

В.Г. Кузьмин

ТЕНДЕНЦИИ В ПРОЕКТИРОВАНИИ ФУНКЦИОНАЛЬНОЙ АРХИТЕКТУРЫ РЕКОМЕНДАТЕЛЬНЫХ СИСТЕМ

Дано определение, проанализирована цель разработки рекомендательных систем, выделены основные этапы функциональной архитектуры рекомендательной системы. Приведена архитектура действующей рекомендательной системы. Представлены примеры других рекомендательных систем. Установлено, что основные этапы функциональной архитектуры в целом остаются неизменными для всех рекомендательных систем.

Ключевые слова: функциональная архитектура, рекомендательная система.

V.G. Kuzmin

TRENDS IN DESIGNING THE FUNCTIONAL ARCHITECTURE OF RECOMMENDER SYSTEMS

This article provides a definition for recommender systems, analyzes reasons for designing recommender systems, highlights and explains the main stages of functional architecture of recommender systems. An example of architecture of an already implemented recommender system is examined. Examples of other recommender systems are given. The conclusion is made about the presented functional architecture being consistent throughout all recommender systems.

Keywords: functional architecture, recommender system.

Вводные замечания

Актуальность данной работы обуславливается развитием информационно-коммуникационных технологий, технологий хранения информации и взрывного роста доступной информации, что привело к невозможности изучения материала «в лоб» и потребности автоматизации процесса извлечения информации.

Одним из способов автоматизации процесса извлечения информации является использование рекомендательных систем. Цель данной работы – провести анализ современных тенденций в проектировании функциональной архитектуры рекомендательных систем.

Определение рекомендательных систем

Рекомендательная система согласно [7] – это совокупность программных средств, которые предоставляют осмысленные предложения пользователям, опираясь на их профиль, предпочтения и действия, зафиксированные во время взаимодействия пользователя с интерфейсом программы или сайта. Рекомендательные системы обычно персонализированы и могут предлагать пользователям информацию различного вида, например, товары, рекомендуемые к приобретению, пользователи со схожими взглядами, книги или статьи, рекомендуемые к прочтению. Рекомендательные системы ставят перед собой цель помочь пользователям более эффективно взаимодействовать с интерфейсом программы или сайта, выделяя именно то, что им в первую очередь необходимо. В качестве примера можно привести сайт YouTube, главная страница которого изображена на Рисунках 1 и 2,

Кузьмин Виталий Геннадьевич

бакалавр Российского нового университета, Москва. Сфера научных интересов: рекомендательные системы.

Электронный адрес: kuzminvg@stud.rosnou.ru

в частности результаты рекомендательной системы. Как видно, рекомендательная система не является для пользователя самоцелью, а интегрирована в сайт и делает взаимодействие с ним более эффективным. Рекомендательная система может работать только при наличии существующего профиля пользователя. Если войти на сайт в качестве анонимного гостя, то сайт отобразит рекомендации, опираясь на глобальную статистику, полученную от всех пользователей.

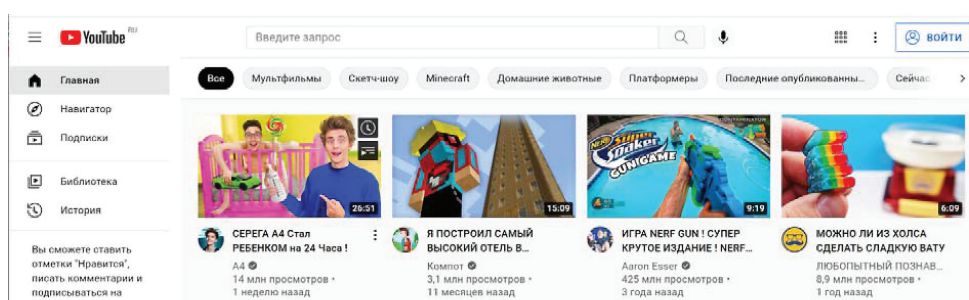


Рисунок 1. Результаты рекомендательной системы сайта YouTube для анонимных пользователей

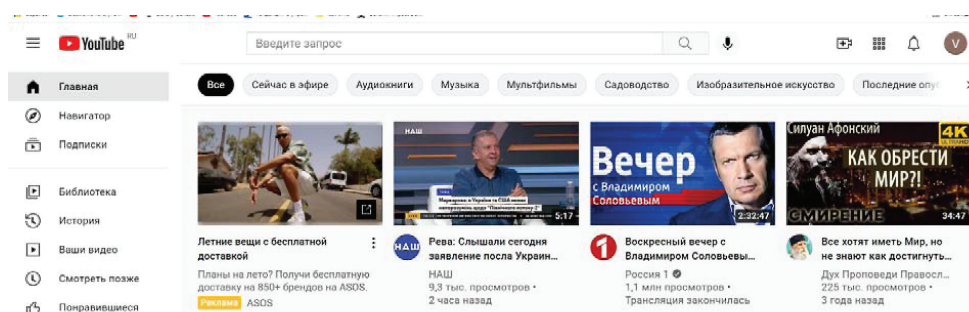


Рисунок 2. Результаты рекомендательной системы сайта YouTube для конкретного пользователя

Рекомендательные системы согласно [7] относятся к семье информационных систем и имеют перед собой задачу предугадать оценку, которую конкретный объект получил бы от конкретного пользователя. Поэтому обычно методология, которой следуют рекомендательные системы, заключается в том, чтобы выстроить три модели – модель пользователя, модель оценок и модель оцениваемого объекта, найти корреляцию между ними и, опираясь на полученную корреляцию, выдвинуть соответствующие рекомендации.

В данном понятии объект – это песня, фильм, товар, пользователь (в случае рекомендации друзей), то, что потребляют пользователи системы рекомендаций, то, что им нуж-

Тенденции в проектировании функциональной архитектуры рекомендательных систем

но рекомендовать. Пользователь – это человек, зарегистрированный в системе, он может покупать, слушать, смотреть, оценивать объекты и пользоваться сервисом рекомендации. Оценка – это результат, выдаваемый специальным алгоритмом оценивания для какого-либо объекта с целью последующего его использования в качестве одного из критериев соответствия объекта интересам пользователя. Рекомендация – это объект или несколько объектов, которые система рекомендации выдает пользователю. Ранжированный список рекомендаций является конечным результатом рекомендательной системы.

Источник [5] дает несколько другое определение, которое не противоречит вышеописанному, но дополняет его. Согласно [5] рекомендательные системы – это научная дисциплина, раздел теории искусственного интеллекта и теории информационного поиска, изучающий методы анализа больших объемов данных, в особенности данных определенных объектов и данных пользователей с целью предоставления актуальных рекомендаций. В техническом плане проблема рекомендательных систем заключается в создании математической модели или функции, которая сможет предугадать, насколько конкретный объект подойдет конкретному пользователю. Тогда, предположив, что выходом математической модели или функции является степень соответствия пользователя и объекта, можно сказать, что задача рекомендательной системы в процессе работы – определение для каждого конкретного пользователя такого конкретного объекта, чтобы выход созданной математической модели или функции принимал максимальное значение.

Функциональная архитектура рекомендательных систем

Функциональная архитектура – это множество функций и их подфункций, упорядоченных между собой, и определяющее череду преобразований потока входной информации, выполняемых системой ради достижения поставленной цели [4]. Рекомендательная система имеет следующую функциональную архитектуру [7] (Рисунок 3).

Рекомендательная система состоит из циклически функционирующих процедур, которые можно поделить на 4 шага.

Шаг 1. Сбор данных.

Сбор данных напрямую зависит от модели данных, используемой в приложении. Данные обычно извлекаются из проектной документации приложения и контекстной информации, с которой приложение работает, с целью дальнейшей их обработки. Модель данных обычно зависит от предметной области, в которой приложение будет функционировать, и выбранных способов хранения и представления данных, например, с помощью реляционных баз данных. На этом этапе происходит наполнение моделей пользователя и объектов, о которых было рассказано выше.

Шаг 2. Фильтрация данных с применением методов, разработанных для рекомендательных систем.

Выбор методов фильтрации зависит от типа данных, с которыми рекомендательная система будет взаимодействовать, а также от типа рекомендаций, которые она будет формировать. На данном этапе происходит наполнение моделей оценок. В целом методы фильтрации данных в рекомендательных системах можно разделить на четыре категории в зависимости от используемого типа данных и вычислительных алгоритмов:

- коллаборативная фильтрация;
- фильтрация, основанная на содержании;
- фильтрация, основанная на знаниях;
- гибридная фильтрация.

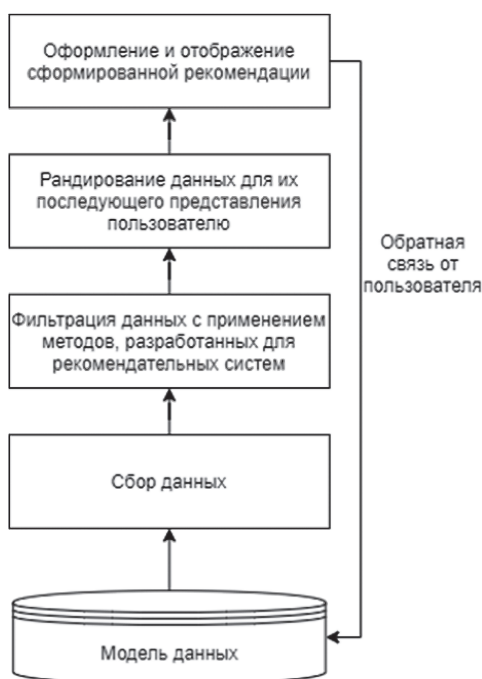


Рисунок 3. Функциональная архитектура рекомендательной системы

Шаг 3. Ранжирование данных для последующего их представления пользователю.

Сбор данных и их обработка через рекомендательную систему генерирует ряд рекомендаций, которые соответствуют правилам, сформированным системой или пользователем, например, исключение данных, находящихся в черном списке пользователя. Это шаг ранжирования полученных рекомендаций. Ранжирование происходит исходя из предпочтений каждого пользователя. На данном этапе происходит корреляция между результатами предыдущих этапов. Рекомендация считается успешной, когда ее положение в ранжированном списке рекомендаций соответствует степени заинтересованности пользователя в ней; например, вторая по счету рекомендация в списке рекомендаций является для пользователя менее предпочтительной по сравнению с первой, но более предпочтительной по сравнению с третьей, если эта рекомендация действительно приемлема для пользователя.

Шаг 4. Оформление и отображение сформированной рекомендации.

Последний этап в функциональном цикле рекомендательной системы – преобразование списка рекомендаций, полученного на предыдущем этапе в формат, который впоследствии будет передан модулю системы, ответственной за отображение результатов работы системы, пользовательского интерфейса и др.

Предполагается, что функциональный цикл рекомендательной системы непрерывен, и на протяжении функционирования системы идет сбор данных о пользовательских действиях, которые свидетельствуют о реакции пользователя и позволяют системе адаптироваться под предпочтения пользователя, которые могут меняться со временем. В функциональной архитектуре, представленной выше, предполагается, что пользователь самостоятельно будет оценивать рекомендации, выданные ему системой, поэтому сбор данных о пользовательских действиях происходит только в конце прохода по циклу. Источник

Тенденции в проектировании функциональной архитектуры рекомендательных систем

[6] утверждает, что, хотя это потребует дополнительных усилий от пользователя, данный подход считается более надежным, а также делает процесс получения обратной связи от пользователя и ее влияние на выдаваемые рекомендации более прозрачными. Но есть вариант косвенного получения обратной связи на протяжении всего прохода по циклу. В этом случае система автоматически пытается понять предпочтения пользователя на основании действий, не связанных непосредственно с оцениванием рекомендаций, например, время нахождения на странице сайта или переходы по ссылкам. Такой подход не предполагает дополнительных усилий от пользователя, поэтому является менее надежным. Источники [1; 6] утверждают, что такой подход в некоторых ситуациях может быть даже предпочтительнее, так как пользователи вполне могут оценивать рекомендации исходя не из своих предпочтений, а, например, из того, что приемлемо в обществе. Источник [5] предлагает использовать оба подхода одновременно, получая обратную связь от пользователя согласно первому подходу и используя второй подход для выяснения естественности полученной обратной связи.

Например, рекомендательный модуль системы Alibaba Cloud согласно [2] имеет следующую архитектуру (см. Рисунок 4).

PAI-based Technical Architecture for a Recommender System

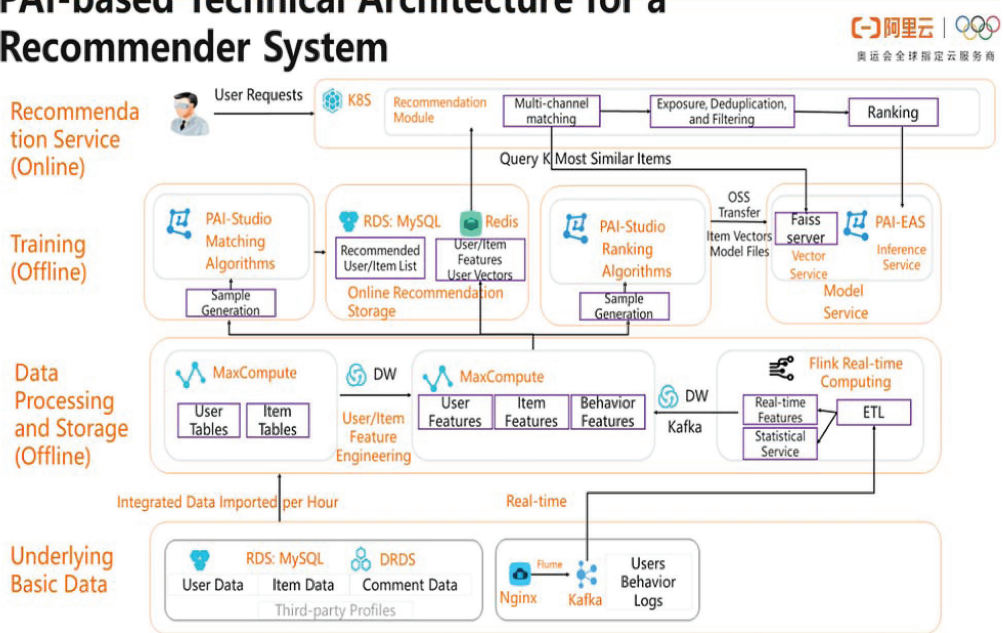


Рисунок 4. Архитектура рекомендательного модуля системы Alibaba Cloud

Архитектура рекомендательного модуля системы Alibaba Cloud также имеет 4 этапа. *Этап 1.* Сбор данных.

Данные в архитектуре разделяются на два вида – статические интегрированные данные, импортируемые в систему из внешних источников раз в час, и данные реального времени, которые следят за действиями пользователей в системе. Данные, импортируемые

в систему, – это данные о самих пользователях, оцениваемых объектах и комментариях, оставляемых пользователями. Всё это происходит офлайн без непосредственного участия пользователей. Далее управление передается следующему этапу.

Этап 2. Обработка и хранение данных.

Импортируемые данные разбиваются по таблицам в реляционной базе данных, а данные реального времени загружаются с помощью процессов ETL в систему Apache Kafka – платформу, специализирующуюся на обработке и хранении данных реального времени. Далее из полученных данных собираются и заполняются модели оценок для анализируемых сущностей – пользователя, объектов, действий пользователя. Итогом данного этапа является формирование оценок анализируемых сущностей, которые впоследствии система будет сравнивать между собой на следующем этапе. Данный этап также происходит офлайн без непосредственного участия пользователей.

Этап 3. Обучение.

Осуществляется сравнение полученных оценок между собой с целью получения возможных рекомендаций, а также составление возможных ранжированных списков рекомендации без привязки к какому-либо конкретному пользователю. Алгоритмы, по которым сравниваются полученные оценки и ранжируются полученные рекомендации, встроены в общую платформу и выбираются исходя из требований. Данный этап происходит офлайн без непосредственного участия пользователей.

Этап 4. Рекомендательный сервис.

На этом этапе уже собраны и обработаны модели пользователей, модели объектов, модели оценок. Было проведено сравнение между данными этих моделей и составлены все возможные рекомендации, которые впоследствии были ранжированы согласно всем возможным правилам. В итоге получена совокупность пар пользователь, ранжированный список рекомендаций. Данный этап происходит онлайн, как только поступает запрос от пользователя, главная задача этапа – отобрать из всех возможных пар наиболее подходящие на данный момент. Эти списки затем могут передаваться на предыдущий этап на хранение с целью более быстрого извлечения в будущем.

Обратная связь, согласно данной архитектуре, извлекается только косвенно с помощью логов, фиксирующих действия пользователя на протяжении работы модуля, а затем эти действия компонуются в отдельную модель и впоследствии объединяются с моделью пользователя. Прямая обратная связь в данной архитектуре не используется.

Описанная выше архитектура тоже имеет 4 этапа, однако отличается от рассмотренной источником [7] функциональной архитектуры следующим.

1. Этап сбора данных разбит на 2 подэтапа – непосредственное получение данных из источников и их последующая обработка и заполнение моделей.

2. Этапы фильтрации и ранжирования объединены в один, который был назван обучением. При этом в [7] предполагается, что этапы ранжирования и фильтрации будут пройдены с привязкой на конкретного пользователя, и следующий этап будет отвечать только за их отображение. Здесь этого не происходит, и данный функционал был перенесен на следующий этап.

3. Последний этап отвечает не только за отображение результатов, но и за подбор необходимого списка рекомендаций из тех, которых были получены на предыдущем этапе. В [7] такой функционал для последнего этапа не предусмотрен.

4. Присутствует функционал передачи результатов последнего этапа на предыдущий этап с целью их более быстрого извлечения в будущем. Очевидно, что данное действие

предназначено лишь для ускорения скорости работы системы и не является серьезным отклонением от архитектуры, предложенной в [7].

Схожую архитектуру можно найти, например, в источниках [3; 8; 9], где описывается архитектура предлагаемых рекомендательных модулей. Вышеописанная функциональная архитектура может быть дополнена или изменена, ее этапы могут быть реализованы по-разному, но тем не менее она является основой, на которую опираются все рекомендательные системы.

Заключение

В ходе обзора современных практик проектирования рекомендательных систем было дано определение рекомендательным системам и выявлены основные этапы функциональной архитектуры рекомендательной системы. Было установлено, что функциональная архитектура различных рекомендательных систем либо не отличается друг от друга, либо отличается слабо; возможные отличия систем были представлены на примере. Установлено, что процесс работы рекомендательной системы регулируется самостоятельно, так как результаты последнего этапа предыдущего цикла работы используются при прохождении последующего цикла работы.

Литература

1. Addressing uncertainty in implicit preferences [Электронный ресурс]. URL: <https://www.macs.hw.ac.uk/~dwcorne/ACMRecSys07/p97-gadanho.pdf>
2. Alibaba Cloud Document Center [Электронный ресурс]. URL: <https://www.alibabacloud.com/help?spm=a2c5v.8452849.6791778070.dnavdocumentation0.185969730yHYWB>
3. An Architecture for Developing Educational Recommender Systems [Электронный ресурс]. URL:
4. https://rcs.cic.ipn.mx/2015_106/An%20Architecture%20for%20Developing%20Educational%20Recommender%20Systems.pdf
5. Functional Architecture (glossary) [Электронный ресурс]. URL:
6. [https://www.sebokwiki.org/wiki/Functional_Architecture_\(glossary\)](https://www.sebokwiki.org/wiki/Functional_Architecture_(glossary))
7. Gorakala S.K. (2016) Building_Recommendation_Engines. Packt Publishing, 452 p.
8. Learning with personalized recommender systems: A psychological view [Электронный ресурс]. URL: <https://www.sciencedirect.com/science/article/abs/pii/S0747563211001956>
9. Recommender Systems Review: Types, Techniques and Applications [Электронный ресурс]. URL: <http://www2.cs.ucy.ac.cy/~george/files/IGI15.pdf>
10. System Architectures for Personalization and Recommendation [Электронный ресурс]. URL: <https://netflixtechblog.com/system-architectures-for-personalization-and-recommendation-e081aa94b5d8>
11. The Architecture and Datasets of Docear's Research Paper Recommender System [Электронный ресурс]. URL:
12. <https://isg.beel.org/pubs/The%20Architecture%20and%20Datasets%20of%20Docear's%20Research%20Paper%20Recommender%20System.pdf>

References

1. Addressing uncertainty in implicit preferences. Available at:
2. <https://www.macs.hw.ac.uk/~dwcorne/ACMRecSys07/p97-gadanho.pdf>
3. Alibaba Cloud Document Center. Available at:
4. <https://www.alibabacloud.com/help?spm=a2c5v.8452849.6791778070.dnavdocumentati on0.185969730yHYWB>
5. An Architecture for Developing Educational Recommender Systems. Available at: https://rcs.cic.ipn.mx/2015_106/An%20Architecture%20for%20Developing%20Educational%20Recommender%20Systems.pdf
6. Functional Architecture (glossary). Available at:
7. [https://www.sebokwiki.org/wiki/Functional_Architecture_\(glossary\)](https://www.sebokwiki.org/wiki/Functional_Architecture_(glossary))
8. Gorakala S.K. (2016) Building_Recommendation_Engines. Packt Publishing, 452 p.
9. Learning with personalized recommender systems: A psychological view]. Available at: <https://www.sciencedirect.com/science/article/abs/pii/S0747563211001956>
10. Recommender Systems Review: Types, Techniques and Applications. Available at: <http://www2.cs.ucy.ac.cy/~george/files/IGI15.pdf>
11. System Architectures for Personalization and Recommendation. Available at: <https://netflixtechblog.com/system-architectures-for-personalization-and-recommendation-e081aa94b5d8>
12. The Architecture and Datasets of Docear's Research Paper Recommender System. Available at: <https://isg.beel.org/pubs/The%20Architecture%20and%20Datasets%20of%20Docear's%20Research%20Paper%20Recommender%20System.pdf>