МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ

Математико-механический факультет
Кафедра прикладной кибернетики

Путников Семен Андреевич

Курсовая работа

Алгоритмы кластеризации: выявление профилей пользователей по анализу активности

Руководитель курсовой работы
_____ Н. В. Кузнецов
«___» _____ 2019 г.

Санкт-Петербург, 2019 г.

Содержание

1 Введение

Мы живем в постиндустриальном обществе, где как известно: "кто владеет информацией тот владеет миром". Люди стараются собирать сведения обо всем: погоде, экономике, политике, демографии и во многих других отрослях. Каждая компания старается собрать как можно больше данных, надеясь занять лидирующую позицию. Однако сама по себе информация не представляет особой ценности, это своего рода сырье, которое еще нужно обработать. Только после анализа информация представляет для человека или компании определенную ценность, стоит отметить, что чем глубже и точнее произведен анализ информации, тем ценнее его результат.
В рамках анализа и обработки информации появился термин Data Mining, который означает: добыча данных, интеллектуальный анализ данных, глубинный анализ данных. Достаточно часто Data Mining связывают с Business intelligence, которые в свою очередь широко применяются в бизнесе.
Задачи, решаемые Data Mining:

1. Классификация — отнесение входного вектора (объекта, события, наблюдения) к одному из заранее известных классов.

2. Кластеризация — разделение множества входных векторов на группы (кластеры) по степени «похожести» друг на друга.

3. Сокращение описания — для визуализации данных, упрощения счета и интерпретации, сжатия объемов собираемой и хранимой информации.

4. Ассоциация — поиск повторяющихся образцов. Например, поиск «устойчивых связей в корзине покупателя».

5. Прогнозирование – нахождение будущих состояний объекта на основании предыдущих состояний (исторических данных)

6. Анализ отклонений — например, выявление нетипичной сетевой активности позволяет обнаружить вредоносные программы.

7. Визуализация данных.

Рассмотрим задачу кластеризации. Она является одной из важнейших задач Data Mining. Благодаря ей существует возможноть прогнозировать, а так же получать детальную информацию об уже имеющихся кластерах. Например, в рамках государства это означает, что администрация может оперативно и точно получать информацию о социальных слоях общества, сегментах бизнеса, категориях годности граждан и др., это позволяет аппарату управления быстро реагировать на изменения, а так же задавать вектор развития страны.

2 Постановка задачи

В данной работе предлагается рассмотреть следующую задачу - анализ имеющихся алгоритмов класстеризации и создание пайплайна для построения класстеризации по массиву данных. Этот массив представляет собой логи REST-запросов пользователей на портале учета древесины и сделок с ней. Проведем декомпозицию задачи и выделим следующие этапы.

- Анализ существующих алгоритмов кластеризации
- Исследование инструментов Elasticsearch
- Обработка массива данных
- Реализация алгоритма кластеризации для поставленной задачи
- Анализ полученных результатов

3 Теоретический аспект

3.1 Метрики

Для работы алгоритмов кластеризации необходимо ввести метрику на данных.

Определение 1. $\rho(x, y) : X \times X \to \mathbb{R}$ *называют метрикой, если выполняются следующие условия:*

1. $\rho(x, y) = 0 \Leftrightarrow x = y$ *(аксиома тождества)*
2. $\rho(x, y) = \rho(y, x)$ *(аксиома симметрии)*
3. $\rho(x, z) \leq \rho(x, y) + \rho(y, z)$ *(неравенство треугольника)*

Приведем наиболее распространенные в Data Mining метрики расстояний.

1. Евклидова метрика

$$\rho(x, y) = \sqrt{\sum_{i}^{n}(x_i - y_i)^2} \tag{1}$$

2. Квадрат евклидова расстояния

$$\rho(x, y) = \sum_{i}^{n}(x_i - y_i)^2 \tag{2}$$

3. Манхэттенская метрика

$$\rho(x, y) = \sum_{i}^{n}|x_i - y_i| \tag{3}$$

4. Расстояние Чебышева

$$\rho(x, y) = \max_{i}|x_i - y_i| \tag{4}$$

5. Степенное расстояние

$$\rho(x, y) = \sqrt[r]{\sum_{i}^{n}(x_i - y_i)^p} \tag{5}$$

6. Дискретная метрика

$$\rho(x, y) = \begin{cases} 0, x = y, \\ 1, x \neq y. \end{cases} \tag{6}$$

3.2 Feature engineering

Первый этап процесса класстеризации - отбор признаков. Они бывают различных типов: бинарные, вещественные, категориальные, текстовые и другие. В исследуемых нами данных признаки текстовые и категориальные. К категориальным относят признаки, значение которых нельзя сравнивать между собой, можно лишь проверять их равенство. Текстовый признак представляет из себе последовательность слов. Рассмотрим наиболее распространенные алгоритмы их отбора.

- **One-hot encoding или "Мешок слов".** В рамках этого методы строится словарь всех уникальных слов в датасете, а потом каждому слово приобретает уникальный индекс. Тогда каждое предложение можно будет отобразить списком, длина которого равна числу уникальных слов в словаре, а в каждом индексе в этом списке будет хранится, сколько раз данное слово встречается в предложении.

Рис. 1: пример работы one-hot encoding.

- **TF-IDF.** В этом методе оценивается не количество вхождений слов, а их оценки важности для текста. Руководствуемся двумя принципами: чем чаще слово встречается в документе, тем оно важнее, и, чем реже слово встречается в остальных документах, тем оно важнее. Рассмотрим формулу на их основе.
  n_{iw} (term frequency) - число вхождений слова w в текст $x_i{}^j$
  N_w (document frequency) - число текстов, содержащих w
  Тогда важность слова w для документа $x_i{}^j$ исчисляется по формуле:

$$TF - IDF(i, w) = n_{dw} log(l/N_w)$$

3.3 Класстеризация

Класстеризация - это отдельный класс задач машинного обучения, отличиющийся от классификации тем, что у объектов обучающей выборки нет заранее заданных ответов учителя. Задача состоит в выделении отдельных класстеров, состоящих из близких объектов так, чтобы объекты разных класстеров существенно различались. Рассмотрим основные алгоритмы класстеризации.

- **Метод k-средних.** На входе этого метода некоторым образом расставляются k первоначальных центров класстеров. Потом для каждого объекта

вычислется его близость к каждому из класстеров. Затем вычисляется новый центр класстера. И так повторяется пока центры класстеров не стабилизируются.

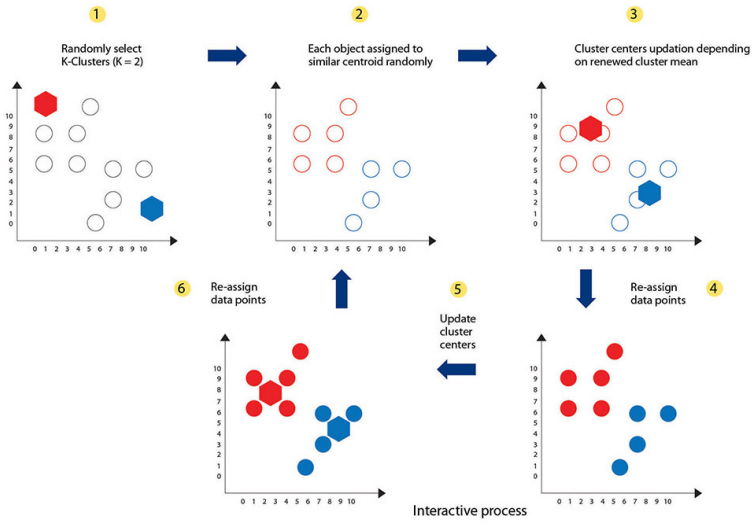

Рис. 2: пример работы метода k-средних.

- **Алгоритм Ланса-Уильямса.** Этот алгоритм относится к классу агломеративных методов иерархической класстеризации. Агломеративность означает то, что метод объединяет мелкие класстеры в более крупные и отображают историю этих объединений в виде дендрограммы.

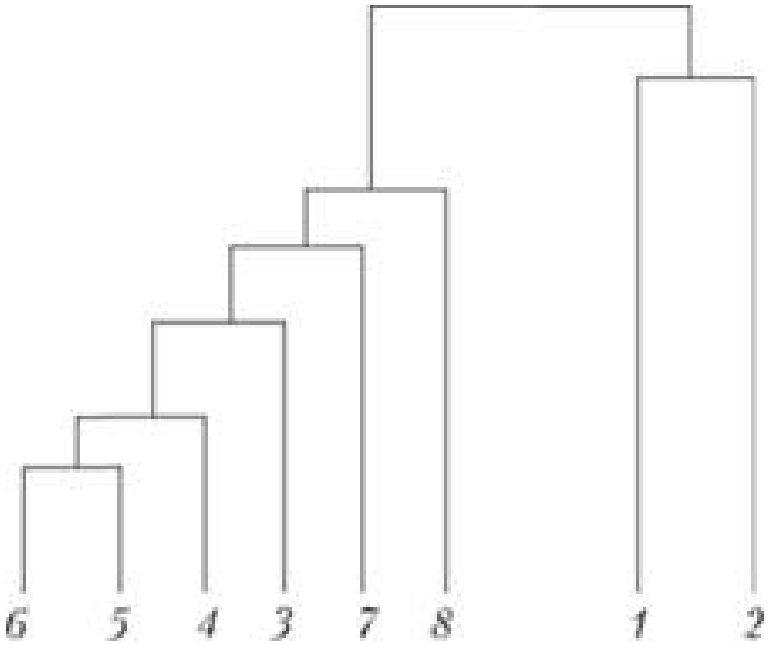

Рис. 3: пример дендрограммы.

Алгоритм основан на том, что сначала мы берём все кластеры одноэлементные, то есть каждый кластер соответствует какой-то одной точке, и кластеров ровно столько, сколько объектов обучающей выборки. А дальше мы находим в текущем множестве кластеров два самых ближайших и

сливаем их в один кластер. Повторяем это ровно до того момента, пока все кластеры не объединятся в один.

Отдельно стоит отметить формулу Ланса-Ульямса, которая используется для нахождения расстояния между кластером W, полученным из объединения U и V, и кластера S, зная расстояния между исходными кластерами.

$$R(U \cup V, S) = \alpha_U R(U, S) + \alpha_V R(V, S) + \beta R(U, V) + \gamma |R(U, S) - R(V, S)|$$

Частные случаи формулы:

1. **Расстояние ближайешего соседа:**

$$R(W, S) = \min_{w \in W, s \in S} \rho(w, s);$$
$$\text{где } \alpha_U = \alpha_V = \frac{1}{2},\ \beta = 0,\ \gamma = -\frac{1}{2}$$

2. **Расстояние дальнего соседа:**

$$R(W, S) = \max_{w \in W, s \in S} \rho(w, s);$$
$$\text{где } \alpha_U = \alpha_V = \frac{1}{2},\ \beta = 0,\ \gamma = \frac{1}{2}$$

3. **Расстояние мужду центрами:**

$$R(W, S) = \rho^2(\sum_{w \in W} \frac{w}{|W|}, \sum_{s \in S} \frac{s}{|S|});$$
$$\text{где } \alpha_U = \frac{|U|}{|W|},\ \alpha_V = \frac{|V|}{|W|},\ \beta = -\alpha_U \alpha_V,\ \gamma = 0$$

4. **Групповое среднее расстояние:**

$$R(W, S) = \frac{1}{|W||S|} \sum_{w \in W} \sum_{s \in S} \rho(w, s);$$
$$\text{где } \alpha_U = \frac{|U|}{|W|},\ \alpha_V = \frac{|V|}{|W|},\ \beta = \gamma = 0$$

4 Практический аспект

Во время поиска практического решения задачи. Появился ряд проблем связанных с размером массива данных. Входные данные представляют собой JSON-файл весом более 100 гигабайт. На персональных машинах такие объемы не обработать, поэтому был создан пайплайн сборки датасета по кусочкам.
Первым делом командой в терминале разбиваем исходный файл на кусочки по 10000 строк и перенесем их в отдельную папку.

```
mkdir parts && cd parts && split -d -l 10000 --additional-suffix=.
    json ../data.json ex && cd ..
```

Дальше рассмотрим сприпт, который обработает файлы по кусочкам и соберет в один датасет. Обходя дерево json, из первой строки получаем название всех признаков.

```
def getIndexList(dataJson):
        indexSet = set()
        for i in dataJson:
                for k in i.keys():
                        if(k != '_source'): indexSet.add(k)
                for j in i['_source'].keys():
                        indexSet.add(j)
        indexList = list(indexSet)
        return indexList
```

По полученному списку создаем пустой Data Frame. Следущим шагом мы обходим по строчке файл и записываем данные в frame.

```
def writeData(indexList,dataJson,i):
        data = getDataFrame()
        for n in atpbar(range(len(dataJson)), name='LOCAL {}'.format(i)):
                pattern = defaultdict(list)
                for g in indexList:
                        pattern[g].append(None)
                for k in dataJson[n].keys():
                        if(k != '_source'): pattern[k] = dataJson[n][k]
                for j in dataJson[n]['_source'].keys(): pattern[j] = dataJson[n
                    ]['_source'][j]
                data = data.append(pd.DataFrame.from_dict(pattern),
                    ignore_index=True, sort = False)
        return data
```

Как только этот json файл закончится, переходим к следующему файлу и повторяем предыдущие шаги.

5 Итог

Подводя итог, можно сказать, что цели проекта были частично достигнуты. Была исследованы существующие методы отбора признаков и класстеризации данных, а также их дальнейшей валидации. В результате практической работы был создан скрипт обработки и подготовки данных для применения класстеризации.

В процессе работы над проектом были предолены ряд сложностей связанных с размером входных данных. Изучены методы для обработки таких массивов.

Проект может иметь продолжение с целями оптимизации алгоритмов класстеризации, а также внедрения данного решения в существующие системы тестирования веб-портала по учету древесины и сделок с ней.

Список литературы

[1] Воронцов К.В. Методы кластеризации –

$$http://shad.yandex.ru/lectures/machine_learning.xml$$

[2] Барсегян А. А. и др. Технологии анализа данных: Data Mining, Visual Mining, Text Mining, OLAP. 2-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2007.

[3] Вагин В. Н., Головина Е. Ю., Загорянская А. А. Достоверный и правдоподобный вывод в интеллектуальных системах. — М.: Физматлит. 2004.

[4] Кацко И. А., Human Н. Б. Практикум по анализу данных на компьютере. — М.: КолосС, 2009.