

МНОГОАГЕНТНАЯ СИСТЕМА ПОИСКА ИНФОРМАЦИИ ПО ЗАДАНЫМ КРИТЕРИЯМ В СЕТИ ИНТЕРНЕТ

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Антоненко А. В.

Самаль Д. И. – канд. техн. наук, доцент

Сеть Интернет растет очень быстрыми темпами, и найти нужную информацию среди сотен миллиардов веб-страниц и сотен миллионов файлов становится все сложнее. Поэтому создание оптимальной системы поиска по заданным критериям всегда будет актуальной задачей. Выбор заданного критерия поиска позволит значительно сузить пространство для поиска, а использование многоагентной архитектуры позволит увеличить скорость и упростить расширяемость системы.

Поисковая система – программно-аппаратный комплекс с веб-интерфейсом, предоставляющий возможность поиска информации в интернете.

Существует четыре типа поисковых систем: с поисковыми роботами, управляемые человеком, гибридные и метапоисковые системы. Система с поисковыми роботами постоянно исследует сеть и хранит архив копий веб-страниц. Поисковые системы управляемые человеком проверяются вручную, следовательно, и релевантность возвращаемых данных более высокая. Гибридные поисковые системы совмещают оба типа.

Для изучения была выбрана метапоисковая система, суть работы которой заключается в том, что при поисковом запросе параллельно опрашивается несколько независимых поисковых систем и возвращаются их результаты одним, объединенным списком результатов без дублирования ссылок, улучшая частные результаты выдачи. Улучшение результатов достигается за счет того, что использование нескольких поисковых систем повышают вероятность обнаружения искомого документа, а также за счет обеспечения возможности выбора тех поисковых систем, которые лучше всего соответствуют текущим потребностям пользователя. Крупномасштабные поисковые системы, такие как Google или Yandex, не могут тратить много времени на обработку каждого отдельного запроса из-за их огромного количества. Метапоисковые системы не имеют такого ограничения и могут фокусироваться на решении специализированных задач поиска. Немаловажным моментом также является и то, что для реализации такой системы не нужно таких огромных ресурсов как для поисковых систем других типов.

Использование многоагентной архитектуры позволяет увеличить скорость обработки запросов, а также облегчить процесс расширения системы, путем добавления новых агентов. Изображение упрощенной структурной схемы поисковой системы представлено на рисунке 1. В системе можно выделить 4 типа агентов: интерфейсный агент, поисковый агент, агент проверки существования страниц, агент ранжирования страниц. После получения запроса от пользователя интерфейсный агент передает его поисковому агенту, который обладает своей базой знаний: знание конкретной поисковой системы, формата получаемых и возвращаемых ею данных. Поисковой агент получив необходимые данные, передает и агенту проверки существования страниц, который делает запросы по полученным ссылкам и в случае недоступности ресурса отсеивает не существующие страницы. Далее данные передаются агенту ранжирования, который формирует выходной список ссылок.

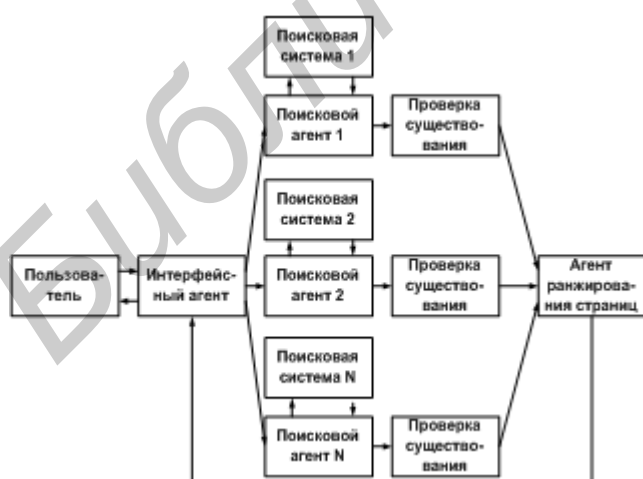


Рис. 1 – Структурная схема многоагентной поисковой системы

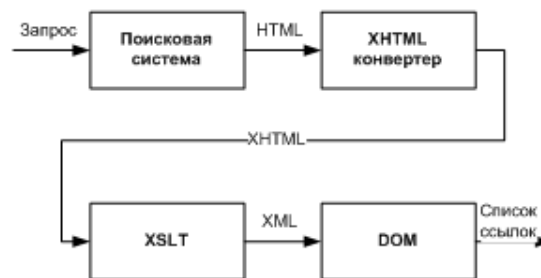


Рис. 2 – Схема получения данных из поисковых систем

На данном этапе в качестве критериев поиска выступают наименования товаров, а именно мобильные телефоны, и доменная зона. Главные проблемы, который были решены в процессе разработки - это построение дерева товаров и получение детальной информации по этим товарам.

Для того, чтобы построить дерево товаров, в xml-файле хранится ссылка на первую страницу интернет-магазина, на которой расположен список представленных товаров, в этом же файле указан шаблон ссылки для перехода на следующую страницу, а также XPath-запрос для получения данных по товарам, специфичный для данного сайта. Процесс выглядит следующим образом:

1. По расписанию запускается программа.
2. Программа считывает xml-файл и находит первую ссылку.
3. Если ссылка доступна, то получает данные используя XPath-запрос.
4. Используя шаблон перехода на следующую страницу, осуществляется получение данных со следующей страницы.
5. Данные формируются в коллекцию.
6. Производится сверка коллекции с уже имеющимися данными в системе и, если находится уникальный товар, то он сохраняется в системе.
7. Далее считывается следующая ссылка и повторяются пункты 2-6 до тех пор, пока есть ссылки.

После того как дерево сформировано, необходимо решить вторую проблему, а именно получение детальной информации по каждому из товаров. Для получения данных можно использовать предоставляемый поисковыми системами интерфейс API, однако не все поисковые системы предоставляют данный интерфейс и зачастую имеющийся интерфейс имеет коммерческие ограничения. Таким образом было предусмотрено собственный вариант получения данных из поисковых систем.

Главное особенностью данного варианта является абстрагирование от исходного кода приложения. То есть при изменении формата данных возвращаемых поисковой системой, код приложения не должен переписываться. Наиболее подходящим для решения такой задачи является XSLT. Первоначально необходимо преобразовать полученный HTML в XHTML, далее к XHTML-документу применяется XSL-таблица, выделяющая из XHTML-документа ссылки на найденные ресурсы и их описания в отдельный XML документ. Эта XSL-таблица является частью знаний поискового агента. На основе полученного XML-документа строится объектное дерево, перемещаясь по которому, выстраиваем список найденных ссылок и их описание. Схематично данный процесс представлен на рисунке 2.

Получение огромного количества данных по запросу от различных поисковых систем, их преобразование и анализ требует достаточно производительного сервера. Для уменьшения нагрузки на сервер запланировано обновление дерева товаров и информации по ним в соответствии с заданным графиком – 1 раз в сутки. Такой подход исключит необходимость для каждого отдельного запроса пользователя выполнять полный цикл поиска, что снизит нагрузку на сервер.

Список использованных источников:

1. Michael Wooldridge, *An Introduction to Multiagent Systems* / John Wiley & Sons; 2nd Edition edition, 2009. – 484 p
2. Википедия: Поисковая система. [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Поисковая_система. – Дата доступа: 15.03.2015.
3. Портал искусственного интеллекта: Многоагентные системы. [Электронный ресурс]. – Режим доступа: <http://www.aiportal.ru/articles/multiagent-systems>. – Дата доступа: 15.03.2015.

МОБИЛЬНАЯ АВТОМАТИЗИРОВАННАЯ СИСТЕМА ОБСЛУЖИВАНИЯ КЛИЕНТОВ ЗАВЕДЕНИЙ ПИТАНИЯ

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Барбасевич А.В., Дубинин А.Ю.

Фролов И.И. – доцент, кандидат технических наук

В наши дни мобильные устройства получают все более широкое применение в повседневной жизни людей. В первую очередь это обусловлено желанием человека постоянно оставаться на связи. Мобильные устройства стали новой ступенью в развитии не только телекоммуникаций между людьми, но и различных других сфер нашей жизни, таких как торговля, маркетинг, экономика, государственное управление и т.д.

В свою очередь одну из лидирующих позиций в мире сейчас занимает компания Apple, которая производит самые популярные смартфоны (iPhone) и планшеты (iPad), предоставляющие неограниченные возможности для пользователей. Данные устройства работают под управлением мобильной операционной системы iOS, которая славится стабильностью, скоростью работы, дизайном и централизованным магазином приложений App Store. С того момента как Apple сделала открытой iOS SDK для разработчиков, App Store начал наполняться различными приложениями: социальными, игровыми, образовательными, медицинскими. За 5 лет существования магазина количество всех приложений достигло цифры в 1 миллион, а число загрузок превысило 60 миллиардов.

Автоматизированная система предназначена для ресторанов, кафе и других заведений питания с целью предоставления возможности посетителю ознакомиться с меню заведения, а также произвести заказ со своего мобильного устройства. Система позволит заведениям увеличить скорость обслуживания клиентов, что должно в свою очередь повысить уровень удовлетворенности посетителей.