УДК 004.42

РЕАЛИЗАЦИЯ РЕКОМЕНДАТЕЛЬНОЙ СИСТЕМЫ НА ОСНОВЕ МЕТОДА К-СРЕДНИХ

Намазбек у. А., М.В. Коржов, И.В. Хмелева

Представлен один из возможных способов реализации рекомендательной системы на основе метода к-средних.

Ключевые слова: метод к-средних; рекомендательные системы; тэг; порог гравитации.

IMPLEMENTATION OF RECOMMENDATION SYSTEM BASED ON THE METHOD OF K-MEANS

Namazbek u. A., M.V. Korzhov, I.V. Khmeleva

This work represents a possible way to implement the recommendation system based on the method k-means.

Keywords: method k-means; recommendation system; tag; limit gravity.

Глобальная сеть – это место расположения различной информации, в которой бывает сложно найти нужные данные. Для упрощения процедуры поиска разработаны различные поисковые алгоритмы и системы, которые на запрос пользователя выдают ссылки на страницы с информацией, указанной в запросе. Развитие социальных сетей усовершенствовало механизмы поиска до выработки рекомендаций [1] для конкретного пользователя на основе анализа его предыдущих запросов, а это предполагает применение интеллектуального анализа данных. В работе представлен один из возможных способов реализации рекомендательной системы.

Программное приложение предназначено для навигации по культурным мероприятиям города. На рисунке 1 приведена диаграмма развертывания приложения, на которой видно, что приложение имеет двухуровневую архитектуру, обработка данных ведется на сервере, чтобы не перегружать "клиента", что позволяет пользоваться приложением с мобильных устройств старой версии.

Рассмотрим задачи каждой стороны подробнее.

Клиент. Перед клиентским приложением стоит задача сбора и анализа данных с различных устройств пользователей и передача этих данных на сервер для дальнейшего анализа и выработки рекомендаций. Процесс сбора анонимной статистики выполняется пакетом средств разработчика (SDK) в фоновом режиме и не затормаживает работу приложения.

Артефакт Parsing реализует автоматический сбор информации обо всех мероприятиях города с сайтов организаций (аналог RSS) и собирает статистику о посещаемости и предпочтениях пользователей приложения. Данные передаются в формате JSON [2], что удобно и для передачи, и для дальнейшей обработки.

Сервер. Сервер помимо хранения данных выполняет анализ собранных данных для выработки рекомендаций для каждого пользователя. Интеллектуальный анализ данных решает задачу кластеризации, которая заключается в делении множества объектов на группы (кластеры) схожих по параметрам. При этом, в отличие от классификации, число кластеров и их характеристики могут быть заранее неизвестны и определяются в ходе построения кластеров исходя из степени близости объединяемых объектов по совокупности параметров [3]. Для решения задачи кластеризации применяется комбинированный метод k-means. Алгоритм k-means является простым повторяющимся алгоритмом кластеризации, который разделяет определенный набор данных на заданное пользователем число кластеров, k. Алгоритм прост для реализации и запуска (вычислительная сложность алгоритма: $O(nkl)$, где k – число кластеров; l – число итераций), относительно быстрый, легко адаптируется и распространен на практике [4].

Рассмотрим алгоритм k-means применительно к поставленной задаче. Объектом исследования

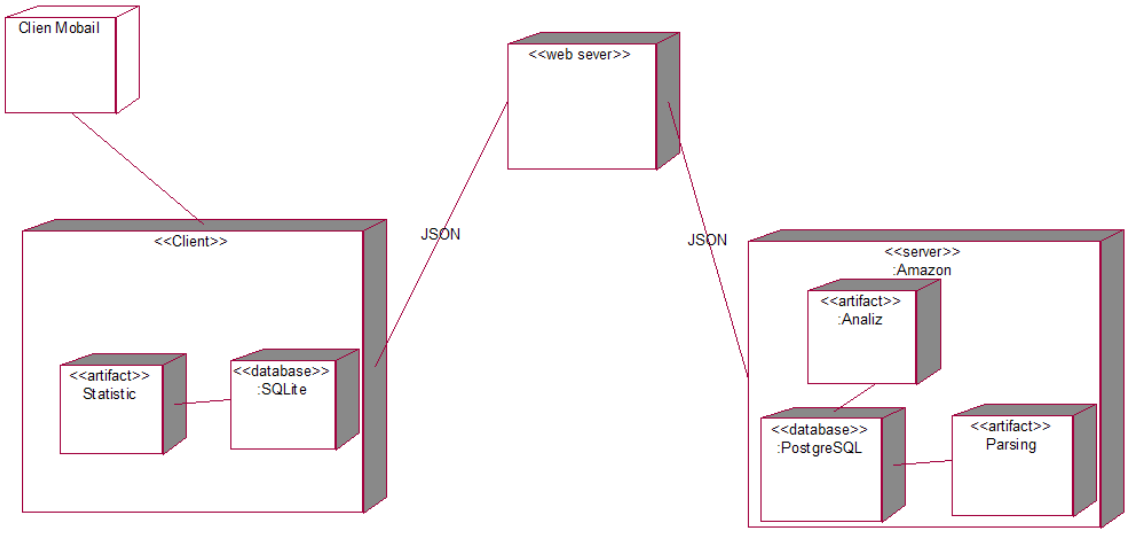

Рисунок 1 – Диаграмма развертывания

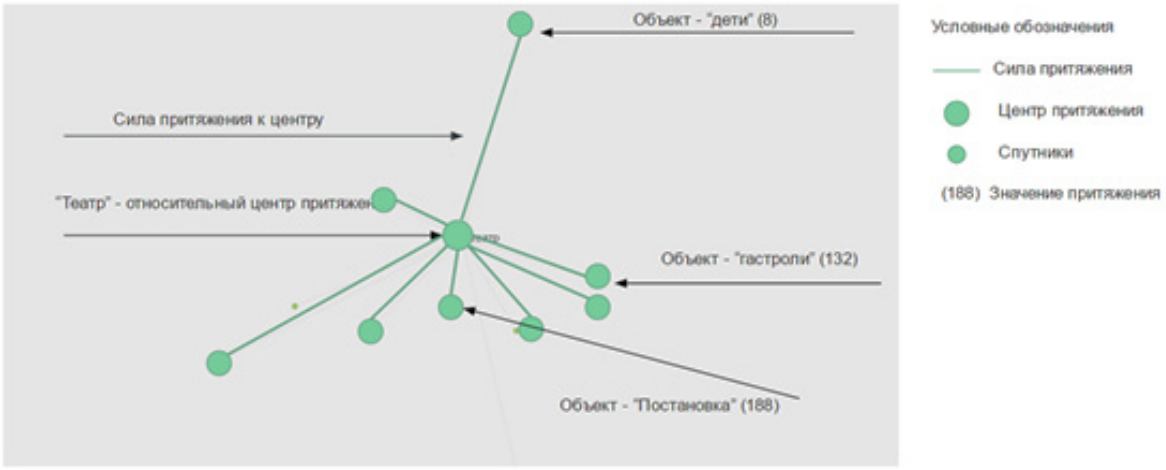

Рисунок 2 – Визуализация объекта “Театр”

являются запросы пользователей системы, обладающие набором параметров (страна, модель устройства, размер экрана, состояние сети, оперативная память, тема события, количество запросов и множество других параметров). Характеристику объекта назовем тегом (обозначим T_i). Тогда пользователя можно представить в виде n-мерного вектора тегов:

$$U = [T_1, T_2, \ldots T_n],$$

где T – это есть тег, который описан двумерным вектором:

$$T_i = [F, R],$$

где i – идентификатор тега; F – частота запросов тега пользователем; R – количество обнулений,

$$n \le N,$$

где N – общее количество тегов в базе данных.

Тег имеет численный показатель притяжения G – “сила гравитации”. Тег также является центром притяжения других тегов, в данном случае спутников. Центр тяжести можно представить N-мерным вектором тегов: $C = [T_1, T_2, \ldots T_n]$, где $n \le N$, N – общее количество тегов в базе данных. В качестве меры близости используется Евклидово расстояние [4].

Для определения схожести тегов между собой следует ввести понятия “достоверных данных” и “недостоверных данных”.

Достоверные данные – это данные, которые принимаются как заведомо истинные. В данном

случае это данные, которые принимаются сервером при создании события пользователем. Создание события всегда сопровождается наличием тегов. И если количество тегов при создании события больше одного, то теги связываются между собой, то есть, относительная "сила гравитации" между ними возрастает на единицу.

Недостоверные данные – это данные, которые проходят алгоритм проверки на определение схожести. В данном случае к недостоверным данным можно отнести запросы пользователя. Пользователь может вводить разные запросы в поиске, и нельзя достоверно сказать о том, что текущий запрос связан с последующим или с предыдущим. Для этого вводится понятие "Порог гравитации" – это численный показатель частоты набора пользователем конкретного тега.

Пример визуализации работы алгоритма для объекта "театр" приведен на рисунке 2.

Здесь пользователь представлен в виде вектора тегов $U = [T_1, T_2, T_3]$. Порог гравитации равен 10, $T_1 = [10,0]$, $T_2 = [2,0]$, $T_3 = [8,0]$. Следует обратить внимание на первые аргументы векторов: (10,2,8). Когда порог гравитации достигает границы, это четко наблюдается у $T_1 = [10,0]$, то данный тег обнуляется, связавшись с ближайшим тегом $T_3 = [8,0]$.

После проведения данной операции теги пользователя будут иметь вид: $T_1 = [0,1]$, $T_2 = [2,0]$, $T_3 = [8,0]$, причем тег T_3 не обнулился.

На рисунке 2 стоит обратить внимание на объект "дети", который связан с объектом "театр" крайне слабо – 8 единиц притяжения. Такой результат можно было бы отнести к погрешности, однако в ходе анализа данных было выявлено, что в театрах проходили детские утренники и спектакли, таким образом, объекты "дети" и "театр" связаны закономерно.

В результате проделанной работы было разработано мобильное приложение для поиска культурных мероприятий города Бишкек с учетом предпочтений пользователя. Механизм формирования предпочтений основан на реализации кластерного анализа методом к-средних. Разработанный сервис запущен в пользование в августе 2016 г. с рабочим названием "CitySpy".

Литература

1. *Джонс М. Тим.* Рекомендательные системы. URL: https://www.ibm.com/developerworks/ru/library/os-recommender1/
2. Язык JavaScript. Электронный учебник. URL: https://learn.javascript.ru/json
3. *Кулаичев А.П.* Методы и средства комплексного анализа данных / А.П. Кулаичев. М.: ИНФРА-М, 2006.
4. Портал знаний. Кластеризация: метод к-средних. URL: http://statistica.ru/theory/klasterizatsiya-metod-k-srednikh/