.3.439

16. *Deviatkin D. A., Suvorov R. E., Sochenkov I. V.* A method for topic clustering for large science publication collections Informacionnye tekhnologii I I vichslitel'nye sistemy. 2013. 1. P. 33–42.

17. *Guttman A.* R-Trees – A Dynamic Index Structure for Spatial Searching. In: Proc. of 13th Int. Conf. on Mang. of Data ACM SIGMOD. 1984. 2. P. 47–57. Available at: http://dx.doi.org/10.1007/springerreference_62807.

18. *Mahesh Kumar K. & Rama Mohan Reddy A.* A fast DBSCAN clustering algorithm by accelerating neighbor searching using Groups method. Pattern Recognition. 2016. 58. P. 39–48. Available at: http://dx.doi.org/10.1016/j.patcog.2016.03.008.

19. *Rousseeuw P. J.* Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics. 1987. 20. P. 53–65. Available at:10.1016/0377-0427(87)90125-7

Golovastova Eleonora A. — PhD student, Lomonosov Moscow State University.
E-mail: golovastova.elina@yandex.ru
ORCID iD: https://orcid.org/0000-0003-2802-0882

Krasotin Dmitrii N. — leading engineer, ZAO "Moscow Research Institute of Television".
E-mail: dima88_kr@mail.ru
ORCID iD: https://orcid.org/0000-0001-6258-5030