Санкт-Петербургский государственный университет

Программная инженерия

Сергеев Егор Федорович

Система профилирования пользователей на основе поведенческих логов

Курсовая работа

Научный руководитель:
к. т. н., доцент Брыксин Т. А.

Консультант:
Основатель “Graphica.ai” Брыксин М. А.

Санкт-Петербург
2020

Оглавление

Введение

В современном мире благодаря развитию веб-технологий и распространенности различных устройств наблюдается стремительный рост популярности веб-сервисов. Люди имеют доступ к многочисленным источникам информации и постоянно находятся в информационном потоке. Из-за этого веб-приложения вынуждены всеми силами бороться за удержание внимания пользователя. В частности, стоит отметить медиа веб-сервисы — те, задачей которых является предоставление доступа к информации в различном виде: текстовом, визуальном и звуковом. При этом количество информации в сети интернет стремительно растет, и ручной отбор медиаматериалов не представляется возможным. Поэтому главной задачей медиа веб-сервисов является подбор требуемого материала для каждого пользователя. От качества решения этой задачи напрямую зависит успех сервиса. Примером, иллюстрирующим важность приведенной выше задачи, можно привести соревнование, организованное компанией Netflix в 2007 году [23]. Призовой фонд составил $1 млн, а целью являлось улучшение точности алгоритма рекомендательной системы всего на 10%. Таких результатов удалось достичь лишь на третий год проведения соревнования. Это говорит о том, что крупные компании готовы тратить большое количество ресурсов на улучшение алгоритмов подбора материалов.

Однако для того, чтобы точно подбирать материалы для пользователей, необходимо иметь представление о них. Для достижения этой цели важно разделить пользователей на некоторые группы со схожими атрибутами. Классические методы сегментирования, применяющиеся в индустрии, наподобии эмпатических интервью [14] и маркетинговых исследований [25], практически не применимы в сфере подбора медиаматериалов из-за высокой ресурсоемкости, небольшой выборки пользователей и плохой масштабируемости. Кроме того, гораздо лучше человека характеризует его поведение и особенно взаимодействие непосредственно с самими медиафайлами. Информацию о шаблонах поведения можно получить, анализируя логи взаимодействия пользователя

с сервисом — поведенческие логи. Они представляют собой записи о совершенных определенным пользователем действиях с материалами веб-сервиса. Сами по себе они слабо структурированы и плохо понятны человеку. Возникает необходимость в системе, способной строить типовые интерпретируемые профили пользователей, основываясь на их поведении, а также относить человека к одной из уже определённых групп. Помимо этого важным аспектом является осознание семантики этого поведения — свойств объектов, с которыми взаимодействует пользователь. Это позволит делать выводы относительно базовых причин совершения человеком тех или иных поступков.

Для того, чтобы оценить значимость интерпретируемых профилей, стоит упомянуть контекст их использования. В рамках этой работы они будут строиться для пользователей веб-сервиса graphica.ai, который предлагает людям творческих профессий возможность найти визуально эстетичные изображения для вдохновения. Из-за того, что понятия “вдохновение” и “эстетика” очень субъективны, этому сервису крайне необходимо иметь информацию о типовых профилях пользователей для создания качественной рекомендательной системы и продвижения. Описанные выше ограничения в лице трудно формализуемых понятий и являются основной причиной интерпретируемости профилей: на данный момент только человек способен осознать, а следовательно, и сделать выводы относительно искусства, которое распространяет graphica.ai. Это накладывает дополнительные ограничения на количество построенных профилей: человек физически не способен просмотреть профиль каждого пользователя. Поэтому необходимо сегментировать пользователей с помощью алгоритмов кластеризации и строить интерпретируемые типовые профили для каждого пользовательского сегмента.

Постановка задачи

Целью данной работы является реализация системы построения типовых профилей пользователей веб-сервиса на основе поведенческих

логов и её интеграция существующую систему graphica.ai.

В ходе работы для достижения описанной цели были поставлены следующие задачи.

- Провести анализ существующих решений и алгоритмов
- Разработать систему построения типовых пользовательских профилей и клиент для тестирования системы
- Провести оценку качества построения типовых профилей пользователей с помощью привлечения экспертов в области дизайна и маркетинга
- Провести интеграцию в архитектуру существующей системы graphica.ai

1. Обзор

Наиболее распространенным решением для описания сегментов пользователей является внедрение систем веб аналитики в клиент веб-сервиса, как правило представляющим собой вебсайт. Яркими представителями таких систем являются Яндекс.Метрика [33], Google Analytics [7], Mixpanel [16] и другие. Их преимущество заключается в простоте внедрения: достаточно разместить небольшой HTML-код на страницах сайта [32]. Это позволит системе веб аналитики собирать данные о переходах пользователей по этим страницам. Интерфейс таких систем предоставляет доступ к агрегированной информации о сегментах пользователей — типовым профилям. В числе характеристик, которые можно получить таким образом, находятся географические (Рис. 1) и демографические категории (Рис. 2), а также сферы интересов пользователей (Рис. 3), полученные исходя из вебсайтов, которые они посещали. Кроме этого имеется возможность оценить поведение сегментов, основанное на посещенных страницах вебсайта и кликах по элементам на этих страницах. Главный недостаток таких систем веб аналитики кроется в их универсальности: они не имеют возможности использовать базу данных сервиса, в который внедряются. Поэтому они не могут извлечь метаинформацию об элементах HTML страниц, с которыми взаимодействовали пользователи. Таким образом, системы веб аналитики не могут оценить сематику поведения людей. Напомню, что под семантикой поведения подразумеваются свойства объектов, с которыми взаимодействовали пользователи.

Страна	Источники трафика
	Пользователи
	809 % от общего количества: 100,00 % (809)
1. Russia	**605** (74,42 %)
2. Ukraine	**102** (12,55 %)
3. Kazakhstan	**38** (4,67 %)
4. Belarus	**25** (3,08 %)
5. Uzbekistan	**7** (0,86 %)
6. Germany	**5** (0,62 %)
7. Latvia	**4** (0,49 %)
8. United States	**4** (0,49 %)
9. Kyrgyzstan	**3** (0,37 %)
10. Moldova	**3** (0,37 %)

Рис. 1: Сегментирование пользователей по их местоположению [7]

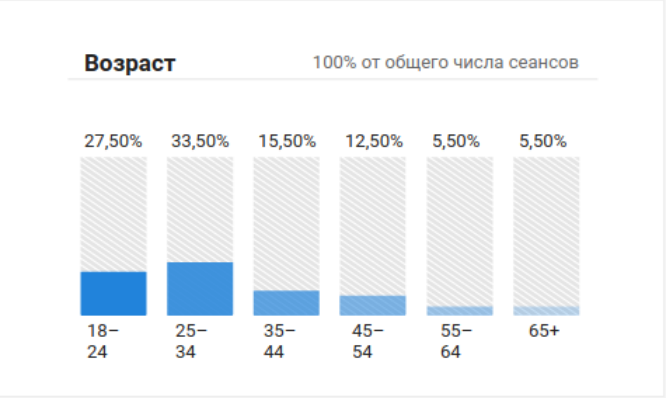

Рис. 2: Сегментирование пользователей по их возрасту [7]

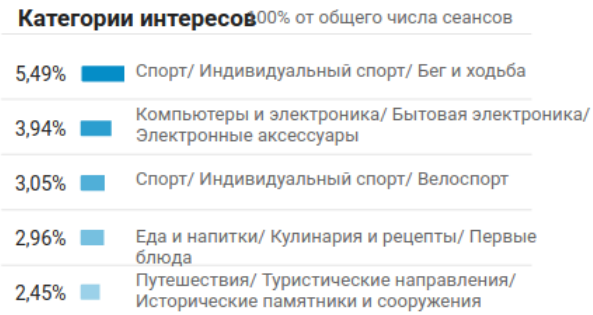

Рис. 3: Сегментирование пользователей по их интересам [7]

Стоит отметить и более низкоуровневые решения в области веб аналитики, такие как Google Tag Manager [8], Facebook Pixel [5] и Segment [21]. Они позволяют собирать данные о любых действиях пользователя. Это осуществляется благодаря возможности пометить элементы HTML страницы специальными метками, взаимодействие с которыми собирается автоматически. Однако для регистрации этого взаимодействия необходимо пометить каждый интересующий элемент. Этот процесс может быть весьма трудоемким для крупных веб-сервисов. Кроме того, низкоуровневые решения представляют собой только инструмент для сбора поведенческих логов — записей, характеризующих события, которые произошли в процессе взаимодействия пользователя с клиентом вебсервиса. Таким образом, они не выполняют главную задачу, поставленную в рамках данной работы: построение типовых профилей.

Помимо систем веб аналитики стоит рассмотреть и ряд подзадач построения типовых профилей и подходов к их решению, которые могут быть использованы при разработке системы профилирования. В первую очередь, такая система предполагает сбор поведенческих логов для их дальнейшей обработки. Тут необходимо определиться с форматом записей и видами регистрируемых событий, таких как клик по изображению или переход на страницу вебсайта. Далее необходимо хранилище поведенческих логов, способное сохранять поток поведенческих логов. Наконец, данные из хранилища должны агрегироваться для формирования типовых профилей — набора атрибутов, соответствующих сегменту пользователей. Ниже в каждом разделе представлен обзор подходов, реализующих эти элементы системы.

1.1. Поведенческие логи

Поведенческие логи представляют собой упорядоченный набор записей о событиях, которые возникли в результате взаимодействия пользователя с сервисом и зарегистрировались на его клиенте. События можно разделить на два типа: явные и неявные. Первые содержат в себе явную оценку медиаматериалов пользователем, например выставление

рейтинга или механика “мне нравится” в социальных сетях. Вторые выражают отношение пользователя лишь косвенно. Такими действиями являются посещение страниц, наведение мышью на объект и другие. В сервисе graphica.ai на момент проведения исследования не существует механик, позволяющих получать события явного типа, поэтому решено было логировать только неявные действия.

Классическим подходом при сборе поведенческих логов является clickstream [19] — поток всех посещенных пользователем URL страниц и формирование упорядоченной по времени последовательности переходов для каждого пользователя. Главным преимуществом такого подхода является простота сбора логов и их дальнейшей кластеризации. Это наглядно продемонстрировано при построении рекомендательных систем [15] и аналитики в реальном времени [10]. Помимо этого существует и более продвинутый подход [1]. В нем кроме переходов по URL страницам собирается также и наличие любой активности на них, что позволяет получить информацию о продолжительности времени фокусировки внимания пользователя на медиаматериалах.

Что касается непосредственно самого формата логов, стоит подробнее рассмотреть системы веб аналитики. Яндекс.Метрика [33] позволяет обращаться к “счетчику” — регистратору событий, привязанному к HTML странице или элементу на ней. С помощью этого обращения можно получить доступ к списку всех событий, которые зафиксировал счетчик. В данном контексте список событий представляет собой набор поведенческих логов, каждый из которых содержит множество значений [34]. Несмотря на то, что такой список атрибутов избыточен в рамках поставленной задачи, можно выделить несколько представляющих интерес атрибутов.

- Время возникновения события
- Идентификатор счетчика
- Адрес и заголовок страницы, на которой было зафиксировано событие

1.2. Хранение логов

Можно выделить два подхода организации хранилища логов: OLTP (Online Transaction Processing) [28] и OLAP (Online Analytical Processing) [27].

OLTP системы оперируют транзакциями — атомарными изменениями состояния базы данных. Ключевыми качествами таких систем являются консистентность и актуальность хранимых данных, а также скорость ответа на запросы, в случае обращения к единичным записям в базе. В контексте поставленной задачи, эти системы могут использоваться в случае, если система построения профилей нацелена на быстрое и надежное сохранение поведенческих логов, а не на скорость их обработки.

OLAP системы направлены на создание аналитических отчетов. При этом главной характеристикой является скорость выполнения запросов, которые их формируют. В контексте поставленной задачи такой подход может быть использован, если от системы построения профилей требуется высокая производительность при вычислении атрибутов, содержащихся в профилях.

1.3. Обработка логов

Задача обработки поведенческих логов заключается в вычислении значений атрибутов, соответствующим пользователям. Атрибуты можно разделить на два типа: характеризующие свойства объектов, с которыми взаимодействовал пользователь (семантику поведения) и описывающие непосредственно само поведение пользователя. Вновь обратимся к системам веб аналитики, как самому распространенному аналогу разрабатываемой системы. Google Analytics [7] обладает широким списком характеристик поведения и личности пользователя [9], однако, как уже отмечалось ранее, не способна характеризовать семантику поведения. Для сервиса graphica.ai это является существенным недостатком, так как свойства изображений, которые просматривают пользователи, напрямую отражают интересы пользователя. А предпочтения человека играют важную роль в задачах из области маркетинга (продвижение,

налаживание контакта с аудиторией) и рекомендательных систем.

Возникает задача сопоставления изображений и их метаинформации с поведенческими логами для вычисления семантических атрибутов. Тут стоит отметить методы снижения размерности и, в частности, выделение признаков [24]. Например, использование ансамбля предобученных сверточной и сиамской нейронных сетей позволило описывать изображения вектором, таким образом, предоставив возможность использования изображений для уточнения семантики взаимодействия [30].

Описывать и упрощать можно не только данные о семантике, но и сами поведенческие логи. Это достигается, например, за счет построения последовательностей действий, совершаемых пользователем за одну сессию, а также их сокращением до типовых шаблонов поведения за счет сравнения этих последовательностей [22]. Таким образом предлагается снизить объем занимаемого пространства на диске при хранении логов.

1.4. Сегментирование пользователей

Конечным результатом работы системы является множество типовых профилей. Типовой профиль — набор атрибутов и их значений, соответствующий сегменту пользователей со схожим поведением. Он позволяет описать группу людей набором характеристик. Например, Google Analytics [7] позволяет группировать пользователей по их географическим, демографическим признакам, типу устройства, и множеству других характеристик. Однако в рамках данной работы наибольший интерес представляет сегментация пользователей, основываясь на их поведении.

Здесь можно выделить два подхода. Первый заключается в выявлении классов пользователей, например при помощи маркетинговых исследований [25] и использовании алгоритмов классификации для предсказания класса каждого конкретного пользователя. Похожий подход продемонстрирован в работе по созданию системы рекомендации блюд [18].

Авторы, собрав личные данные пользователей при помощи опроса, выделили группы респондентов, основываясь на их культуре, религии, состоянии здоровья и личных предпочтениях. Полученные классы использовались для определения особых текстовых запросов, характерных каждой группе. Это позволило определять класс каждого нового пользователя, исходя из его поисковых запросов, и использовать профиль класса для рекомендации определенных блюд (Рис. 4). Главным недостатком такого подхода является трудоемкость процесс выявления классов пользователей. Помимо того, сложно провести эксперимент для определения предпочтений людей в искусстве (предметная область graphica.ai).

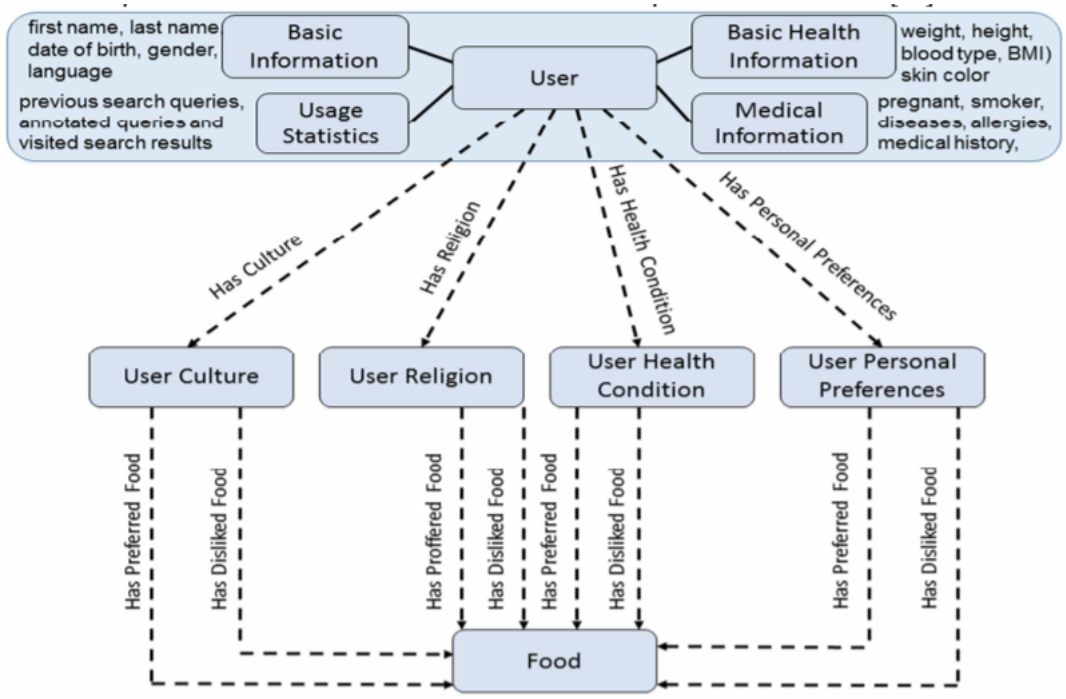

Рис. 4: Структура рекомендательной системы с использованием профиля пользователя [18]

Гораздо больший интерес в рамках поставленной задачи представляет второй подход, характерный отсутствием заранее известных классов пользователей. Для сегментирования пользователей применяется класс алгоритмов машинного обучения без учителя. Одним из таких алгоритмов является SOM (Self-organizing map) [12]. Его преимущество

заключается в способности спроецировать пространство высокой размерности входных данных в выходное пространство более низкой размерности. Помимо этого, он способен кластеризовать данные и устойчив к выбросам, что выгодно выделяет его на фоне других алгоритмов снижения размерности. В случае кластеризации поведенческих логов это дает возможность получить результат в виде групп пользователей со схожим набором поведенческих логов [20]. Недостаток методов снижения размерности заключается в том, что выходной набор атрибутов не является интерпретируемым. А этот критерий является основным в рамках задачи построения профилей.

Среди алгоритмов кластеризации стоит также выделить два типа: с фиксированным количеством кластеров и без него. Обзор первого типа произведен в работе [3] в контексте создания рекомендательных систем. Классический алгоритм k-Means [13] можно использовать для предварительной кластеризации пользователей перед применением SOM. В работе [6] применяется алгоритм k-Means для создания микросегментов пользователей на основе их характеристик и упорядочивания полученных кластеров. Использование приведенного выше алгоритма призвано снизить количество признаков входных данных для SOM, что, в свою очередь, повышает качество и точность сегментирования.

Качество сегментирования сильно колеблется в зависимости от выбранного числа кластеров. При этом аудитория сервиса graphica.ai постоянно растет, что может приводить к появлению новых сегментов пользователей. Поэтому алгоритмы, способные автоматически определять число кластеров, вызывают повышенный интерес. Одним из таких алгоритмов является Meanshift [26]. Он показывал одни из наиболее стабильных результатов в сравнении [2] на разнообразных наборах данных, в том числе и на Statlog — наборе данных с характеристиками клиентов банка и их кредитными рисками [11]. Этот набор данных достаточно сильно приближен к данным задачи сегментирования пользователей.

2. Описание подхода

Разрабатываемая система собирает, хранит и анализирует поведенческие логи для построения типовых профилей, а также предоставляет интерфейс для их получения. Рассмотрим процесс работы такой системы (Рис. 5).

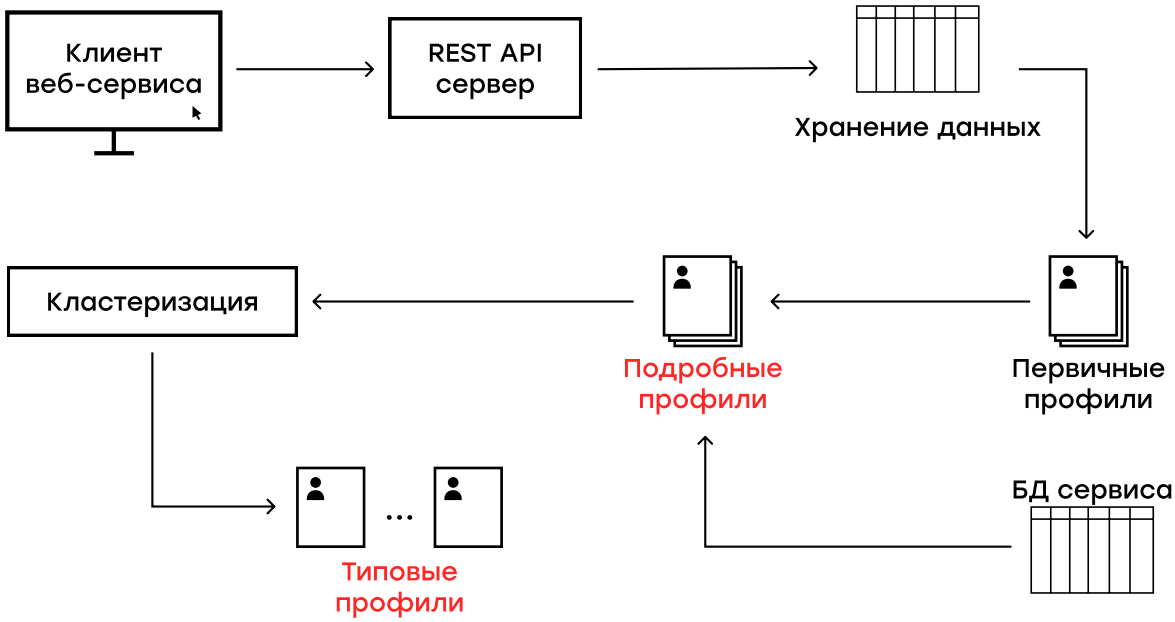

Рис. 5: Процесс работы системы профилирования пользователей

На клиенте сервиса регистрируются события, формируются поведенческие логи и отправляются на сервер. Там логи проверяются на корректность и записываются в базу данных. Далее в соответствии с расписанием или по запросу сервер рассчитывает профили для каждого пользователя и типовые профили. Это происходит следующим образом. Сырые логи агрегируются до более сложных конструкций, содержащих продолжительность события. Из них формируются атрибуты пользователей (первичные профили). После этого сервер обращается к базе данных веб-сервиса graphica.ai для получения информации об объектах и медиаматериалах, с которыми взаимодействовали пользователи, и дополняет список атрибутов (например, величиной, характеризующей разнообразность просмотренных медиаматериалов). Таким обра-

зом формируются профили для каждого пользователя. Далее происходит кластеризация всех профилей для разделения на группы схожих по поведению пользователей и рассчитываются усредненные значения атрибутов для каждой группы, формируя типовые профили. Наконец, сервер также предоставляет интерфейс доступа к построенным профилям.

3. Реализация

Для реализации предлагаемого подхода была разработана система, включающая серверную и клиентскую части (Рис. 6). Подробное описание реализации представлено в секциях ниже.

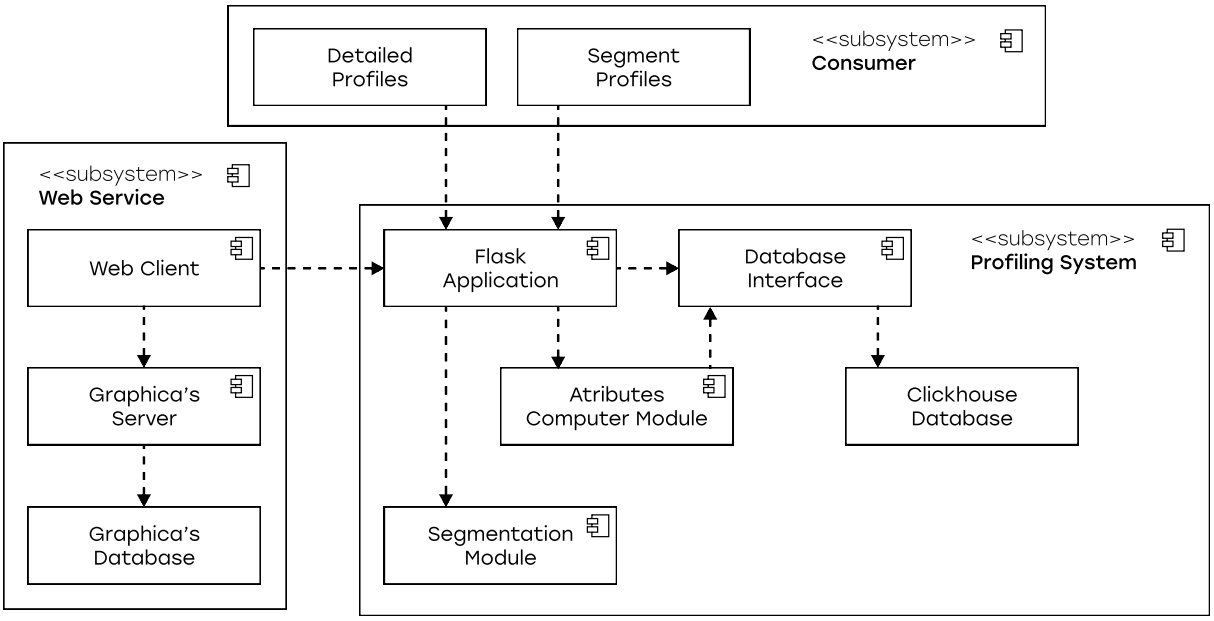

Рис. 6: Диаграмма компонентов системы профилирования пользователей

3.1. Сервер

Серверная часть приложения ("Profiling System" на Рис. 6) реализует значительную долю процесса работы системы, однако для внешнего использования предлагает всего две функции.

1. Предоставление интерфейса для сохранения поведенческих логов
2. Предоставление интерфейса доступа к построенным профилям

Поддержка этих функций реализована на языке Python при помощи библиотеки Flask[1] и интерфейса REST API. Помимо этого, на сервере содержится служба APScheduler[2], которая запускает пересчет профи-

[1]https://flask.palletsprojects.com/en/1.1.x/

[2]https://apscheduler.readthedocs.io/en/stable/

лей в соответствии с расписанием: раз в три часа. Благодаря возможности изменять периодичность пересчета профилей, получается контролировать баланс между нагрузкой на основной сервер веб-сервиса (за счет частоты обращений к базе данных) и актуальностью профилей пользователей.

Что касается построения профилей, по аналогии с процессом (Рис. 5), можно выделить несколько модулей:

1. Хранение поведенческих логов (“Clickhouse Database” на Рис. 6)
2. Вычисление атрибутов двух типов, приведенных ниже, и их объединение для построения профиля каждого пользователя (“Atributes Computer Module” на Рис. 6):
   (a) Атрибуты, характеризующие поведение пользователей
   (b) Атрибуты, характеризующие медиаматериалы, с которыми взаимодействовали пользователи
3. Кластеризация профилей и вычисление усредненных атрибутов для каждой группы (“Segmentation Module” на Рис. 6)

3.1.1. Хранение поведенческих логов

Одной из особенностью хранения поведенческих логов в рамках поставленной задачи (“Clickhouse Database” на Рис. 6) является совмещение принципов систем OLTP (Online Transaction Processing) и OLAP (Online Analytical Processing). Набор входящих на сервер поведенческих логов представляет собой непрерывный поток записей, который необходимо сохранять. Обработка такого поведения характерна OLTP системам. Однако огромное количество таких записей может негативно сказаться на скорости выполнения аналитических запросов к большому подмножеству записей, которыми являются запросы, формирующие атрибуты пользователей.

Этим критериям соответствует СУБД Clickhouse в силу того, что она фокусируется на принципах OLAP систем. Ее колоночная структура позволяет выполнять аналитические запросы быстрее аналогов [17] и

хранить данные с высокой степенью сжатия [31], что позволяет снизить размер расходов веб-сервиса на содержание системы профилирования. Кроме этого, команда graphica.ai обладает опытом использования данной СУБД, что положительно скажется на поддержке системы профилирования. Что касается сохранения потока логов, Clickhouse имеет возможность создавать буферные таблицы [29], которые хранятся в оперативной памяти. Благодаря этому имеется возможность вставлять входящие логи в буферную таблицу и при наступлении некоторого события (например, заполнения оперативной памяти или с временным интервалом) производить вставку большого количества записей в основную таблицу. У такого подхода есть существенный недостаток: в случае отключения сервера базы данных теряются все записи из буферной таблицы. Однако в силу высокой стабильности облачных серверов [4], вероятность отключения базы данных невелика. При этом из-за того, что профили формируются по логам за большой промежуток времени, возможная потеря нескольких секунд логирования практически не влияет на результат.

Поэтому потери могут оказаться незначительными (буферная таблица хранит логи за последнюю секунду), так как исчезают логи за небольшой промежуток времени, что слабо повлияет на конечный результат работы системы — профили пользователей.

3.1.2. Вычисление атрибутов

Большая часть логики вычисления атрибутов содержится в множестве представлений и SQL запросов (“Database Interface” на Рис. 6): это позволяет по максимуму использовать производительность колоночной СУБД. Построение профилей для каждого пользователя можно разделить на два этапа. Сначала логи обобщаются до предопределенных шаблонов поведения, содержащих длительность взаимодействия. Примеры такого обощения приведены ниже.

- Несколько событий прокрутки страницы объединяются в одно
- Клик по изображению и любое другое последующее событие на

других страницах формируют начало и конец события просмотра изображения

- Наведение мыши на объект и уведение мыши с объекта обобщаются до события наведения на протяжении определенного времени
- События каждого пользователя разделяются на сессии (группу событий), исходя из времени прошедшего с момнента возникновения последнего события на момент наступления каждого из них

Из множества таких шаблонов поведения вычисляются атрибуты, характеризующие поведение каждого пользователя. Ниже приведены примеры таких атрибутов.

- Интенсивность прокрутки
- Доля обратной прокрутки
- Количество времени, проведенное в сервисе
- Среднее время просмотра изображения
- Количество просмотренных изображений
- Среднее число сессий в неделю
- Средняя продолжительность сессии

Для вычисления атрибутов, характеризующих просмотренные изображения, система обращается к базе данных веб-сервиса graphica.ai с помощью технологии GraphQL и получает следующую информацию.

- Название изображения
- Дата загрузки
- Автор
- Список категорий, к которым относится изображение

Затем эти данные сопоставляются с множеством всех изображений, которые просматривали пользователи, для вычислений следующих атрибутов для каждого пользователя.

- Наиболее любимые категории
- Любимый автор
- Широта интереса по категориям
- Широта интереса по авторам

Набор атрибутов и их значений для каждого пользователя объединяется в один набор данных (таблицу), формируя, таким образом, множество профилей для каждого пользователя. Ниже приведен пример профилей для пользователей “А”, “Б”, “В”. Количество атрибутов сокращено с 16 до 3 для удобства демонстрации.

Пользователь	Среднее время просмотра изображения	Количество просмотренных изображений	Средняя продолжительность сессии
А	6.72с	11	177с
Б	3.6с	1	7с
В	15.1с	4	199с

3.1.3. Кластеризация профилей

Как уже упоминалось ранее, одной из целью системы профилирования пользователей является построение профилей для групп пользователей со схожим поведением — пользовательских сегментов. Для достижения этой цели было решено кластеризовать множество профилей пользователей, полученное в результате работы модуля вычисления атрибутов. При этом конечной целью являются профили пользовательских сегментов, которые может интерпретировать человек — специалист в области маркетинга или дизайна. Помимо интерпретируемости центров кластеров, полученных в результате работы модели кластеризации, существует ограничение связанное с оценкой качества модели.

В силу отсутствия заранее известных классов пользователей невозможно оценить модель с помощью внешних метрик, поскольку они основаны на сравнении результата кластеризации с априори известным разделением на классы. При этом небольшое количество пользователей сервиса graphica.ai приводит к неточным оценкам с использованием внутренних мер: алгоритм, показывающий хорошие результаты на небольшом объеме данных может оказаться неэффективным с ростом пользовательской базы. Поэтому было принято решение полагаться на мнение экспертов — маркетологов и дизайнеров — для оценки качества модели. С ростом числа пользователей могут появляться новые сегменты пользователей. Это накладывает ограничения на алгоритм кластеризации: он должен уметь адаптироваться к изменяющемуся числу кластеров. С этой задачей прекрасно справляется MeanShift [26]. Благодаря автоматическому подбору количества кластеров, он позволит показывать стабильные результаты с увеличением числа пользователей [2]. Помимо этого, он способен выделять объекты, значительно отличающиеся от основной массы. В рамках системы профилирования это позволит выявлять пользователей с аномальным поведением. Была использована реализация этого алгоритма из инструментария Scikit-learn. С ее помощью множество профили пользователей разбивается на кластеры, после чего из центров кластеров формируются типовые профили пользователей. При этом сохраняется полный набор атрибутов, характеризующий каждого пользователя, а их значения вычисляются с помощью усреднения характеристик пользователей, принадлежащих одному кластеру.

3.2. Клиент

Для сбора поведенческих логов и тестирования системы профилирования был разработан клиент при помощи библиотеки React[3] и сервер, моделирующий сервер graphica.ai при помощи Django[4]. Этот клиент

[3]https://reactjs.org/

[4]https://www.djangoproject.com/

призван воспроизвести взаимодействие пользователя с медиаматериалами в клиенте веб-сервиса graphica.ai, от которого и исходил запрос на создание инструмента для создания профилей. Основная задача тестового клиента — обеспечение сбора поведенческих логов. Эта потребность реализована при помощи обработки JavaScript событий и интерфейса REST API.

Регистрируются следующие виды событий.

- Наведение мыши на изображение
- Уведение мыши с изображения
- Прокрутка страницы
- Клик по изображения

При этом собирается информация, которая формирует пользовательский лог.

- Идентификатор пользователя
- Тип объекта, с которым произошло взаимодействие
- Идентификатор объекта
- Вид события
- Значение, генерируемое событием (например, степень прокрутки страницы)
- Время возникновения события

Наконец, пользовательские логи отправляются клиентом на сервер при помощи библиотеки Axios[5] и интерфейса REST API для дальнейшей обработки.

[5]https://github.com/axios/axios

4. Тестирование

Для оценки работоспособности системы было необходимо интегрировать предложенное решение в веб-сервис graphica.ai. В силу сжатых сроков разработки, не удалось внедрить сбор данных в клиент этого сервиса. Однако взаимодействие с базой данных graphica.ai для получения информации об изображениях было налажено. Таким образом, получилось интегрировать систему профилирования пользователей с серверной частью веб-сервиса. Этого было достаточно, чтобы продемонстрировать работоспособность системы. Для этой демонстрации был поставлен следующий эксперимент.

Было приглашено 17 людей, которым предложили воспользоваться клиентом сервиса для просмотра изображений. В качестве клиента выступал описанный в предыдущем разделе тестовый клиент, который производил сбор поведенческих логов и их отправку на сервер системы профилирования. В результате работы системы 17 пользователей были разделены на 4 сегмента и для них были построены типовые профили (Рис. 7). Для каждого сегмента были дополнительно подсчитаны уровни внимательности, вовлеченности и опыта исходя из их атрибутов. Они приведены в таблице под Рис. 7.

	scroll_intensity	avg_session_duration	avg_views_per_session	sessions_amount	categories_breadth_of_interest	users_count
0	0.136	130.5	6.375	1.0	0.621	10
1	0.464	3.4	0.000	0.6	0.000	5
2	0.002	245.0	0.000	1.0	0.000	1
3	0.000	302.0	1.750	5.0	0.528	1

Рис. 7: Часть типовых профилей построенных в результате эксперимента

Сегмент	Внимательность	Вовлеченность	Опыт
0	Высокая	Средняя	Средний
1	Низкая	Низкая	Средний
2	Средняя	Низкая	Средний
3	Высокая	Высокая	Большой

Система выделила две группы (1 и 2) с крайне низкой продолжительностью сессии (Рис. 8). Они разделены между собой прежде всего по количеству времени, проведенному за просмотром каталога картинок: пользователь из группы 2 потратил в сервисе довольно много времени, в отличие от группы 1 — те ушли очень быстро (Рис. 9). Отсюда вытекает более высокий уровень внимательности группы 2.

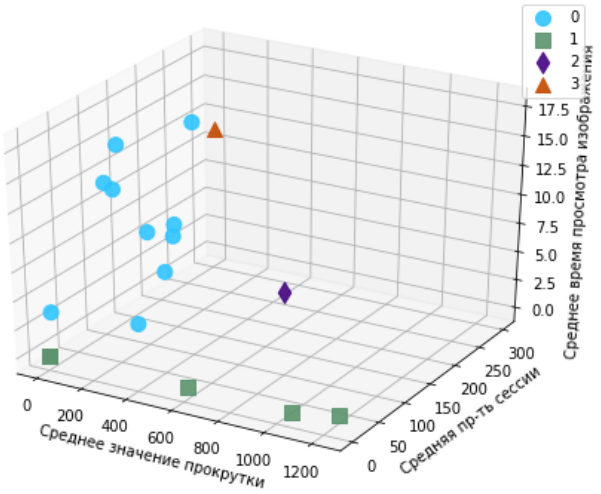

Рис. 8: Сегменты пользователей

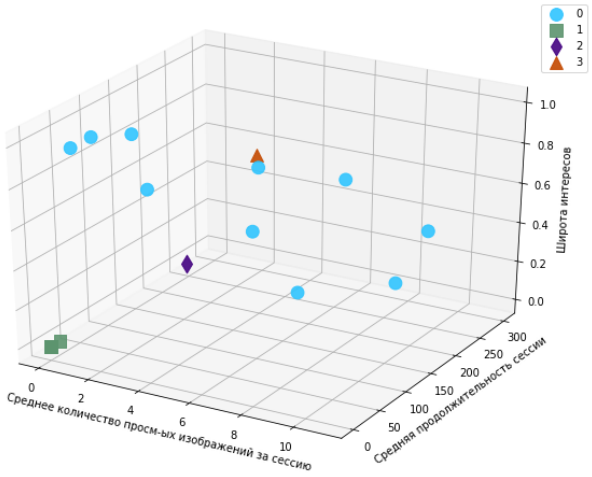

Рис. 9: Сегменты пользователей

Среди оставшихся пользователей были выделены две группы: группа 3, с более узкой областью интересов и группа 0 с меньшей продолжительностью сессии (Рис. 10). Таким образом получилось сделать вывод, что пользователи из группы 3 сильнее вовлечены (дольше и больше просматривают изображения) и опытны (сфокусированы на меньшем количестве категорий).

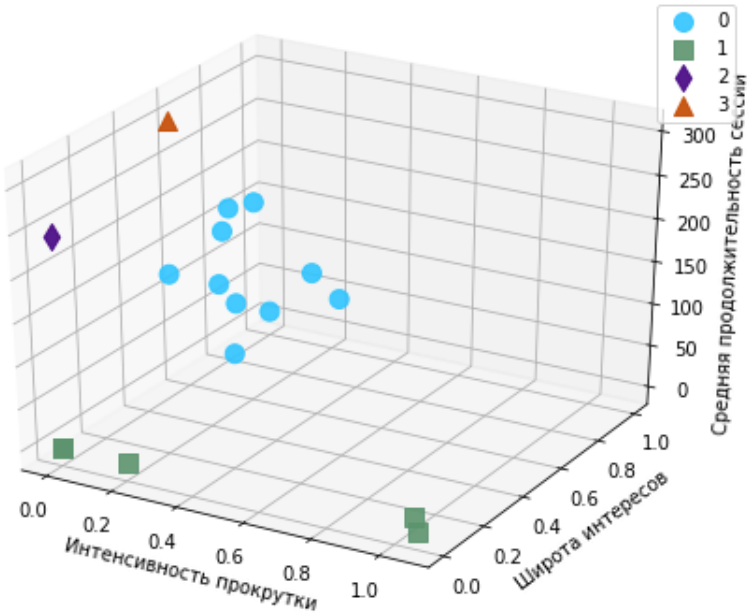

Рис. 10: Сегменты пользователей

Полученные результаты были продемонстрированы команде graphica.ai, которая включает в себя экспертов в области дизайна и маркетинга. Они согласились, что такое разделение пользователей на сегменты оправдано, и положительно оценили результат работы системы профилирования.

> “Эта дополнительная информация поможет нам подходить к пользователю индивидуально, например, предлагать сотрудничество, персонализированные рекомендации и так далее”.

Заключение

В результате работы были выполнены следующие задачи.

- Проведен анализ существующих решений и алгоритмов
- Разработана система построения типовых пользовательских профилей и клиент для тестирования системы
- Получена положительная оценка качества построения типовых профилей пользователей с помощью привлечения экспертов в области дизайна и маркетинга
- Проведена интеграция в архитектуру существующей системы graphica.ai

Таким образом, была разработана система построения типовых профилей пользователей, выгодно отличающаяся от существующих аналогов способностью оценить информацию об объектах, с которыми взаимодействовали пользователи. Это дало возможность характеризовать группу пользователей не только особенностями их поведения, но и характеристиками изображений, которые просматривали пользователи. Такое решение положительно оценили члены команды graphica.ai.

Список литературы

[1] Beyond clicks: Dwell time for personalization / Xing Yi, Liangjie Hong, Erheng Zhong et al. // RecSys 2014 - Proceedings of the 8th ACM Conference on Recommender Systems. — 2014. — 10. — P. 113–120.

[2] Chakraborty Tanmoy. EC3: Combining Clustering and Classification for Ensemble Learning // CoRR. — 2017. — Vol. abs/1708.08591. — 1708.08591.

[3] Comparison of Collaborative Filtering Algorithms: Limitations of Current Techniques and Proposals for Scalable, High-Performance Recommender Systems / Fidel Cacheda, Víctor Carneiro, Diego Fernández, Vreixo Formoso // ACM Trans. Web. — 2011. — Feb. — Vol. 5, no. 1. — Access mode: https://doi.org/10.1145/1921591.1921593.

[4] Digitalocean. 2018 // DigitalOcean. — 2018. — Режим доступа: https://www.digitalocean.com/docs/platform/droplet-policies/ (дата обращения: 15.05.2020).

[5] Facebook. Facebook Pixel // Facebook for Business. — Режим доступа: https://www.facebook.com/business/learn/facebook-ads-pixel (дата обращения: 09.12.2019).

[6] Goncarovs Pavels. Using Data Analytics for Customers Segmentation: Experimental Study at a Financial Institution. — 2018. — 10. — P. 1–5.

[7] Google. Google Analytics // Google Analytics. — Режим доступа: https://analytics.google.com/analytics/web/provision/#/provision (дата обращения: 09.12.2019).

[8] Google. Tag Manager overview // Google Tag Manager. — Режим доступа: https://support.google.com/tagmanager/answer/6102821?hl=en (дата обращения: 09.12.2019).

[9] Google. Параметры и показатели // Справка — Google Analytics. — Режим доступа: https://support.google.com/analytics/answer/1033861 (дата обращения: 18.05.2020).

[10] Hanamanthrao Ramanna, Thejaswini S. Real-time clickstream data analytics and visualization // 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT). — 2017. — P. 2139–2144.

[11] Hofmann Professor Dr. Hans. Statlog (German Credit Data) Data Set // UCI. Machine Learning Repository. — Режим доступа: https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data) (дата обращения: 09.12.2019).

[12] Kohonen Teuvo. Self-Organization and Associative Memory / T. Kohonen. — 1984. — 01.

[13] Li Youguo, Wu Haiyan. A Clustering Method Based on K-Means Algorithm // Physics Procedia. — 2012. — 12. — Vol. 25. — P. 1104–1109.

[14] Messier Stacey. Design Thinking: What is an Empathy Interview? // Medium. — 2017. — Jan. — Режим доступа: https://medium.com/@StaceyDyer/design-thinking-what-is-an-empathy-interview-25f71bd496d7 (дата обращения: 09.12.2019).

[15] Ming He Xiaofei Wu Jiuling Zhang Ruihai Dong. UP-TreeRec: Building Dynamic User Profiles Tree for News Recommendation // China Communications. —

2019. — Vol. 16, no. 4. — P. 219. — Access mode: http://www.cic-chinacommunications.cn/EN/abstract/article_890.shtml.

[16] Mixpanel. Product and User Behavioral Analytics for Mobile, Web, More // Mixpanel. — Режим доступа: https://mixpanel.com/ (дата обращения: 09.12.2019).

[17] Monakhov Andrey. ClickHouse vs Amazon RedShift Benchmark // Altinity. — 2017. — Режим доступа: https://www.altinity.com/blog/2017/6/20/clickhouse-vs-redshift (дата обращения: 15.05.2020).

[18] Nazer Ahmed, Helmy Tarek, Al-Mulhem Muhammed. User's Profile Ontology-based Semantic Framework for Personalized Food and Nutrition Recommendation // Procedia Computer Science. — 2014. — 12. — Vol. 32. — P. 101–108.

[19] Patterns and Sequences: Interactive Exploration of Clickstreamsto Understand Common Visitor Paths / Zhicheng Liu, Yang Wang, Mira Dontcheva et al. // IEEE Transactions on Visualization and Computer Graphics. — 2016. — 01. — Vol. 23. — P. 1–1.

[20] Scalable parallel SOM learning for web user profiles / Lukas Vojacek, Jiří Dvorský, Kateřina Slaninová, Jan Martinovic // International Conference on Intelligent Systems Design and Applications, ISDA. — 2014. — 10. — P. 283–288.

[21] Segment. Customer Data Infrastructure (CDI) // Segment. — Режим доступа: https://segment.com/ (дата обращения: 09.12.2019).

[22] Slaninová Kateřina. User behavioural patterns and reduced user profiles extracted from log files // International Conference on Intelligent Systems Design and Applications, ISDA. — 2014. — 10. — P. 289–294.

[23] Wikipedia. Netflix Prize // Википедия, свободная энциклопедия. — 2007. — Режим доступа: https://en.wikipedia.org/wiki/Netflix_Prize (дата обращения: 08.12.2019).

[24] Wikipedia. Feature extraction // Википедия, свободная энциклопедия. — 2019. — Режим доступа: https://en.wikipedia.org/wiki/Feature_extraction (дата обращения: 08.12.2019).

[25] Wikipedia. Marketing research // Википедия, свободная энциклопедия. — 2019. — Режим доступа: https://en.wikipedia.org/wiki/Marketing_research (дата обращения: 09.12.2019).

[26] Wikipedia. Mean shift // Википедия, свободная энциклопедия. — 2020. — Режим доступа: https://en.wikipedia.org/wiki/Mean_shift (дата обращения: 19.05.2020).

[27] Wikipedia. Online analytical processing // Википедия, свободная энциклопедия. — 2020. — Режим доступа: https://en.wikipedia.org/wiki/Online_analytical_processing (дата обращения: 20.05.2020).

[28] Wikipedia. Online transaction processing // Википедия, свободная энциклопедия. — 2020. — Режим доступа: https://en.wikipedia.org/wiki/Online_transaction_processing (дата обращения: 20.05.2020).

[29] Yandex. Buffer // Clickhouse. — 2020. — Режим доступа: https://clickhouse.tech/docs/en/engines/table-engines/special/buffer/ (дата обращения: 15.05.2020).

[30] Yang Longqi, Hsieh Andy, Estrin Deborah. Beyond Classification: Latent User Interests Profiling from Visual Contents Analysis. — 2015. — 12.

[31] team Altinity. Compression in ClickHouse // Altinity. — 2017. — Режим доступа: https://www.altinity.com/blog/2017/11/21/compression-in-clickhouse (дата обращения: 15.05.2020).

[32] Яндекс. Как создать и установить счетчик // Метрика. Помощь. — Режим доступа: https://yandex.ru/support/metrica/general/creating-counter.html (дата обращения: 18.05.2020).

[33] Яндекс. О сервисе // Метрика. Помощь. — Режим доступа: https://yandex.ru/support/metrica/ (дата обращения: 09.12.2019).

[34] Яндекс. Описание типа log request // Метрика. Помощь. — Режим доступа: https://yandex.ru/dev/metrika/doc/api2/logs/fields/hits-docpage/ (дата обращения: 18.05.2020).