Оценка востребованности онлайновой научной публикации на основе анализа данных log-файла web-сервера

Ю.Г. Ревякин

ИПМ им. М.В. Келдыша РАН

Аннотация. Работа посвящена практическим вопросам создания программного инструмента для оценки востребованности онлайновой научной публикации на основе анализа данных log-файла. Обсуждается проблема фильтрации запросов от программ-роботов для корректного определения показателей посещаемости научных ресурсов интернет.

Ключевые слова: научная публикация, статистика посещений, анализ log-файла, web-роботы, фильтрация запросов

Evaluating the demand for an online scientific publication based on analysis of web log-file

Yurii Revyakin

Keldysh Institute of Applied Mathematics

Abstract. The work is devoted to practical issues of creating a software tool for evaluating the demand for an online scientific publication based on log data analysis. The problem of filtering requests from robots for correct determination of attendance indicators of scientific resources on the Internet is discussed.

Keywords: scientific publication, visitor statistics, web log file analysis, web robots, query filtering

1. Задача оценки востребованности интернет-ресурсов

Наверное, любой автор, разместивший свой материал во Всемирной паутине, рано или поздно задается вопросами – нашло ли его произведение своих читателей, сколько их и как они относятся к его творению.

Вопросы эти возникали так часто, что появилась отдельная дисциплина web-аналитики как набора методик для сбора и анализа

количественных и качественных данных об использовании интернет-ресурсов.

Зародившись почти одновременно с WWW, web-аналитика прошла большой путь развития – от простых счетчиков обращений к одностраничным web-сайтам до систем анализа поведения пользователей при посещении интернет-порталов. Этот путь эволюции и развития web-аналитики во многом определяется потребностями растущей сферы электронной торговли, но не стоит забывать и тот факт, что Всемирная Паутина была и остается крупнейшей платформой обмена и распространения научной информацией. Некоторые из методик и критериев современной wed-аналитики могут быть напрямую использованы для оценки эффективности научных сайтов, применение других требует их адаптации и переосмысления [1].

2. Выбор критерия оценки востребованности научной публикации

Но стандартные показатели, используемые в web-аналитике для оценки эффективности интернет-ресурсов, в большинстве своем адресованы тем, кто занимается развитием и сопровождением web-сайтов. Авторов же отдельных научных публикаций, размещенных в интернет, интересуют прежде всего данные о востребованности их собственных произведений.

Основным показателем такой востребованности является, безусловно, число запросов полного текста онлайновой публикации, при условии, что эти обращения исходят от реальных посетителей интернет-ресурса. Такая статистика может быть представлена в простой и сжатой форме и объединена с описанием самой публикации в электронном каталоге публикаций.

3. Log-файл web-сервера как источник первичных данных оценки востребованности публикации

При просмотре пользователем интернет-ресурса каждый запрос на загрузку html-страницы и ее составляющих фиксируется, как правило, web-сервером в отдельном файле журналирования (log-файле). Формат записи log-файла определяется соответствующей директивой конфигурации web-сервера, и такая запись содержит достаточно подробную информацию о запросе и его источнике.

Приведем в качестве примера запись log-файла, которая остается после обработки web-сервером обращения к заглавной странице сайта www.keldysh.ru:

194.226.57.126 - - [11/May/2020:16:48:49 +0300] "GET / HTTP/1.1" 200 16541 "-" "Mozilla/5.0 (Windows NT 6.1; Win64; x64)

AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.90 Safari/537.36"

Поля представленной записи содержат все основные атрибуты выполненного web-сервером запроса: IP-адрес источника; временную отметку выполнения запроса; строку команды запроса, включая URL запрашиваемого ресурса; строку описания удаленного клиентского приложения.

Таким образом, имея доступ к log-файлу web-сервера, задача определения числа обращений к полным текстам научных публикаций представляется достаточно простой и не требующей дальнейшего обсуждения.

4. «Искусственный» трафик сети интернет

Но авторов публикаций интересуют в первую очередь запросы, исходящие от реальных, «живых» читателей их произведений.

Поэтому необходимо уметь выделять и исключать из дальнейшего рассмотрения среди всех запросов на загрузку полного текста публикации запросы, порожденные сетевыми приложениями, автоматически сканирующими страницы сайта. Такие сетевые приложения принято называть программами-роботами. Предварительное распознавание запросов, исходящих от программ-роботов, необходимо еще и потому, что научные публикации в первую очередь представляют интерес для достаточно небольшого круга специалистов, круг их потенциальных читателей относительно невелик. Любая же программа-робот, запущенная для сканирования сайтов, способна исказить статистику обращений к сайту до неузнаваемости.

Чтобы выделить автоматически генерируемые запросы программ-роботов из общего трафика, необходимо разобраться: кто и зачем использует программы-роботы. В первую очередь программы-роботы используются всеми крупнейшими поисковыми системами сети Интернет (Google, Yandex и др.) для извлечения информации со страниц сайтов и последующего индексирования в своих базах данных. Такие программы-роботы подчиняются при сканировании сайтов определенным соглашениям и явно обозначают себя в заголовке запроса к сайту. Определение запросов, исходящих от программ-роботов, в этом случае не составляет большего труда. Существует еще множество подобных более специализированных сетевых приложений, используемых для сбора коммерческой и социальной информации в интернете. То, насколько они в своей реализации соответствуют общепринятым стандартам, полностью зависит от доброй воли их разработчиков. Но программы-роботы могут использоваться и как вредоносное сетевое приложение. Один из самых распространенных сценариев использования их в этом качестве – это атака

сервера сайта многочисленными запросами, ставящая своей целью вызвать отказ сервера в обслуживании дальнейших обращений (так называемая DDoS-атака). Более изощренным примером недобросовестного использования программ-роботов является технология рассылки нежелательной рекламной информации, получившая название "referrer spam". Программы-роботы используются для автоматической генерации и рассылки http-запросов, в которых в качестве адреса страницы - источника запроса сознательно указывается URL-адрес рекламируемого сайта. Таким образом, URL рекламируемого сайта попадает не только в отчеты, генерируемые системой web-аналитики, но и индексируется, в свою очередь, роботами поисковых систем, что повышает поисковый рейтинг рекламируемого сайта. Как правило, недобросовестные программы-роботы игнорируют правила, ограничивающие сканирование документов web-ресурса, и маскируют свою активность, фальсифицируя в заголовке http-запроса имя программы - отправителя запроса. Чтобы выявить подобные запросы в log-файле, приходится применять более сложные подходы и программные алгоритмы.

Проблема распознавания запросов от программ-роботов особенно актуальна для сайтов онлайновых научных библиотек: как показывает практика, число запросов за сутки к тексту научной публикации редко превышает два-три десятка, при этом согласно статистике, приведенной в [3], около трети из них может исходить от программ-роботов. Отметим, что жертвами программ-роботов в интернете являются не только интернет-библиотеки онлайновых научных публикаций. С того момента как объем трафика, приходящего на сайт, превратился в товар со своей вполне конкретной стоимостью, появилось множество технологий манипулирования этой величиной. И для реализации этих технологий создаются все новые программы-роботы, увеличивающие и без того существенную долю «искусственного» трафика в сети Интернет.

5. Методы распознавания запросов от программ-роботов

Если в качестве источника первичных данных для определения популярности научной публикации выступают записи log-файла сервера, то определение запросов, исходящих от программ-роботов, выполняется на этапе первичного разбора записей log-файла. Используемые для распознавания методики очень разнообразны и могут варьироваться от простой проверки полей запроса на совпадение со списком заранее заданных строк до сложных аналитических алгоритмов, использующих вероятностные модели и машинное обучение. Классификация таких методик подробно рассмотрена в [2].

Самый очевидный способ обнаружения запросов, принадлежащих программам-роботам, основан на простом сравнении отдельных полей записи log-файла с заранее известными списками ключевых значений. Так

мы можем обнаружить запрос от программы-робота по идентификатору программного приложения, указанному в строке описания источника запроса. Или сравнить IP-адрес источника запроса с базой данных адресов, активно используемых программами-роботами. Такой способ определения запросов от программ-роботов прост и однозначен, но его применение основано на двух существенных допущениях:

- поля запроса содержат достоверную информацию;
- заранее известен набор ключевых значений, который можно использовать для определения программы-робота по содержимому полей записи log-файла.

Отметим также, что отсутствие или недопустимое значение для некоторых полей записи log-файла также могут рассматриваться как признак принадлежности запроса программе-роботу. Так, «анонимные» запросы с неопределенным полем описания источника запроса, как правило, исходят от программ-роботов, нарушающих сетевые протоколы и соглашения.

Другие методики обнаружения запросов от программ-роботов основаны на анализе общих характеристик последовательности запросов, относящихся к одному визиту пользователя. К числу таких характеристик относятся: распределение запросов по используемым методам запроса и типам запрашиваемых web-ресурсов, интенсивность запросов в единицу времени и объем переданных при этом данных. Процедура выделения запросов от программ-роботов распадается на два этапа: сначала для всех запросов одного визита вычисляются необходимые метрики, затем их значения используются для классификации источника запроса. Причем такая классификация основана на критериях, которые чаще всего определяются эмпирическим путем: известно, например, что программы-роботы во время просмотра сайта чаще запрашивают только атрибуты документов и, как правило, не загружают файлы с изображениями, размещенными на web-странице. Окончательное решение о характере источника запросов принимается при условии выполнения нескольких таких критериев одновременно. Необходимо подчеркнуть, что принятое решение верно только для рассмотренной последовательности запросов. Механизм динамического выделения IP-адресов не позволяет распространить сделанный выбор на все запросы, источником которых является данный IP-адрес.

Развитием методов предыдущей группы являются алгоритмы, основанные на машинном обучении и применении вероятностных моделей. Как и в предыдущем подходе, на первом этапе определяются последовательности запросов, относящиеся к одному визиту пользователя, и для каждого отдельного визита вычисляется набор характеристик. Затем, используя заранее построенную вероятностную модель, принимается

решение о характере источника запросов каждого визита («живой» посетитель или программа-робот). Мы не будем останавливаться подробно на методах этой группы, так как они не используются в рассматриваемой далее реализации счетчика обращений к полным текстам онлайновых научных публикаций.

6. Счетчик числа обращений к полным текстам публикаций онлайновой библиотеки ИПМ РАН

Процедура подсчета числа обращений к полным текстам публикаций онлайновой библиотеки ИПМ РАН реализована как независимая программная утилита, ее текущая версия учитывает обращения пользователей только к полным текстам препринтов ИПМ РАН. Для определения источника запроса применяются как методы, использующие синтаксический анализ полей запроса, так и методы, основанные на анализе характеристик визитов пользователей. Программа не содержит средств формирования статистических отчетов, а только передает вычисленные параметры для последующей обработки программному обеспечению онлайновой библиотеки. После обработки данные о числе запросов полного текста препринта представляются непосредственно в карточке самого препринта, размещенного в онлайновой библиотеке. Программа реализована на языке Ruby [7] и использует дополнительные библиотеки для разбора записей log-файла web-сервера, доступа к базам данных и внешним источникам данных в формате JSON.

7. Счетчик обращений — описание алгоритма

Очень схематично можно разбить выполнение программы подсчета числа обращений к полным текстам препринтов, размещенных в онлайновой библиотеке ИПМ РАН, на три основных этапа:

- анализ первичных данных и выделение запросов к полным текстам препринтов. В качестве источника первичных данных используется log-файл web-сервера Apache версии 2.2,
- определения запросов, исходящих от программ-роботов.
- подсчет числа обращений читателей к полным текстам препринтов и запись результатов в служебную БД онлайновой библиотеки

Анализ первичных данных. Выполняется формирование базы данных (БД) запросов к web-серверу. Все записи, содержащиеся в log-файле web-сервера, последовательно считываются и для каждой записи log-файла создается запись БД, содержащая следующую информацию об исходном http-запросе:

- IP-адрес запроса;
- тип http-запроса;

- время создания запроса;
- код завершения запроса;
- URL-адрес запрашиваемого ресурса;
- строка описания клиентского приложения, создавшего запрос, включая идентификацию программы-клиента.

Для хранения информации об обрабатываемых запросах используется СУБД MongoDB [6] – документо-ориентированная, нереляционная СУБД, позволяющая хранить в одной базе данных документы различной структуры. Программирование запросов к объектам, хранимым в базе данных, во многом облегчается применением Mongoid, инструментария для языка Ruby, реализующего объектно-ориентированный интерфейс доступа к СУБД MongoDB.

Среди всех запросов, занесенных в БД, выбираются успешные http-запросы к файлам, содержащим полные тексты препринтов. Отобрать такие запросы несложно, поскольку имена файлов с полными текстами препринтов формируются единообразно, в соответствии с соглашениями, принятыми в онлайновой библиотеке ИПМ РАН.

Определение запросов, исходящих от программ-роботов. Распознавание запросов от программ-роботов выполняется в несколько этапов, последовательным применением нескольких программных тестов, каждый из которых использует собственные критерии отбора запросов. Рассмотрим методы тестирования запросов, используемые в текущей версии программы, подробнее.

Определение источника запросов по имени программного приложения. Записи о запросах, отобранные для дальнейшего рассмотрения на предыдущем шаге, проверяются на совпадение имени приложения – источника запроса – со списком имен приложений, используемых программами-роботами. Имена приложений программ-роботов хранятся в отдельной базе данных, которая формируется на основе свободно распространяемого списка регулярных выражений, представленного в формате JSON [4]. Каждое регулярное выражение определяет все вариации включения и написания имени программы-робота в строке описания источника http-запроса. Список содержит более 400 регулярных выражений, соответствующих идентификаторам наиболее распространенных программ-роботов, и периодически обновляется. Запросы, которые определяются как исходящие от программы-робота по имени программы-клиента, из дальнейшего рассмотрения исключаются.

Проверка IP-адреса источника запроса. Для каждого запроса к тексту препринта проверяется принадлежность IP-адреса источника к подмножествам адресов, которые используются программами-роботами. Диапазоны адресов хранятся в отдельной базе данных, формируемой на основе текстового списка, получаемого из открытого источника [5].

Проверка на запрос фрагмента файла. Спецификация http-протокола позволяет сформировать запрос для передачи только части документа, указав в заголовке запроса диапазон байт запрашиваемого фрагмента. Таким образом, можно разбить загрузку большого по объему файла на несколько частей и использовать для загрузки каждой части свой запрос. Этим свойством http-протокола широко пользуются современные браузеры для оптимизации и распараллеливания процесса загрузки. Но с точки зрения web-статистики последовательность запросов на загрузку фрагментов файла должна расцениваться как одно обращение пользователя к документу. Четких формальных критериев для выделения такой последовательности запросов нет – каждый браузер реализует собственный алгоритм разбиения загрузки большого документа на несколько запросов. Поэтому одно из решений этой проблемы основывается на допущении, что все запросы на полную или частичную загрузку файла, исходящие от определенного IP-адреса в течение достаточно малого интервала времени, являются следствием одного обращения пользователя к документу. Из выделенной таким образом последовательности запросов далее рассматривается только первый по времени запрос, остальные игнорируются.

На следующем этапе отбора запросов рассматриваются общие характеристики сессии пользователя, включающей текущий запрос к тексту препринта. Для этого необходимо предварительно выделить все запросы, относящиеся к данной сессии. При этом считается, что в одну сессию с текущим запросом входят все запросы, которые удовлетворяют следующим условиям:

- отправлены с того же IP-адреса, что и текущий запрос;
- созданы одним и тем же клиентским приложением (строки описания источника запроса совпадают);
- временной интервал между последовательными запросами в сессии не превышает заранее определенного порогового значения.

Проверка на загрузку файла robots.txt во время сессии. Файл robots.txt был разработан консорциумом W3C для управления поведением программ-роботов при обходе сайта. Этот текстовый файл, размещаемый в корневом каталоге сайта, содержит инструкции, которые регламентируют доступ программам-роботам к отдельным файлам или каталогам сайта. Следование инструкциям, приведенным в файле robots.txt, является для программ-роботов добровольным, однако большинство программ-роботов известных поисковых систем придерживается их и загружает файл robots.txt в начале каждого сеанса просмотра сайта. Таким образом, факт загрузки файла robots.txt во время сеанса с большой вероятностью свидетельствует о том, что все запросы сессии исходят от программы-робота и их можно исключить из дальнейшего рассмотрения.

Все запросы, прошедшие описанные выше этапы отбора, рассматриваются как верифицированные запросы к текстам препринтов от реальных посетителей онлайновой библиотеки.

Подсчет числа обращений и запись результатов. Выделив запросы, которые были определены как исходящие от реальных читателей, несложно подсчитать общее число обращений посетителей к полному тексту каждого препринта. Эти данные заносятся в служебную базу данных SQL-сервера онлайновой библиотеки. Данные о числе обращений для каждого препринта представлены в базе данных записями следующего формата:

- дата обращения;
- URL файла полного текста препринта;
- число обращений к файлу.

Сервер баз данных онлайновой библиотеки реализован на платформе Microsoft SQL Server; для доступа к служебной базе данных используется объектно-ориентированный интерфейс доступа к SQL-базам данных, реализованный в пакете Sequel.

8. Представление статистики обращений к онлайновой публикации

Записанные в служебную SQL базу данных результаты работы программы затем обрабатываются средствами программного обеспечения онлайновой библиотеки и используются для актуализации статистики обращений в основной базе данных электронных изданий. Суммарная статистика обращений к препринтам ИПМ представляется на карточках электронных изданий в следующем формате (см. рис.1):

- общее число обращений от дня публикации препринта или от дня запуска программы подсчета обращений к препринтам (09.01.2020);
- число обращений за последние 30 дней;
- разница в числе обращений по сравнению с предыдущим 30-дневным интервалом сбора статистики.

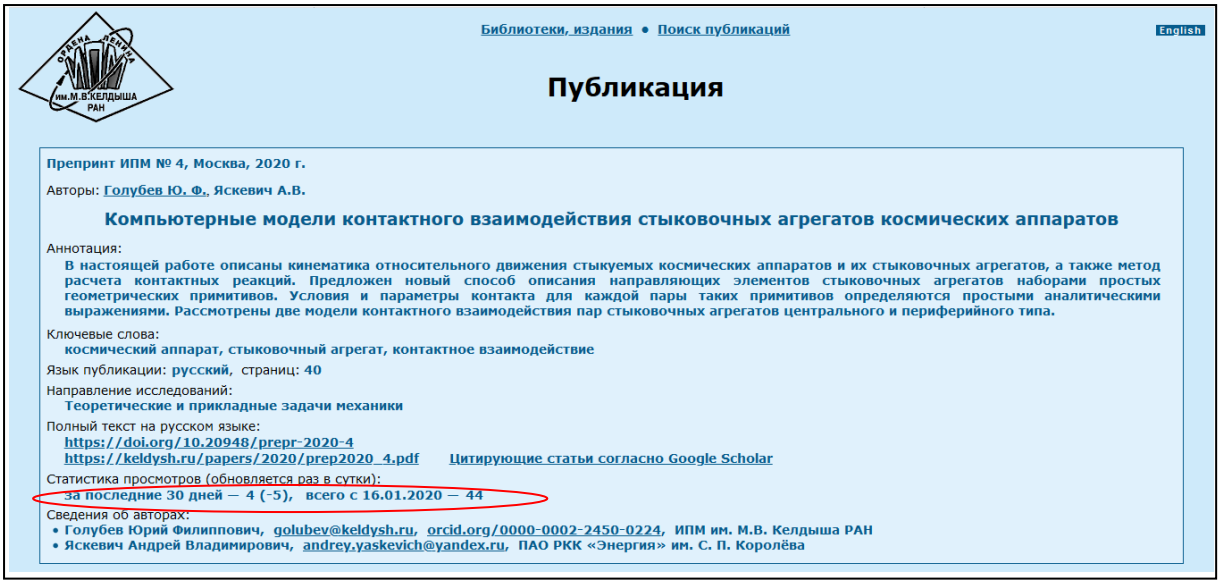

Рис. 1. Карточка электронного издания с представленной информацией о статистике обращений (выделена красным цветом)

Программа подсчета числа обращений к текстам препринтов запускается автоматически в фоновом режиме раз в сутки. Соответственно, раз в сутки обновляется и информация о числе обращений, представленная на карточке онлайновой публикации.

9. Заключение

Необходимость оценки востребованности онлайновых научных публикаций становится все более актуальной по мере вытеснения традиционных, "бумажных", технологий распространения научных знаний интернет-технологиями, предлагающими качественно новые возможности для обмена информацией. Разработка представленного программного инструмента для решения этой задачи преследовала две основные цели:

- получить с достаточной степенью достоверности статистику обращений к полному тексту публикации;
- представить данные о востребованности публикации в простой и сжатой форме, включив их в состав метаданных описания публикации в онлайновой библиотеке.

Реализованный механизм определения источника запросов позволяет распознавать запросы, исходящие от программ-роботов, последовательным применением цепочки программных тестов; набор таких тестов может быть расширен в последующих версиях программы.

Работа выполнена при поддержке Российского фонда фундаментальных исследований, проект 19-01-00069 А.

Литература

1. Ревякин Ю.Г. Возможности web-аналитики для оценки эффективности научных публикаций // Препринты ИПМ им. М.В.Келдыша. 2020. №50. 42с. https://doi.org/10.20948/prepr-2020-50
2. Derek Doran, Swapna S. Gokhale. Web robot detection techniques: overview and limitations// Data Mining and Knowledge Discovery, June, 2011. https://doi.org/10.1007/s10618-010-0180-z
3. Huntington, P., Nicholas, D., & Jamali, H.R. Web robot detection *in the* scholarly information environment // Journal of Information Science, 34(5), *2008, pp.*726–741
4. Syntactic patterns of HTTP user-agents used by bots / robots / crawlers / scrapers / spiders. URL: https://github.com/monperrus/crawler-user-agents
5. Free IP2Location Firewall List by Search Engine. URL: https://www.ip2location.com/free/robot-whitelist.
6. MongoDB official site. URL: https://www.mongodb.com
7. Ruby programming language official site. URL: https://www.ruby-lang.org

References

1. Reviakin Iu.G. Vozmozhnosti web-analitiki dlia otsenki effektivnosti nauchnykh publikatsii // Preprinty IPM im. M.V.Keldysha. 2020. №50. 42c. https://doi.org/10.20948/prepr-2020-50
2. Derek Doran, Swapna S. Gokhale. Web robot detection techniques: overview and limitations// Data Mining and Knowledge Discovery, June, 2011. https://doi.org/10.1007/s10618-010-0180-z
3. Huntington, P., Nicholas, D., & Jamali, H.R. Web robot detection *in the* scholarly information environment // Journal of Information Science, 34(5), *2008, pp.*726–741
4. Syntactic patterns of HTTP user-agents used by bots / robots / crawlers / scrapers / spiders. URL: https://github.com/monperrus/crawler-user-agents
5. Free IP2Location Firewall List by Search Engine. URL: https://www.ip2location.com/free/robot-whitelist.
6. MongoDB official site. URL: https://www.mongodb.com
7. Ruby programming language official site. URL: https://www.ruby-lang.org