

УДК 004.048

**Родионов Алексей Владимирович**

кандидат технических наук,
кафедра математических методов
и цифровых технологий,
Байкальский государственный университет,
г. Иркутск, Российская Федерация,
e-mail: avr-v@yandex.ru

**Юрков Георгий Юрьевич**

магистрант,
кафедра математических методов
и цифровых технологий,
Байкальский государственный университет,
г. Иркутск, Российская Федерация,
e-mail: gogganesko@gmail.com

ОБЗОР АЛГОРИТМОВ ПОСТРОЕНИЯ РЕКОМЕНДАТЕЛЬНОЙ СИСТЕМЫ ДЛЯ ВЫБОРА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ФИРМЫ «1С»

Аннотация. Данная статья посвящена обзору алгоритмов построения рекомендательной системы для выбора программных продуктов компании 1С. Рассмотрены такие методы, как фильтрация по содержимому (контент-фильтрация), а также коллаборативная фильтрация (совместная фильтрация). Далее было проведено исследование относительно того, как можно применить вышеприведенные методы для построения собственной рекомендательной системы выбора программного продукта, был выбран наиболее подходящий метод, проведено обоснование.

Ключевые слова: рекомендательные системы, контент-фильтрация, фильтрация по содержимому, коллаборативная фильтрация, совместная фильтрация.

Aleksei V. Rodionov

PhD in Technical Sciences,
Department of Mathematical Methods and Digital Technologies,
Baikal State University,
Irkutsk, Russian Federation,
e-mail: avr-v@yandex.ru

Georgiy Yu. Yurkov

Master's Degree Student,
Department of Mathematical Methods and Digital Technologies,
Baikal State University,
Irkutsk, Russian Federation,
e-mail: gogganesko@gmail.com

REVIEW OF ALGORITHMS FOR CONSTRUCTING A RECOMMENDER SYSTEM FOR SELECTING SOFTWARE OF THE COMPANY «1C»

Abstract. This article is devoted to an overview of the algorithms for building a recommender system for choosing 1C company software products. Such methods as filtering by content (content filtering), as well as collaborative filtering (joint filtering) are considered. Further, a study was conducted on how the above methods can be applied to build their own recommender system for choosing a software product, the most appropriate method was chosen, and a justification was carried out.

Keywords: recommender systems, content filtering, content filtering, collaborative filtering, collaborative filtering.

Введение

Доля предприятий, использующих информационные технологии для ведения своей основной деятельности с каждым годом, увеличивается и сейчас трудно представить компанию, не использующую какого-либо программного обеспечения. Одним из важных типов программных продуктов являются решения для автоматизации основных бизнес-процессов организации, связанных с бухгалтерским, управленческим, кадровым и другими видами учета. На сегодняшний день лидером по продажам такого рода ПО являются продукты компании 1С для автоматизации данных операций на предприятии.

Выбор конфигурации 1С может быть сложной задачей для фирмы, так как они предлагают множество различных решений, каждое из которых предназначено для решения определенных задач. Прежде чем внедрить какое-либо решение, перед предприятием встает задача выбора наиболее подходящего программного продукта, поскольку некоторый функционал на разных версиях конфигураций часто пересекается и может стать излишним для заказчика, что понесет за собой финансовые потери. Важно также учитывать возможность масштабирования выбранной конфигурации, ее совместимость с другими системами и возможность интеграции с существующими бизнес-процессами.

Одним из инструментов для «облегчения» процесса выбора необходимой конфигурации может стать рекомендательная система — комплекс сервисов и программ, который анализирует предпочтения пользователей и пытается предсказать, что может их заинтересовать [1]. Рекомендательная система может быть полезна для выбора конфигурации 1С, так как она позволит автоматизировать и оптимизировать процесс выбора на основе данных о предпочтениях пользователей и рекомендациях от других пользователей.

Рекомендательные системы имеют очень широкую область применения и назначение, поэтому их можно подразделять на группы по различным параметрам: предмету, цели, источнику рекомендации, пользовательскому контексту и степени персонализации. Например, цель систем, которые рекомендуют фильмы, книги, новости и другой контент — привлечь пользователя к потреблению новой информации на конкретной платформе. В этом случае предложение контента основано не только на контексте действий самого пользователя, но и на

новизне информации, оценках экспертов и популярности у других пользователей. Иначе работают системы, которые советуют купить товары повседневного спроса. Их задача — не только предложить новые продукты, исходя из контекста действий пользователей, но и напомнить о часто покупаемых товарах при наличии истории покупок.

Обзор алгоритмов построения рекомендательных систем

Существует несколько алгоритмов для построения рекомендательных систем, каждый из которых имеет свои преимущества и недостатки.

1. Коллаборативная фильтрация предполагает использование истории взаимодействий пользователей с продуктами с целью определения рейтинга продуктов для конкретного пользователя и работает на основе «подобия» между пользователями или продуктами [2,3]. Однако, этот метод имеет проблему т.н. холодного старта [4], когда для новых пользователей или продуктов нет достаточного количества исторических данных.

2. Алгоритмы контентной фильтрации (фильтрация по содержанию) основаны на характеристиках продуктов и профиле пользователя [5]. Используется информация о продуктах, такие как ключевые слова, категории и теги, чтобы создать профиль продукта. Затем сопоставляется профиль продукта с профилем пользователя, чтобы определить, какие продукты можно рекомендовать конкретному пользователю. Этот метод также может столкнуться с проблемой холодного старта, когда новые продукты еще не имеют достаточной описательной информации.

3. Анализ социальных сетей предполагает использование данных о взаимодействиях пользователей в социальных сетях, чтобы определить, какие продукты рекомендовать [6,7]. Этот метод может быть особенно полезен для продуктов, которые имеют социальную значимость или связаны с определенными сообществами.

4. Машинное обучение использует алгоритмы машинного обучения для создания моделей, которые могут предсказывать предпочтения пользователей и рекомендовать продукты [8-10]. Машинное обучение может использоваться как самостоятельный метод, так и в комбинации с другими методами рекомендательных систем.

5. Гибридные модели предполагают комбинацию разных методов, и позволяют улучшить точность и полезность рекомендаций [11]. Гибридные модели могут использовать как коллаборативную, так и контентную фильтрацию, а также другие методы, такие как анализ социальных сетей и машинное обучение.

С учетом специфики предметной области при разработке рекомендательной системы для построения рекомендательной системы выбора программного продукта компании 1С стоит более детально рассмотреть алгоритмы коллаборативной и контентной фильтрации. Алгоритмы анализа социальных сетей в данном случае мало применимы в связи с тем, что не позволяют учесть характеристики конфигурации 1С, такие как ее функции, возможности, цена и пр, кроме этого они будут ограничены только теми пользователями, которые уже используют какую-либо конфигурацию 1С. Таким образом, для выбора конфигурации

1С необходимо использовать алгоритмы, которые учитывают характеристики конфигурации, предпочтения и требования пользователей, а также опыт других пользователей, и анализ социальных сетей не является наиболее подходящим инструментом для этой задачи.

Фильтрация по содержанию

Фильтрация по содержанию выдает рекомендации на основе сходств признака предмета и пользовательских предпочтений. Применимо к выбору программного продукта, мы можем классифицировать покупателей, например по организационно-правовой форме (ООО, ИП, ЗАО и т.д.), либо по количеству пользователей будущего программного продукта и предлагать определенные варианты, основываясь на предыдущей статистике. В общем случае алгоритм может быть реализован следующим образом:

1. Необходимо собрать данные о характеристиках и свойствах различных конфигураций 1С (признаков), таких как функциональные возможности, совместимость с операционными системами, типы баз данных и т.д.
2. Определить метод оценки сходства между конфигурациями 1С на основе их характеристик и свойств. Это может быть достигнуто путем использования различных методов машинного обучения, таких как алгоритмы кластеризации или классификации.
3. Собрать данные о предпочтениях пользователя, которые могут включать в себя информацию о типе бизнеса, операционных системах, требуемых функциональных возможностях и т.д.
4. На основе собранных данных о характеристиках конфигураций и предпочтениях пользователя, создать профиль пользователя, который будет отражать его/ее потребности и предпочтения.
5. Используя метод оценки сходства между конфигурациями 1С и профилем пользователя, выбрать конфигурации, которые максимально соответствуют его/ее предпочтениям и потребностям.
6. Предоставить пользователю список рекомендованных конфигураций 1С, отсортированных по уровню сходства с профилем пользователя.
7. В случае если пользователь не нашел подходящей конфигурации, можно предоставить ему список альтернативных опций, основанных на более широких параметрах, таких как цена или тип функций.

При фильтрации по содержанию используются методы извлечения признаков, которые позволяют выделить наиболее значимые характеристики продукта для дальнейшего сравнения с другими продуктами и рекомендации пользователю. В частности, можно задействовать метод «Мешок слов» (Bag of Words), который использует частоту слов в описании продукта для определения его характеристик, при этом описание продукта рассматривается как коллекция слов, которые считаются независимыми друг от друга [12]. Затем, с помощью метода TF-IDF оценим важность каждого слова в описании продукта [12]. Это снижает значение слов, которые часто встречается в описаниях программных продуктов:

$$tf - i df_{ij} = tf_{ij} \times \log \left(\frac{N}{df_i} \right) \quad (1)$$

Здесь tf_{ij} – частота i в документах j , N – общее число документов, df_i – номер док., содержащего i .

Для сравнения сходства между соответствующим вектором программного продукта и предпочтениями пользователя будем использовать косинусный коэффициент, чтобы соотнести пользователя с программными продуктами:

$$\cos \theta = \frac{u \cdot v}{\|u\| \cdot \|v\|} \quad (2)$$

Здесь $\cos \theta$ – косинусный коэффициент, u, v – два вектора.

Использование косинусного коэффициента обосновывается рядом преимуществ: косинусный коэффициент учитывает длину векторов и нормализует значения, что позволяет устранить различия в длине векторов и сделать их сравнимыми; может быть использован для работы с разреженными данными, так как он не учитывает нулевые значения векторов, что может быть важно при работе с большим количеством продуктов или пользователей; имеет простой и быстрый алгоритм вычисления, что позволяет работать с большими объемами данных и быстро обрабатывать запросы на рекомендации.

Косинусный коэффициент может принимать значения от -1 до 1. Если коэффициент равен 1, то векторы полностью совпадают и имеют максимальную степень сходства. Если коэффициент равен -1, то векторы противоположны и имеют минимальную степень сходства.

Стоит отметить, что в данном алгоритме очень важна предварительная обработка данных. Необходимо просматривать весь словарь мешка слов и ограничивать его размер. Стоп-слова необходимо удалить. Помимо этого, необходимо удалить те слова в топе, которые не связаны с решением поставленной задачи.

Коллаборативная фильтрация

Фильтрация по содержимому имеет ограничение по качеству описания, предоставляемого контент-провайдером. Технически, существует ограничение на то, какие признаки могут быть извлечены из ограниченного объема контентной информации. Извлечение «контентной» информации у пользователя также является сложной задачей, поскольку в реальности легче судить о людях по тому, что они делают, чем по тому, что они говорят [10]. Коллаборативная фильтрация – алгоритм, основанный на анализе пользовательских данных, таких как история покупок, оценки, рейтинги и т.д., чтобы определить отношения между пользователями и предметами и дать рекомендации на основе этого анализа. Этот метод предполагает, что если два пользователя имеют схожие вкусы, то они, скорее всего, заинтересованы в одинаковых предметах. В общем виде алгоритм работы коллаборативной фильтрации можно представить так:

1. Необходимо собрать данные о предпочтениях пользователей относительно конфигураций 1С, например, данные о том, какие конфигурации были выбраны пользователями ранее, удовлетворены ли они ими и т.п.

2. Создать матрицу пользователь-конфигурация, в которой каждый элемент отражает оценку пользователя для конкретной конфигурации.

3. На основе предсказанных оценок для конфигураций создать ранжированный список рекомендуемых конфигураций для каждого пользователя.

4. Предоставить пользователю список рекомендованных конфигураций, отсортированных по высокой вероятности того, что пользователь оценит их высоко.

5. Обновлять матрицу пользователь-конфигурация и повторять процесс предсказания рекомендаций на основе новых данных о предпочтениях пользователей.

При выборе программного продукта можно рассматривать примеры организаций с разными организационно-правовыми формами, либо отталкиваться от основного направления деятельности компании и основываться на их отзывах о выбранной конфигурации. С другой стороны, вместо того чтобы находить сходства в клиентах, мы можем находить сходства в программных продуктах (предмет-предмет) и делать предложения на основе того, какие конфигурации уже используются. Что касается проблемы холодного старта, предложения вида предмет-предмет могут быть проще, когда информация о новом пользователе ограничена.

Одно из возможных применений совместной фильтрации — это нахождение ближайших соседей и использование их среднего значения в качестве нашего прогноза. Или же мы можем вычислить сходства между одним клиентом и всеми другими и использовать их для вычисления средневзвешенного значения возможных оценок. Т.к. оценка всегда субъективна, то нам понадобится нормализация голоса каждого клиента, прежде чем находить их сходство. Для вычисления сходства можно использовать коэффициент корреляции Пирсона:

$$u_{xy} = \frac{(r_{xj} - \bar{r}_x)(r_{yj} - \bar{r}_y)}{\sigma_x \sigma_y} \quad (3)$$

Здесь $(r_{xj} - \bar{r}_x)$ — оценка предмета j от пользователя x , $(r_{yj} - \bar{r}_y)$ — средний рейтинг пользователя y , $\sigma_x \sigma_y$ — стандартное отклонение рейтинга пользователя y . Данный коэффициент измеряет линейную зависимость между двумя переменными и находится в диапазоне от -1 до 1. Значение коэффициента равно 1, если две переменные полностью коррелируют, 0, если они не коррелируют, и -1, если они полностью обратно коррелируют.

Обсуждение результатов и выводы

Использование рекомендательных систем для продуктов 1С является эффективным способом помочь пользователям выбрать наиболее подходящую конфигурацию для их бизнеса. Алгоритмы коллаборативной фильтрации и фильтрации по содержанию могут быть использованы для этой цели, так как они предоставляют различные подходы для анализа пользовательских предпочтений и поведения. Кроме этого, совместное использование данных алгоритмов имеет свои преимущества и может дать более точные рекомендации, чем просто использование одного из этих методов.

Алгоритмы контентной фильтрации могут быть полезны для выбора конфигурации 1С, так как они учитывают характеристики самой конфигурации, такие как ее функции, возможности, цена и т.д. Это позволяет рекомендовать конфигурации, которые наиболее подходят к запросам пользователя, исходя из их требований и предпочтений.

С другой стороны, коллаборативная фильтрация основывается на анализе предпочтений других пользователей и учитывает схожесть между пользователями. Это позволяет предлагать конфигурации, которые были выбраны другими пользователями с похожими требованиями и предпочтениями, что может быть полезно для пользователей, которые еще не имеют большого опыта работы с 1С и не знают, какую конфигурацию выбрать.

В качестве основы построения такой системы можно взять существующие наработки, которые имеются на предприятии, например сценарий входящего телефонного разговора с клиентом. Из него мы можем подчеркнуть основные признаки «интеллектуального помощника».

Некоторые наводящие вопросы, которые обычно задаются при принятии входящего звонка от клиента:

1. Какой вид деятельности предприятия? Является этот вид отраслевым направлением (автосервис, салон красоты, общественное питание и т.д.) или только продажа товаров?
2. Какие задачи должна решать программа, которую вы хотите купить?
3. Имеется ли лицензионное программное обеспечение 1С, которым они пользуются в данный момент?
4. Сколько человек будет работать в программе?
5. Ведение бухгалтерского учета планируется вести (или уже ведется) самостоятельно или аутсорсинг?
6. Продажи только розничные или есть оптовые (есть необходимость выставлять счета клиентам)?

Уже по такому списку вопросов можно определить нужно ли клиенту комплексное решение, например конфигурация «Управление торговлей», либо узкоспециализированное, такое как «1С: Садовод». Помимо этого, мы можем определить количество лицензий, нужны ли дополнительные сервисы и т.д.

Таким образом мы можем составить некоторые признаки (организационно-правовая форма предприятия – клиента, количество будущих пользователей программы, основная задача учета и т.д.), необходимые нам для дальнейшего исследования и решить проблему холодного старта, внеся уже накопленную информацию со звонков в систему. Помимо этого, мы можем собирать отзывы клиентов о качестве подбора программных продуктов, значительно повышая со временем качество будущих рекомендаций.

Заключение

Алгоритмы рекомендательных систем, основанные на математических методах, являются основой построения интеллектуальных помощников. В данной статье был проведен обзор некоторых из них, показано, что использование ал-

горитмов контентной и коллаборативной фильтрации может помочь пользователям выбрать наиболее подходящую конфигурацию 1С, учитывая их требования и предпочтения, а также опыт других пользователей. В целом, использование рекомендательных систем для продуктов 1С может существенно упростить процесс выбора конфигурации для пользователей и помочь им выбрать наиболее подходящую опцию.

Список использованной литературы

1. Рекомендательные системы: что это, зачем нужно и как создать собственную в Yandex Cloud. – Москва, 2022. – URL: <https://cloud.yandex.ru/blog/posts/2022/05/recommendation-system-instruction> (дата обращения: 20.04.2022).
2. Оболенский Д. М. Обзор современных методов построения рекомендательных систем на основе коллаборативной фильтрации / Д. М. Оболенский, В. И. Шевченко // Мир компьютерных технологий : Сборник статей всероссийской научно-технической конференции студентов, аспирантов и молодых ученых, Севастополь, 06–10 апреля 2020 года / Науч. редактор Е.Н. Мащенко. – Севастополь: Федеральное государственное автономное образовательное учреждение высшего образования «Севастопольский государственный университет», 2020. – С. 97-102. – EDN JEDZEN.
3. Музалевский Д. С. Пример проектного подхода к обучению в области обработки больших данных на основе построения рекомендательной системы с применением методов коллаборативной фильтрации с использованием Apache Spark и Python / Д. С. Музалевский, Ю. Е. Гапанюк // Наука и образование: научное издание МГТУ им. Н.Э. Баумана. – 2016. – № 7. – С. 251-259. – EDN XEJZMV.
4. Проблема «холодного старта» / Р. З. Омаров, А. В. Востротина, А. Д. Ли [и др.]. Текст : непосредственный // Молодой ученый. – 2019. – № 26 (264). – С. 85-88. – URL: <https://moluch.ru/archive/264/61285/> (дата обращения: 28.04.2023).
5. Ботов Д. С. Разработка рекомендательной системы для музыкального сервиса на основе методов контентной и коллаборативной фильтрации / Д. С. Ботов, Н. В. Меньшикова, И. В. Портнов // Информационные технологии и системы : труды Шестой Международной научной конференции Научное электронное издание, Банное, 01–05 марта 2017 года. – Банное: Челябинский государственный университет, 2017. – С. 37-46. – EDN ZSCPRH.
6. Авхадеев Б. Р. Разработка рекомендательной системы на основе данных из профиля социальной сети «ВКонтакте» / Б. Р. Авхадеев, Л. И. Воронова, Е. П. Охупкина // Вестник Нижневартковского государственного университета. – 2014. – № 3. – С. 68-76. – EDN SPLNGN.
7. Monastirev V. V. Recommendation System Based on User Actions in the Social Network / V. V. Monastirev, P. D. Drobintsev // Proceedings of the Institute for System Programming of the RAS. – 2020. – Vol. 32, No. 3. – P. 101-108. – DOI 10.15514/ISPRAS-2020-32(3)-9. – EDN RVNFWM.

8. Мосалов О. П. Применение методов кластеризации и машинного обучения для построения рекомендательной системы для определения актуальности научных публикаций / О. П. Мосалов, И. А. Иванов, М. А. Першин // Информационно-технологический вестник. – 2021. – № 4(30). – С. 89-102. – EDN NSJIKF.
9. Применение методов машинного обучения для построения рекомендательной системы отбора анкет абитуриентов / В. В. Малышев, С. С. Сливкин, В. С. Рукавишников, Е. В. Базаркин // Научный вестник Новосибирского государственного технического университета. – 2017. – № 2(67). – С. 109-119. – DOI 10.17212/1814-1196-2017-2-109-119. – EDN ZEHUAX.
10. Машинное обучение: рекомендательные системы // VC.ru URL: <https://vc.ru/ml/132779-mashinnoe-obuchenie-rekomendatelnye-sistemy> (дата обращения: 09.04.2023).
11. Заграновская А. В. Проектирование гибридных рекомендательных систем / А. В. Заграновская, Д. Ю. Митюра // Международный научный журнал. – 2019. – № 5. – С. 88-94. – DOI 10.34286/1995-4638-2019-68-5-88-94. – EDN CGBSVS.
12. Prasoon Singh. Fundamentals of Bag of Words and TF-IDF. – 2019. – URL: <https://medium.com/analytics-vidhya/fundamentals-of-bag-of-words-and-tf-idf-9846d301ff22> (дата обращения: 20.04.2022).