УДК 004.4, 681.3

ПОДХОД К ОБРАБОТКЕ ОБРАТНОЙ СВЯЗИ ПОЛЬЗОВАТЕЛЯ ДЛЯ ПОВЫШЕНИЯ КАЧЕСТВА РЕЗУЛЬТАТОВ РАБОТЫ АЛГОРИТМА КЛАСТЕРИЗАЦИИ

© 2020 П.В. Дударин, В.Г. Тронин, Н.Г. Ярушкина

Ульяновский государственный технический университет, Ульяновск, Россия

Статья поступила в редакцию 05.10.2020

Набор данных может иметь более одного «правильного» варианта разбиения на группы в результате работы метода кластеризации в зависимости от целей исследователя. В случае неудовлетворенности результатами исследователю приходится вносить изменения в пространство признаков для корректировки результата. Зачастую эта связь не прозрачна, что приводит к большому числу итераций. В данной работе представлен подход на базе нейронных сетей, позволяющий итеративно учитывать обратную связь без корректировки пространства признаков.

Ключевые слова: кластеризация, интерактивная кластеризация, кластеризация с ограничениями, нейронные сети, глубинное обучение, глубинная кластеризация, глубинное представление, обратная связь.

DOI: 10.37313/1990-5378-2020-22-5-94-105

ВВЕДЕНИЕ

Методы кластеризации традиционно относят к методам, обучающимся без учителя. Такое обучение возможно благодаря информации, содержащейся в самих данных, которую и призваны выявить методы кластеризации [14]. Тем не менее, на практике исследователь редко не обладает никакими знаниями об исследуемом наборе данных, будь то экономические данные [30], данные, собранные с датчиков, приборов [35] или каким-либо иным образом компьютерной программной [32]. Практически при каждом решении реальной задачи участие исследователя необходимо либо для построения корректного разбиения на группы, либо принятия решения о структуре иерархии, либо способно существенно повысить качество результата за счет знаний, не включенных в пространство признаков обрабатываемых данных. Особенно это актуально при обработке текстовой информации [31]. Тексты, являясь многомерными объектами, представляют особенную сложность для алгоритмов кластеризации [11]. Без участия эксперта, без выявления его скрытых интенций невозможно заранее определить, какое именно разбиение ожидается в результате работы алгоритма [33]. Помимо очевидной группировки по тематике, тексты могут быть сгруппированы на основании того, от чьего лица ведется повествование, по целевой аудитории текста, по правовому статусу текста или комбинации различных признаков. В этой области абсолютно верно высказывание: «не бывает правильной кластеризации, бывает полезная» [2]. Таким образом, включение эксперта в процесс кластеризации не является недостатком метода, а, наоборот, является желательным [34]. При этом важно, чтобы участие эксперта было органичной частью алгоритма кластеризации, не требовало бы понимания внутренних деталей работы алгоритма, и, при этом, причинно-следственная связь между действиями эксперта и результатами работы алгоритма была бы явной.

Недавние исследования отмечают значительный рост в последние годы числа работ, посвященных алгоритмам кластеризации, предполагающим участие исследователя. Например, в работе [2] приводится статистика по годам, представленная на Рис. 1.

В данной работе предлагается подход, позволяющий включить использование обратной связи от эксперта по результатам кластеризации в широкое семейство современных алгоритмов кластеризации, основанных на использовании нейронных сетей, в частности представлена реализация на основе алгоритма DEC (Unsupervised Deep Embedding for Clustering Analysis) [25].

Дальнейшее содержимое статьи организовано следующим образом: в первом разделе приводится обзор и анализ смежных работ, во втором разделе сформулирована задача настоящего исследования, в третьем разделе представлен предлагаемых подход, в четвертом – приведены эксперименты подтверждающие работоспособность и эффективность предлагаемого подхода, и в заключении подводятся итоги и описываются возможные направления развития.

Дударин Павел Владимирович, аспирант кафедры "Информационные системы". E-mail: p.dudarin@ulstu.ru
Тронин Вадим Георгиевич, кандидат технических наук, доцент кафедры "Информационные системы". E-mail: v.tronin@ulstu.ru
Ярушкина Надежда Глебовна, доктор технических наук, профессор, ректор, заведующая кафедрой «Информационные системы». E-mail: jng@ulstu.ru

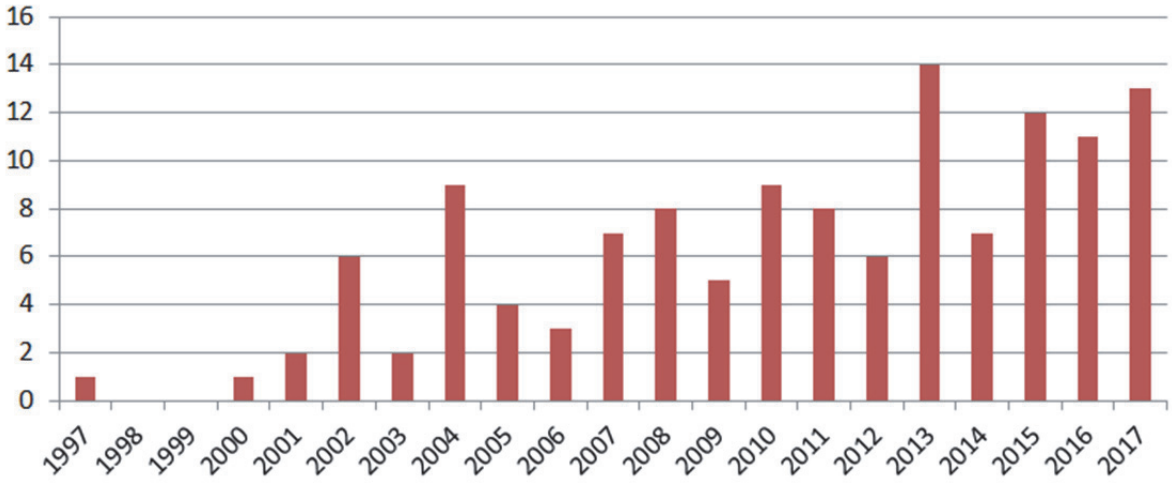

Рис. 1. Результаты прогнозирования с помощью нейронной сети

ОБЗОР И АНАЛИЗ СМЕЖНЫХ РАБОТ

Для определения круга смежных работ следует уточнить определения и прояснить классификацию в области методов кластеризации. В современной научной литературе сложилась практика обозначения методов кластеризации, в которых используется та или иная дополнительная информация, не включенная в набор данных, методами кластеризации с частичным привлечением учителя (semi-supervised), кластеризацией с ограничениями (constrained clustering) [6]. При этом в подавляющем большинстве таких методов информация дана a priori и подается на вход алгоритму кластеризации совместно с набором данных в виде частично промаркированных объектов [5], заданных ограничений на пары объектов [9], ограничения на структуру иерархии кластеров, перенос знания в виде предобученной нейронной сети (transfer learning) [24], например, на задаче классификации в схожей предметной области и т.д. При этом и ограничения на объекты и метки могут быть заданы не жестко (soft labels) [20].

Однако, существуют методы предполагающие получение дополнительной информации непосредственно в процессе кластеризации. Их подробный обзор произведен в работе [2]. Такие методы называются методами интерактивной кластеризации. Одним из первых таких методов стал нечеткий метод [22]. От характера взаимодействия и получаемой информации они подразделяются на: активную кластеризацию как пример активного обучения [8, 12]; кластеризация с подкреплением, получаемой в виде обратной связи от среды в которой происходит кластеризация [3]; интерактивная кластеризация с обратной связью (interactive clustering under feedback, mixed-initiative clustering), подразумевающая получение обратной связи от пользователя в виде оценки результатов или указаний по корректировке алгоритма. Последние методы позволяют выявить скрытые интенции пользователя и получить по настоящему полезную кластеризацию, т.к. хорошо соответствуют тезису: «пользователь узнает правильный результат, когда увидит его».

Исследователи отмечают, что к интерактивным методам зачастую ошибочно относят и методы кластеризации с интерактивными операциями: методы интерактивной визуализации результатов кластеризации, методы подбора выбора алгоритмов кластеризации и т.п. [2].

Для полноты картины следует упомянуть методы вспомогательной кластеризации (assisting clustering) [7], в которых ведущая роль отдана исследователю; именно он определяет количество кластеров и их характеристики, а алгоритм предлагает варианты их наполнения и корректировки структуры. Однако этим методы на данный момент не получили значительного распространения.

Методы интерактивной кластеризации с обратной связью можно разделить на два множества по тому, на что направлена обратная связь от исследователя. В первом более многочисленном семействе методов исследователь интерактивно и итеративно может влиять на параметры алгоритма кластеризации, метрику схожести (близости), модифицировать пространство признаков [16]. Во втором множестве методов исследователь взаимодействует непосредственно с результатами кластеризации, указывая, какие кластеры необходимо объединить или разъединить, какие элементы добавить или исключить из кластера, каким образом образовать новый кластер или куда отнести элементы, выпадающие из кластеризации [2, 4]. Подход, предлагаемый в данной работе, относится именно ко второму множеству, что позволяет исследователю не погружаться в детали реализации алгоритма и использовать новые появляющиеся методы, не меняя характер своей работы.

Первым этапом интерактивной кластеризации, очевидно, является обычная кластеризация без учителя. Таким образом, все методы интерактивной кластеризации базируются на методах без учителя, добавляя в них механизмы работы с обратной связью. Согласно исследованию [2] результаты, которого представлены на Рис. 1. Большинство методов интерактивной кластеризации основываются на классических методах кластеризации, таких как: k-means, c-means, вариациях иерархической кластеризации и кластеризации графов. Малое число методов использует нейронные сети, при этом используются сети SOM (Kohonen self-organized maps).

Классические методы кластеризации успешно применяются на практике и показывают высокие результаты, тем не менее, существует большое количество современных методов, демонстрирующих намного лучшие результаты (state-of-the-art results). Большинство этих методов основываются на использовании сетей с глубинным обучением (deep neural network) [18, 23, 26, 27]. Такое превосходство объясняется способностью сетей обучаться на смежных предметных областях или схожих задачах (transfer learning, learning to cluster) и строить сложные нелинейные преобразования для получения пространства признаков (representation learning, embedding learning) одновременно содержащего максимум информации и «удобного» для алгоритма кластеризации (например, сильное понижение размерности входных данных) [29]. Но самым главным вкладом использования нейронных сетей в методы кластеризации является возможность построения непрерывной кластеризации (end-to-end clustering), в которой отсутствует явное разделение алгоритма на две фазы: построение пространства признаков и разбиения на группы [13, 19]. При таком подходе обучение сети подходящему представлению данных происходит одновременно с итерациями разбиения множества на кластеры или построения иерархии из них. В ряде методов авторы показывают возможность дальнейшего переноса полученных знаний сети на смежные задачи, например, использование сети, обученной для кластеризации одного вида изображений на другой вид изображений. Существуют работы, посвященные кластеризации с частично размеченным набором данных на базе нейронных сетей [15, 24], но они используют эту маркировку в процессе первоначального обучения сети, а не получают в виде обратной связи, т.е. не подстраиваются в процессе обработки результатов под нужды исследователя.

ПОСТАНОВКА ЗАДАЧИ ИССЛЕДОВАНИЯ

В своем исследовании [1] авторы предлагают обобщенную схему построения современных методов кластеризации с использованием нейронных сетей, в которую укладывается подавляющее большинство методов (Рис. 2.) [10, 25, 28]. Из этой схемы видно, что обработка обратной связи на уровне единой целевой функции (функции потерь, штрафной функции, loss function) позволит одновременно строить представление объектов в соответствии с интенцией исследователя, влияя на веса нейронной сети и корректировать ошибки кластеризации, влияя на результат следующей итерации кластеризации (например, смещая центры кластеров).

Задачей данного исследования является построение метода интерактивной кластеризации с обратной связью на базе современных методов кластеризации, укладывающихся в выше обозначенную обобщенную схему. В качестве основы могут быть использованы методы кластеризации на базе нейронной сети с единой целевой функцией, основанной на метрике Кульбака-Лейблера (Kullback-Leibler). В частности, будет представлена реализация на базе метода кластеризации DEC (Unsupervised Deep Embedding for Clustering Analysis) [25]. Встраивание обработку обратной связи в этом случае возможно благодаря тому, что данная целевая функция, по сути, управляет силой притяжения между элементами и центрами кластеров. Таким образом, оказывая точечное воздействие на эту силу, можно предсказуемо управлять результатом кластеризации.

ПОДХОД К ОБРАБОТКЕ ОБРАТНОЙ СВЯЗИ

Придерживаясь идеи, что для исследователя наиболее простой и точной обратной связью будет критика полученных результатов кластеризации, в данном подходе предполагается обратная связь двух видов: «элемент X_i должен принадлежать кластеру C_j» и «элементу X_i не следует находиться в кластере X_j». Одновременно может быть получено произвольное коли-

Рис. 2. Обобщенная схема построения методов кластеризации на базе нейронных сетей

чество таких ограничений, в частности, легко задается ограничение «поменять местами элементы X_i и X_j» комбинацией двух ограничений первого вида.

Как уже было указано выше, в качестве базового был выбран метод кластеризации DEC (Unsupervised Deep Embedding for Clustering Analysis), при этом аналогичный подход может быть применен ко множеству других алгоритмов, например, к DEPICT [10]. Входной набор данных $X=\{x_i \mid i \in [0, N)\}$, N – кол-во элементов в наборе. Это множество с помощью энкодера, являющегося частью заранее обученного автоэнкодера, отображается в пространство меньшей размерности: $f_\theta: X \to Z$, где θ – параметры нейронной сети, Z – скрытое пространство признаков. Пространство признаков называется в данном случае скрытым, т.к. его построение происходит в процессе обучения нейронной сети и затем оно формируется неявным образом в процессе обучения автоэнкодера и решения задачи кластеризации. В данной работе используется энкодер следующей структуры: d-50-50-20-k, где d-размерность входного набора данных, k – число кластеров. Результатом работы алгоритма является набор центров кластеров в пространстве Z: $\{\mu_j \in Z \mid j \in [0, k)\}$, где k – заданное число кластеров. Инициализация центров кластеров происходит при помощи алгоритма k-means, который применяется к представлению векторов, полученному в результате обучения автоэнкодера.

Процессы определения оптимального расположения центров кластеров и построения пространства признаков происходят одновременно за счет определения общей функции потерь. Для этого в качестве меры расстояния между элементом и центром кластера используется метрика, основанная на распределении Стьюдента с одной степенью свободы.

$$q_{ij} = \frac{(1 + ||z_i - \mu_j||^2)^{-1}}{\sum_{l=0}^{k}(1 + ||z_i - \mu_l||^2)^{-1}}).$$

Целевая функция (функция потерь, loss function) или штрафная функция строится как метрика Кульбака-Лейблера (Kullback-Leibler divergence) между фактическим и целевым распределением.

$$L = KL(P||Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}}.$$

В качестве целевого распределение используется следующее распределение:

$$p_{ij} = \frac{{q_{ij}}^2/f_j}{\sum_{l=0}^{k} {q_{il}}^2/f_l}, \qquad f_j = \sum_j q_{ij}.$$

Это распределение обладает следующим свойствами: усиливает вклад от элементов с большой долей принадлежности кластеру и нормализует влияние больших кластеров, не позволяя им чрезмерно притягивать к себе удаленные элементы за счет своего размер (аналог гравитации).

Нетрудно заметить, что целевая функция направлена на то, чтобы q_{ij} было больше p_{ij}. Если посмотреть на частные производные для обновления весов нейронной сети и векторов центров кластеров.

$$\frac{\partial L}{\partial z_i} = 2 \sum_j \left(1 + \left||z_i - \mu_j\right||^2\right)^{-1} * (p_{ij} - q_{ij}) * (z_i - \mu_j),$$

$$\frac{\partial L}{\partial \mu_j} = -2 \sum_i \left(1 + \left||z_i - \mu_j\right||^2\right)^{-1} * (p_{ij} - q_{ij}) * (z_i - \mu_j),$$

то можно понять, что в случае отрицательной разницы (pij-qij) будет происходить «выталкивание» элемента из кластера, несмотря на отсутствие штрафа со стороны целевой функции.

Для учета обратной связи предлагается использовать следующие формулы градиентов:

$$\frac{\partial L}{\partial z_i} = 2 \sum_j \left(1 + \left||z_i - \mu_j\right||^2\right)^{-1} * |p_{ij} - q_{ij}| * t_{ij} * (z_i - \mu_j),$$

$$\frac{\partial L}{\partial \mu_j} = -2 \sum_i \left(1 + \left||z_i - \mu_j\right||^2\right)^{-1} * |p_{ij} - q_{ij}| * t_{ij} * (z_i - \mu_j),$$

где $T = \{t_{ij}\}$ – матрица обратной связи (подсказок пользователя, tips), в которой:

$$t_{ij} = \begin{cases} > 0, & \text{для включения элемента } i \text{ в кластер } j \\ < 0, & \text{для исключения элемента } i \text{ из кластера } j \\ 0, & \text{иначе.} \end{cases}$$

Абсолютное значение t_{ij} определяет скорость (силу притяжения), с которой элементы и центры кластера будут стремиться друг к другу или отталкиваться друг от друга. Также на эту скорость влияет выставленный уровень обучения (learning rate) у нейронной сети. В данной работе в экспериментах использовалось значение 1000 для обоих видов обратной связи. При этом эксперименты показали, что для случая выталкивания элемента из кластера имеет смысл использовать большие абсолютные величины, чем при притяжении.

ОПИСАНИЕ ЭКСПЕРИМЕНТОВ И АНАЛИЗ РЕЗУЛЬТАТОВ

В данном разделе для демонстрации работоспособности предложенного подхода представлены два вида экспериментов. Вначале показывается работа на сгенерированном наборе данных из простых векторов, процесс кластеризации которых является тривиальным. При этом в силу вариативности исследовательского намерения, любой алгоритм кластеризации может расположить их неправильно. В экспериментах показывается, каким образом исследователь может уточнить свое намерение, и кластеризация подстраивается под него. Во втором раз-

деле эксперимент проводится на каноническом наборе данных – «Ирисы Фишера», для демонстрации того, как исследователь может добавить информацию, не содержащуюся в данных, и тем самым улучшить результаты кластеризации, и на наборе данных новостного агентства Reuters для сравнения эффективности предлагаемого подхода с аналогичным методом.

Демонстрация работы на синтетическом наборе данных. Для демонстрации работы был сгенерирован набор данных из 400 элементов, следующим образом:

1. За основу взяты 4 вектора {(1,0,0,0); (0,1,0,0); (0,0,1,0); (0,0,0,1)}.
2. Для каждого из 4-х векторов сгенерированы 125 векторов добавлением к каждой компоненте случайной величины из равномерного распределения U[0, 1/10]. Добавленный случайный шум совсем небольшой, т.к. в данном эксперименте задачей ставится показать влияние обратной связи, а не качество работы алгоритма самого по себе.
3. Векторы в выборке расположены последовательно четверками, таким образом, первые 4 вектора содержат по 1 представителю от каждого базового класса. В результатах экспериментов детально будут показаны только первые 12 векторов для краткости и ясности картины результата.

Для указанного набора данных был запущен алгоритм кластеризации с разбиением множества на 2 кластера. Результаты работы алгоритма кластеризации представлены в Таблице 1, при этом для автоэнкодера достигнутое значение функции потерь на проверочной выборке равно 0.000341. На Рисунке 3 показано распределение первых 12 векторов по кластерам. Далее будут показаны три эксперимента для различных видов обратной связи: указание о необходимости включения вектора X_0 в кластер C_0; указание о необходимости исключения вектора X_1 из кластера C_1; комплексная обратная связь по замене векторов X_2 и X_3 местами в кластерах C_0 и C_1 соответственно. Все эксперименты выполняются как первая итерация после первоначальной кластеризации, а не последовательно, исключительно для более удобного сравнения. Последовательное применение, очевидно, возможно, без каких либо ограничений или особенностей в работе алгоритма.

В первом эксперименте предположим, что исследователь обладает информацией, о том, что вектор X_1 семантически ближе к векторам X_2 и X_3, чем к вектору X_0. Поэтому для алгоритма кластеризации формируется обратная связь в виде

Таблица 1. Список первых 4 векторов набора данных

X	Координаты				Кластер
$\mathbf{X_0}$	**1.0191519**	0.06221088	0.04377278	0.07853585	C_0
$\mathbf{X_1}$	0.07799758	**1.0272592**	0.02764643	0.08018722	C_0
$\mathbf{X_2}$	0.09581394	0.08759326	**1.0357817**	0.05009951	C_1
$\mathbf{X_3}$	0.06834629	0.07127021	0.03702508	**1.0561196**	C_1

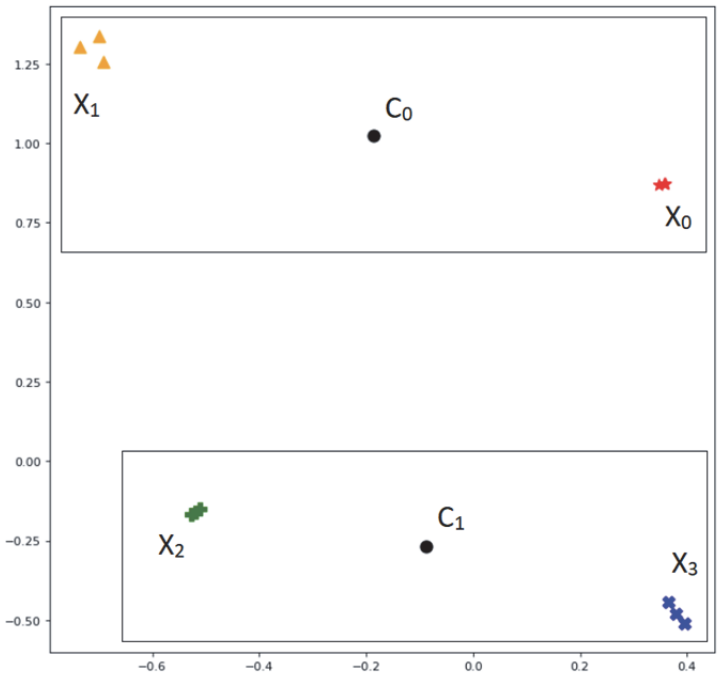

Рис. 3. Результат кластеризации для первых 12 векторов

матрицы $T_{[500,4]} = \{t_{ij} \mid i \in [0, 500), j \in [0, 4)\}$, указывающей, что вектор X_1 должен перейти в кластер C_1.

$$t_{ij} = \begin{cases} 1000, & i = 1, \quad j = 1 \\ 0, & \text{иначе} \end{cases}$$

Результаты 100 эпох работы алгоритма кластеризации представлены на Рис. 4. Вместе с вектором X_1 в кластер C_1 переместились и все остальные векторы 2 класса, при этом можно заметить, что взаимное расположение между 3-м и 4-м классами сохранилось. Также сохранилось взаимное расположение большинства векторов внутри каждого класса и относительно центра соответствующего кластера.

Во втором эксперименте предположим, что исследователь обладает информацией, о том, что вектор X_2 семантически далек от вектора X_3, поэтому он должен выйти из кластера C_1. При этом кластер реципиент неизвестен исследователю, в случае более чем 2-х кластеров предпочтение никакому из кластеров исследователь не отдает. Для алгоритма кластеризации формируется обратная связь в виде матрицы $T_{[500,4]} = \{t_{ij} \mid i \in [0, 500), j \in [0, 4)\}$ указывающей, что вектор X_2 должен выйти кластера C_1.

$$t_{ij} = \begin{cases} -1000, & i = 2, \quad j = 1 \\ 0, & \text{иначе} \end{cases}$$

Результаты 100 эпох работы алгоритма кластеризации представлены на Рис. 5. Вектор X_2 покинул кластер C_1 и вместе с вектором X_2 в кластер C_1 переместились и все остальные векторы 3 класса. При этом можно заметить, что т.к. исключение из кластера это, по сути, ослабление силы притяжения между вектором и центром кластера, то исключаемый класс оказался максимально далеко от центра кластера C_1 и достаточно далеко от центра кластера C_0. Также можно заметить предсказуемую более низкую скорость сходимости алгоритма кластеризации при операции исключения из кластера, чем при операции включения в кластер.

В третьем эксперименте предположим, что исследователь обладает информацией о необходимости изменить расположение сразу двух векторов. Вектор X_1 требуется перемесить в кластер C_1, а вектор X_2 переместить в кластер C_0. Для алгоритма кластеризации формируется обратная связь в виде матрицы $T_{[500,4]} = \{t_{ij} \mid i \in [0, 500), j \in [0, 4)\}$ следующего вида.

$$t_{ij} = \begin{cases} 1000, & i = 1, \quad j = 1 \\ 1000, & i = 2, \quad j = 0 \\ 0, & \text{иначе.} \end{cases}$$

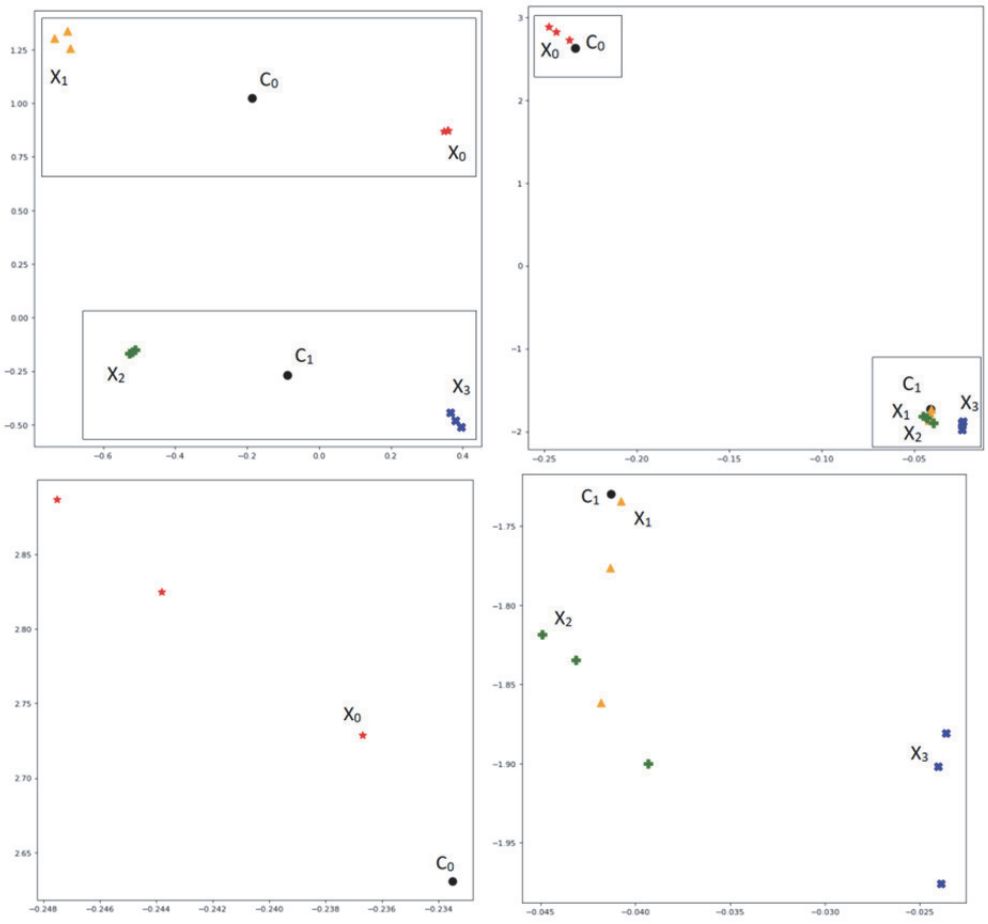

Рис. 4 (a,b,c,d). Результат кластеризации для перемещения вектора X_1 в кластер C_1:
4a – исходные данные, 4b – результат кластеризации,
4c и 4d – увеличенное представление полученных кластеров

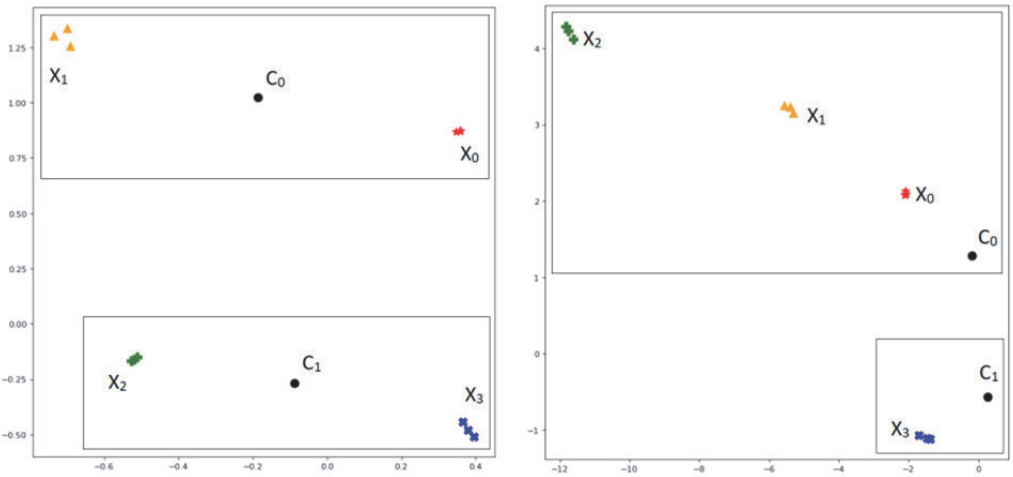

Рис. 5 (a,b). Результат кластеризации для исключения вектора X2 из кластера C1: 5a – исходные данные, 5b – результат кластеризации

Результаты 100 эпох работы алгоритма кластеризации представлены на Рис. 6. Классы 2 и 3 поменялись местами вслед за своим представителями векторами X_1 и X_2.

В четвертом эксперименте изменим набор данных, увеличив уровень шума в векторах в десять раз добавлением к каждой компоненте случайной величины из равномерного распределения U[0, 1]. Первые 4 вектора и результат кластеризации указаны в Таблице 2. На Рисунке 7a показаны первые 12 векторов и результат кластеризации. Видно, что векторы X_0 и X_2 соотнесены с кластерами неверно. Для исправления результата алгоритма кластеризации формируется обратная связь в виде матрицы $T_{[500,4]} = \{t_{ij} \mid i \in [0, 500), j \in [0, 4)\}$ следующего вида:

$$t_{ij} = \begin{cases} 1000, & i = 0, \quad j = 0 \\ 1000, & i = 2, \quad j = 1 \\ 0, & \text{иначе.} \end{cases}$$

Результаты 100 эпох работы алгоритма кластеризации представлены на Рис. 7b. Ошибочно соотнесенные векторы перешли в корректные классы, при этом остальные представители классов по-прежнему соотнесены корректно, т.к. внесенные изменения объективно улучшили качество кластеризации, и изменения оказали точечное воздействие, в отличие от предыдущих экспериментов.

Демонстрация работы на примере набора данных «Ирисы Фишера». Набор данных “Ирисы Фишера” является классическим для задач классификации и кластеризации [17]. В наборе представлены длина и ширина наружной и внутренней долей околоцветника для трех видов (‘setosa’, ‘versicolor’, ‘virginica’). В Таблице 3 представлены примеры 3-х векторов по одному для каждого вида. К исходным данным к каждой компоненте добавлена случайная величина из равномерного распределения U[0, 1/10] для усложнения задачи и генерации 600 примеров из 150 имеющихся.

На Рисунке 8a. представлены результаты работы алгоритма кластеризации для 3-х кластеров. Кластер, соответствующий виду ‘setosa’, отчетливо и безошибочно отделен,

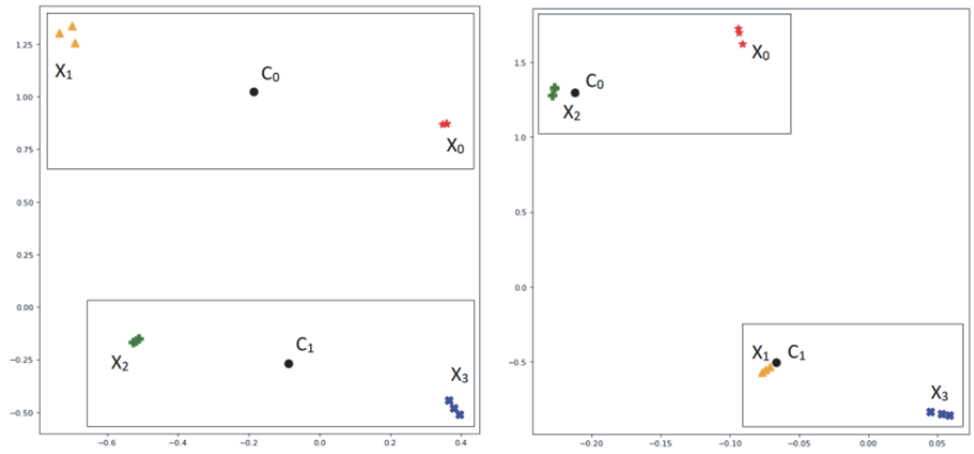

Рис. 6 (a,b). Результат кластеризации замены местами векторов X1 и X2 в кластерах: 6a – исходные данные, 6b – результат кластеризации

Таблица 2. Векторы с увеличенным уровнем шума

X	Координаты				Кластер
$\mathbf{X_0}$	**1.191519**	0.6221088	0.4377278	0.7853585	C_1
$\mathbf{X_1}$	0.7799758	**1.272592**	0.2764643	0.8018722	C_0
$\mathbf{X_2}$	0.9581394	0.8759326	**1.357817**	0.5009951	C_0
$\mathbf{X_3}$	0.6834629	0.7127021	0.3702508	**1.561196**	C_1

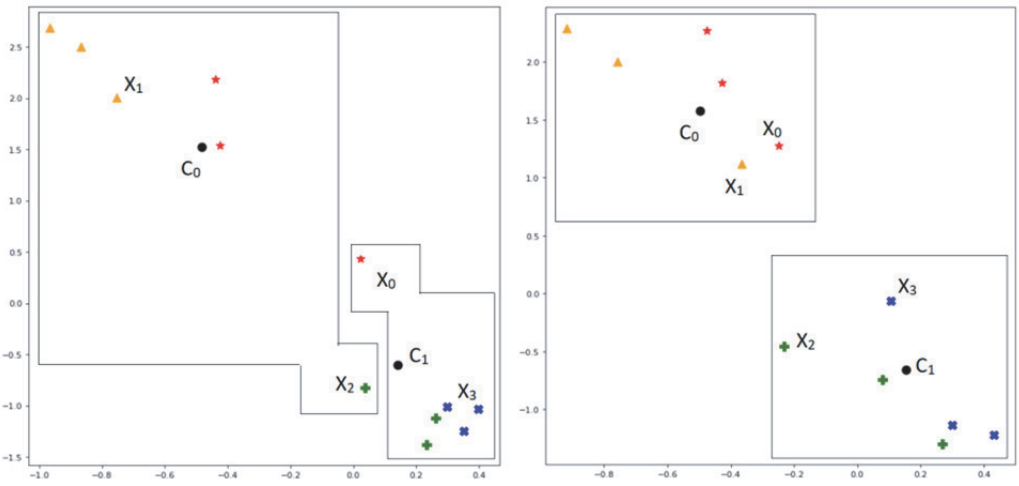

Рис. 7 (a,b). Результат кластеризации векторов с увеличенным шумом с точечным исправлением: 7a – исходные данные, 7b – результат кластеризации

Таблица 3. Пример векторов из набора данных «Ирисы Фишера»

X	Координаты				Кластер
$\mathbf{X_0}$	5.1191519	3.5622108	1.4437727	0.2785358	C_0 (setosa)
$\mathbf{X_1}$	7.0779975	3.2272592	4.7276464	1.4801872	C_1 (versicolor)
$\mathbf{X_2}$	6.3958139	3.3875932	6.0357817	2.5500995	C_2 (virginica)

а в кластерах двух других видов имеется 13 и 16 перепутанных местами векторов. Виды ‘versicolor’ и ‘virginica’ являются близкими, и даже в задачах классификации их не удается однозначно разделить. Тем не менее, предположим, что исследователь обладает информацией о двух векторах (экземплярах цветка, видовая принадлежность которых ему вполне могла быть известна), которые необходимо поменять местами. В данном примере это векторы с порядковым номером 7 и 50. Для этого формируется обратная связь в виде матрицы $T_{[600,3]} = \{t_{ij} \mid i \in [0, 600), j \in [0, 3)\}$ следующего вида:

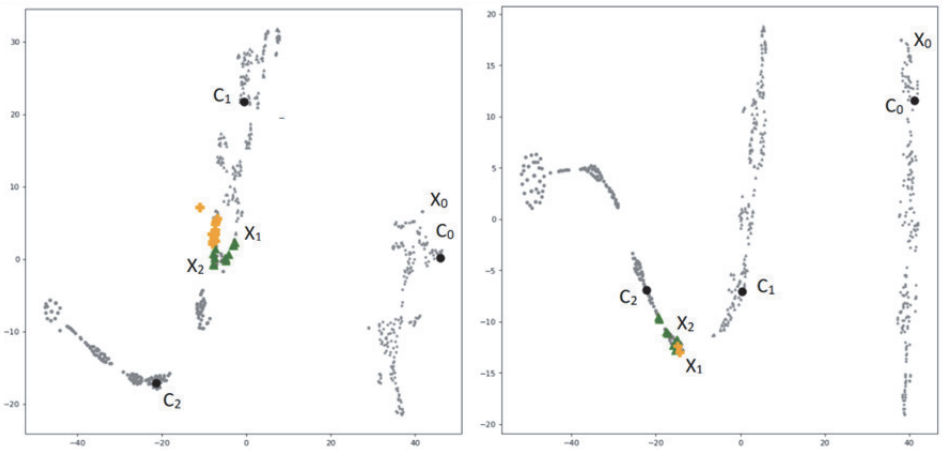

Рис. 8 (a,b). Результат изменения распределения по кластерам в наборе «Ирисы Фишера». Количество неверно соотнесенных векторов уменьшилось с 26 (8a) до 10 (8b)

$$t_{ij} = \begin{cases} 1000, & i = 7, \quad j = 2 \\ 1000, & i = 50, \quad j = 1 \\ 0, & \text{иначе.} \end{cases}$$

Результаты 100 эпох работы алгоритма кластеризации представлены на Рис. 8b (для проекции на плоскость 3-х мерных данных использовался алгоритм t-SNE, реализованный в python библиотеке sklearn [21]). Ошибочно соотнесенные векторы перешли в корректные классы, при этом большая часть ошибок также исправилась. Перепутанными остались 8 и 2 векторов во 2-ом и 3-м классах соответственно. Это дает точность (accuracy), равную 0.98(3). Такие результаты трудно достижимы даже для алгоритмов классификации, средним результатов для лучших из которых является 0.971. Алгоритмы кластеризации зачастую полностью не способны различить 2-ой и 3-ий виды [17].

Демонстрация работы на примере набора данных Reuters (RCV1-v2/LYRL2004). Набор данных RCV1-v2 содержит 810 000 новостей на английском языке новостного агентства Reuters.

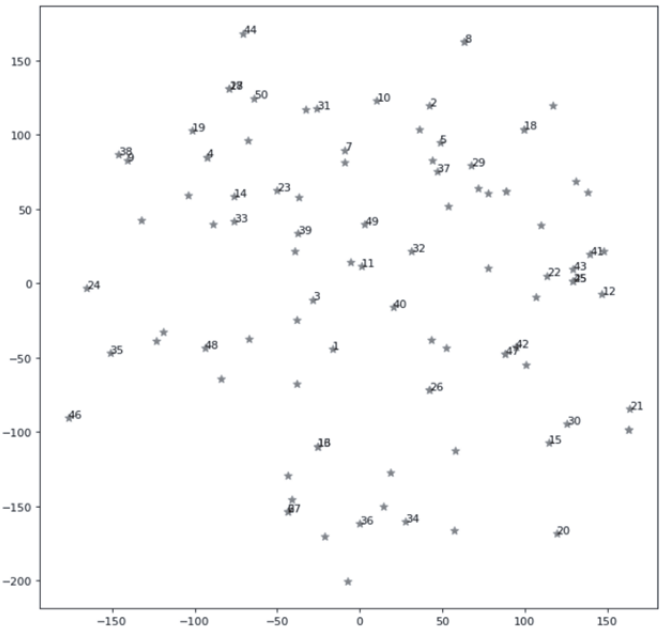

Рис. 9. Результат работы первой итерации алгоритма кластеризации

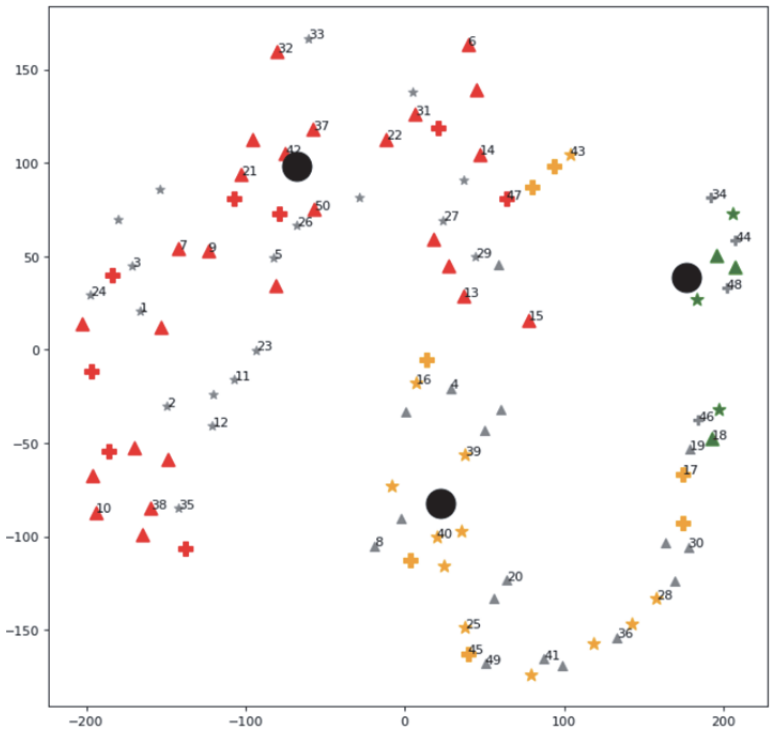

Рис. 10. Результат работы первой итерации алгоритма кластеризации

Новости классифицированы по 103 категориям, которые, в свою очередь, сгруппированы по 4-м крупным тематикам: государство и социальная сфера, экономика, корпоративные новости и производство, биржевые рынки и акции. Данный набор данных является классическим для проведения сравнения эффективности алгоритмов кластеризации. В частности в книге [6] в главе "Semi-supervised Clustering with User Feedback" алгоритм кластеризации без учителя применялся к набору данных Reuters, из которого были отобраны 5 групп по 25 новостных статей по различным подгруппам. Т.к. в статье не описывается принцип формирования групп, а в ряде других статей [25] используется кластеризация текстов по 4-м корневым категориям, для проведения эксперимента было отобрано 4 группы по 32 новостных статьи в каждой. Были выбраны статьи из следующих тематик:

- C21 – продукты и производство, поставки, сервисы и мероприятия, минеральная и сельскохозяйственная промышленность (products and production, output, services and activities, mineral and agricultural production)
- E41 – трудоустройство, безработица, взаимоотношения с работодателями (employment, unemployment, labour and labour relations)
- GCRIM – Гражданское и уголовное право, правовые нарушения, нарушения порядка, преступления в сфере оборота наркотиков, преступления в сфере бизнеса, преступления, мошенничество, убийства, преступники, мафия, полиция (civil and criminal law; law and order issues, drug related crime, corporate crime; crime; fraud; murder; criminals; mafia; police)
- M11 – Рыночные курсы, биржевые котировки (stock exchanges, performance of equities)

В качестве векторов признаков были использованы стандартные векторы набора данных RCV1-v2 на основе TF-IDF, т.к. именно такой способ был использован экспериментах для сравнения. Эти векторы имеют размерность 47 236.

Первая итерация алгоритма показала чистоту кластеров в 25-50%. На рисунке 9 представлены результаты после преобразования t-SNE к двумерному пространству. Как видно, явное разделение кластеров отсутствует. Для сравнения в [6] приводятся следующие результаты: на первом этапе кластеризации без учителя чистота кластеризации колеблется от 44-50% и далее при получении 5-15 ограничений виде обратной связи чистота кластеров повышается и достигает 70-80%.

В проводимом эксперименте кластеризации с помощью представленного в данной работе метода, применение 5-10 ограничений приводит к повышению чистоты кластеров в 75-85%. Что в среднем на 5% выше результатов в работе, выбранной для сравнения, при этом объем необходимых дополнительных ограничений в 1,5-2 раза меньше.

При этом стоит отметить, что достигнутый результат сравним с результатами классификации (обучение с учителем), как в случае, если бы все значения меток для всего набора данных, использованного в обучении, были бы известны.

ЗАКЛЮЧЕНИЕ

В данной работе показана актуальность проблемы разработки методов интерактивной кластеризации с обратной связью, особенно с учетом большого числа современных методов кластеризации без учителя, демонстрирующих выдающиеся результаты. В связи с этим, представлен подход к построению метода интерактивной кластеризации с учетом обратной связи на базе современных непрерывных методов кластеризации с использованием нейронных сетей. В частности представлена реализация на основе метода кластеризации DEC, и продемонстрирована работоспособность и эффективность подхода. При этом замечена недостаточная чувствительность алгоритма к обратной связи вида «исключить из кластера».

В качестве следующих этапов работы предполагается продолжить исследования вариантов выбора целевого вспомогательного распределения и его влияния на качество и эффективность кластеризации. Также планируется доработать подход, увеличив количество допустимых видов обратной связи. Это позволит накладывать ограничения на попарную связь элементов в кластере и на общую структуру кластеров. В частности, для ряда задач важным является равномерность распределения элементов по кластерам. Кроме того, планируется доработка метода для повышения эффективности уже имеющихся видов обратной связи.

БЛАГОДАРНОСТИ

Исследование выполнено при финансовой поддержке РФФИ и Правительства Ульяновской области в рамках научного проекта № 18-47-00019

СПИСОК ЛИТЕРАТУРЫ

1. *Aljalbout E., Golkov V., Siddiqui Y., Strobel M., Cremers D.* Clustering with Deep Learning: Taxonomy and New Methods // arXiv:1801.07648, 2018.
2. *Bae J., Helldin T., Riveiro M. Nowaczyk S., Bouguella M., Falkman G.* Interactive Clustering: A Comprehensive Review // ACM Comput. Surv., 2020, Vol. 53, No. 1.
3. *Bagherjeiran A., Eick C. F., Chen C.-S., Vilalta R.* Adaptive clustering: obtaining better clusters using feedback and past experience // Fifth IEEE International Conference on Data Mining (ICDM'05), Houston, TX, 2005.

4. *Balcan M.F., Blum A.* Clustering with Interactive Feedback. // Freund Y., Györfi L., Turán G., Zeugmann T. (eds) Algorithmic Learning Theory. Lecture Notes in Computer Science, vol 5254. Springer, Berlin, Heidelberg, 2008.
5. *Basu S., Banerjee A., Mooney R.* Semi-supervised Clustering by Seeding // In Proceedings of 19th International Conference on Machine Learning, 2002.
6. *Basu S., Davidson I., Wagstaff K.* Constrained Clustering: Advances in Algorithms, Theory, and Applications // CRC Press, 2008.
7. *Basu S., Fisher D., Drucker S.M., Lu H.* Assisting Users with Clustering Tasks by Combining Metric Learning and Classification. // Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, 2010.
8. *Dasgupta S., Ng V.* Which Clustering Do You Want? Inducing Your Ideal Clustering with Minimal Feedback // arXiv:1401.5389, 2014. -https://arxiv.org/abs/1401.5389.
9. *Demiriz A., Bennett K.P., Embrechts M.J.* A Genetic Algorithm Approach for Semi-Supervised Clustering. // International Journal of Smart Engineering System Design, 2002, vol. 4.
10. *Dizaji K.G., Herandi A., Deng C., Cai W., Huang H.* Deep Clustering via Joint Convolutional Autoencoder Embedding and Relative Entropy Minimization. // IEEE International Conference on Computer Vision (ICCV), Venice, 2017.
11. *Dudarin P.V., Tronin V.G., Svyatov K.V.* A Technique to Pre-trained Neural Network Language Model Customization to Software Development Domain // Kuznetsov S., Panov A. (eds) Artificial Intelligence. RCAI 2019. Communications in Computer and Information Science, vol 1093. Springer, Cham, 2019.
12. *Fatehi K., Bozorgi A., Zahedi M.S., Asgarian E.* Improving semi-supervised constrained k-means clustering method using user feedback. // Journal of Computing and Security, 2014, Volume 1, Number 4.
13. *Greff K., van Steenkiste S., Schmidhuber J.* Neural Expectation Maximization. // Advances in Neural Information Processing Systems 30, 2017.
14. *Hastie T., Tibshirani R., Friedman J.* The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Second Edition. // Springer Series in Statistics book series, 2009.
15. *Hoffer E., Ailon N.* Deep Metric Learning Using Triplet Network // In: Feragen A., Pelillo M., Loog M. (eds) Similarity-Based Pattern Recognition. Lecture Notes in Computer Science, vol 9370. Springer, Cham, 2015.
16. *Huang Y.* Mixed-Iterative Clustering // PhD thesis at Language Technologies Institute School of Computer Science Carnegie Mellon University Pittsburgh, PA 15213, 2010.
17. *Leela V., Sakthipriya K., Manikandan R.* Comparative Study of Clustering Techniques in Iris Data Sets // World Applied Sciences Journal 29 (Data Mining and Soft Computing Techniques), 2014.
18. *Li L., Kameoka H.* Deep Clustering with Gated Convolutional Networks // IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, 2018.
19. *Meier B.B., Elezi I., Amirian M., Dürr O., Stadelmann T.* Learning Neural Models for End-to-End Clustering. // Artificial Neural Networks in Pattern Recognition edited by Pancioni L., Schwenker F., Trentin E., Lecture Notes in Computer Science, vol 11081. Springer, Cham, 2018.
20. *Nebu C.M., Joseph S.* Semi-supervised clustering with soft labels // International Conference on Control Communication & Computing India (ICCC), Trivandrum, 2015.
21. *Pedregosa F., Varoquaux G, Gramfort A., Michel V., Thirion B., Grisel O., Blondel M., Prettenhofer P., Weiss R., Dubourg V., Vanderplas J., Passos A., Cournapeau D., Brucher M., Perrot M., Duchesnay É.* Scikit-learn: Machine Learning in Python. // Journal of Machine Learning Research, 2011, vol. 12.
22. *Pedrycz W.* Algorithms of fuzzy clustering with partial supervision // Pattern Recognition Letters, Volume 3, 1985.
23. *Suresh T., Meena Abarna K.T.* LSTM Model for Semantic clustering of user-generated content using AI Geared to wearable Device // Semanticscholar. org Corpus ID: 212585860, 2017. - URL: https://www.semanticscholar.org/paper/LSTM-Model-for-Semantic-clustering-of-content-using-Suresh-Abarna/7b72349284b78803fe2581a041e5c7a19a081bdc
24. *Wang Z., Mi H., Ittycheriah A.* Semi-supervised Clustering for Short Text via Deep Representation Learning // Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning, Association for Computational Linguistics, Berlin, Germany, 2016.
25. *Xie J., Girshick R., Farhadi A.* Unsupervised deep embedding for clustering analysis // ICML'16: Proceedings of the 33rd International Conference on International Conference on Machine Learning, 2002.
26. *Xu J., Xu B., Wang P., Zheng S., Tian G., Zhao J.* Self-Taught Convolutional Neural Networks for Short Text Clustering // IEEE Neural Networks, 2017, Volume 88.
27. *Yang C., Shi X., Jie L., Han J.* I Know You'll Be Back: Interpretable New User Clustering and Churn Prediction on a Mobile Social Application // the 24th ACM SIGKDD International Conference, 2018.
28. *Yang J., Parikh D., Batra D.* Joint Unsupervised Learning of Deep Representations and Image Clusters // IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016.
29. *Yang B., Fu X., Sidiropoulos N.D., Hong M.* Towards K-means-friendly spaces: Simultaneous deep learning and clustering // Proceedings of the 34th International Conference on Machine Learning, Volume 70, 2017
30. *Дударин П.В., Пинков А.П., Ярушкина Н.Г.* Методика и алгоритм кластеризации объектов экономической аналитики // Автоматизация процессов управления. 2017. №1.
31. *Дударин П.В., Ярушкина Н.Г.* Алгоритм построения иерархического классификатора коротких текстовых фрагментов на основе кластеризации нечеткого графа // Радиотехника. 2017. № 6.
32. *Дударин П.В., Тронин В.Г., Святов К.В., Белов В.А., Шакуров Р.А.* Подход к оценке трудоемкости задач в процессе разработки программного обеспечения на основе нейронных сетей // Автоматизация процессов управления. 2019. № 3.
33. *Дударин П.В., Ярушкина Н.Г.* Подход к трансформации кластерного дерева признаков в векторное пространство признаков // Радиотехника. 2018. № 6.

34. *Шелехова Н.В., Римарева Л.В.* Управление технологическими процессами производства алкогольной продукции с применением информационных технологий // Хранение и переработка сельхозсырья, Пищевая промышленность. 2017. № 3.

35. *Шелехова Н.В., Поляков В.А., Серба Е.М., Шелехова Т.М., Веселовская О.В., Скворцова Л.И.* Информационные технологии в аналитическом контроле качества алкогольной продукции // Пищевая промышленность. 2018. №8.

AN APPROACH TO USER FEEDBACK PROCESSING ORDER TO INCREASE QUALITY OF CLUSTERING RESULTS

© 2020 P.V. Dudarin, V.G. Tronin, N.G. Yarushkina

Ulyanovsk State Technical University, Ulyanovsk, Russia

Dataset clustering could have more than one “right” result depending on a user intention. For example, texts could be clustered according to their topic, style or author. In case of unsatisfactory results, a data scientist needs to re-construct a feature space in order to change the results. The relation between the feature space and the result are often quite complicated. The latter results in building several clustering models to explore useful relations. Interactive clustering with feedback is aimed to cope with this problem. In this paper an approach to user feedback processing during clustering is presented. The approach is based on end-to-end clustering and uses an autoencoder neural network. This technique allows to adjust iteratively the computing clusters without changing feature space.

Key words: clustering, interactive clustering, mixed-initiative clustering, constrained clustering, semi-supervised clustering, end-to-end clustering, learning to cluster, clustering with intent, deep embedding, deep representation, feedback, neural networks.

DOI: 10.37313/1990-5378-2020-22-5-94-105

Pavel Dudarin, Post Graduate Student.
E-mail: p.dudarin@ulstu.ru
Vadim Tronin, Candidate of Technical Science, Associate Professor. E-mail: v.tronin@ulstu.ru
Nadezhda Yarushkina, Doctor of Technical Science, Professor, Rector. E-mail: jng@ulstu.ru