

УДК 519.6



СБОР И ОБРАБОТКА ИНФОРМАЦИИ В ИНТЕРНЕТ С ИСПОЛЬЗОВАНИЕМ ИНТЕЛЛЕКТУАЛЬНЫХ АГЕНТОВ

М.В. Збитнева, В.В. Дударь, А.Г. Деревянко

ХНУРЭ, г. Харьков, Украина

В статье предлагается усовершенствованный метод обработки информации о товарах по запросу пользователя с использованием интеллектуальных агентов в среде Интернет. В качестве формального аппарата использована алгебра предикатов первого порядка. Выполнен анализ существующих архитектур программных интеллектуальных агентов с точки зрения их применимости к задаче сбора и обработки web информации.

АГЕНТ, ОБРАБОТКА ИНФОРМАЦИИ, ИНТЕРНЕТ, ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ

Введение

Деятельность отдельных людей и организаций сейчас все больше зависит от имеющейся у них информации и способности ее эффективно использовать. Одним из основных средств получения информации сегодня является глобальная сеть Интернет – динамичная гетерогенная распределенная среда. Эффективный поиск информации в Интернете по мере увеличения объема и рассредоточения ее источников становится все более сложным и трудоемким. При этом критичным является не столько время поиска, сколько отбор информации, релевантной запросу пользователя. Информационный поиск представляет собой процесс сопоставления запроса пользователя со сведениями об информационных ресурсах, известных информационно-поисковой системе, к которой поступил этот запрос. В настоящее время доступ к информации, размещенной в глобальной сети Интернет, в подавляющем большинстве случаев обеспечивается при помощи поисковых машин. По оценкам, к поисковым службам обращается 71% всех пользователей Интернет [5]. Для формирования базы данных об информационных ресурсах поисковая система либо самостоятельно их индексирует, либо использует базы данных других поисковых систем. Второй способ значительно менее трудоемок, однако, то, что структура этих БД уже задана разработчиками, ограничивает параметры поиска.

В зависимости от потребности пользователей сети Интернет скорость поиска определенной информации в ней варьируется и зачастую требует автоматизации для облегчения работы. Это актуально в случае необходимости постоянного отслеживания какой-либо информации в Сети, её обновления, либо манипуляций с уже полученными данными (сортировка, вывод по определенным критериям и так далее). Для такого рода действий используются специальные прикладные программы – «агенты Интернет» [4].

Интеллектуальными агентами Интернет называют программы, автоматизирующие поиск, распознавание, извлечение и анализ информации

во всемирной Сети, ориентированные на нужды конкретного пользователя (или группы пользователей) [4]. Интеллектуальные агенты отличаются от широко используемых в настоящее время поисковых систем Интернет тем что:

1. Они способны работать самостоятельно в течение длительных промежутков времени (дни, недели и более), выполняя задание, порученное пользователем.

2. Как любая программа, однажды созданный агент может быть использован в будущем любое количество раз, в то время как запрос, посланный универсальной поисковой системе, вызывает однократную операцию сбора информации.

3. Поисковый агент передает пользователю не просто результаты работы машины (машин) поиска, а предварительно просматривает документы и выбирает из них наиболее релевантные с его точки зрения. Для этого агент использует определенную технологию оценки релевантности.

4. Агент может настраиваться на предпочтения пользователя, то есть учитывать ряд ограничений на поиск, формулируемых владельцем агента.

5. Некоторые агенты могут работать в off-line режиме, то есть когда пользователь дает задание агенту и отключается от Интернет, а агент отправляется на сервер, где выполняет работу. Когда пользователь снова подключается, агент посылает ему выполненное задание.

6. Агенты могут быть настроены на поиск по расписанию – искать информацию каждый час, день, неделю, месяц и так далее. Эта опция полезна при поиске, например, новостей, какой-то информации, обновление которой происходит довольно часто или информации, которая пользователю необходима постоянно в его работе.

7. Агенты могут обучаться. Пользователь оценивает работу агента, а агент, учитывая эти оценки, может корректировать свои критерии отбора информации.

Таким образом, поисковые агенты могут рассматриваться как интеллектуальная надстройка над машинами поиска.

Агент — это аппаратная или программная сущность, способная действовать в интересах достижения целей, поставленных пользователем. Они описываются также рядом свойств, которые характеризуют понятие агента. Обычно агент обладает набором из следующих свойств [2,3]:

1. Адаптивность: агент обладает способностью обучаться.

2. Автономность: агент работает как самостоятельная программа, ставя себе цели и выполняя действия для достижения этих целей.

3. Коллаборативность: агент может взаимодействовать с другими агентами несколькими способами, например, играя роль поставщика/потребителя информации или одновременно обе эти роли.

4. Способность к рассуждениям: агенты могут обладать частичными знаниями или механизмами вывода, например, знаниями, как приводить данные из различных источников к одному виду.

5. Коммуникативность: агенты могут общаться с другими агентами.

6. Мобильность: способность к передаче кода агента с одного сервера на другой.

В настоящей статье рассматриваются способы получения информации в системе World Wide Web — не просто в искусственной моделированной среде, а в той сложной, постоянно развивающейся среде, которая используется каждый день миллионами людей. Исследование будет проводиться на примере работы программных интеллектуальных агентов для поиска товаров в Интернет.

1. Постановка задачи

В данной статье проводится исследование способов получения информации о товарах в сети Интернет. Далее рассматриваются варианты создания приложений интеллектуальных агентов для поиска информации о предлагаемых интернет-магазинами товарах, определяется оптимальная архитектура и тип программы агента для приложений такого рода, и, в завершении, проводится формализация метода получения информации из Сети.

Задачи данного исследования:

1. Усовершенствовать существующие технологии по автоматическому извлечению информации из интернет-магазинов [1, 4], разработав приложение-агент со следующими характеристиками:

- работа в незнакомых средах (множество интернет-магазинов, разных по своей структуре);
- работа в долговременной перспективе, выполнение задач, растянутых во времени;
- самостоятельный анализ и сравнение предложений, выдача лишь релевантных результатов;
- невысокие требования к ресурсам клиентской машины и каналу в Интернет;
- обслуживание большого количества пользователей одновременно;

— возможность хранения индивидуальной информации;

— удобная обратная связь с пользователем.

2. Определение оптимальной архитектуры приложения для работы с интернет-магазинами, выбор типа интеллектуального агента.

3. Формализация предлагаемого метода получения информации о товарах в Интернет в виде формул алгебры предикатов первого порядка.

2. Выбор типа интеллектуального агента и определение архитектуры приложения

Разработку торгового агента можно проводить на основе одного из нескольких типов интеллектуальных агентов. Функцию агента реализует программа агента. Существует целый ряд основных проектов программ агента, соответствующих характеру явно воспринимаемой информации, которая используется в процессе принятия решения. Разные проекты характеризуются различной эффективностью, компактностью и гибкостью, а выбор наиболее подходящего проекта зависит от характера среды и предъявляемых агенту требований. Различают четыре основных типа программ агентов [1]:

- простые рефлексные агенты;
- рефлексные агенты, основанные на модели;
- агенты, действующие на основе цели;
- агенты, действующие на основе полезности.

Последние три типа интеллектуальных агентов могут также иметь способность к обучению, что намного усложняет их архитектуру, но позволяет выполнять задания в первоначально неизвестных вариантах среды и поэтому иметь более широкие сферы применения.

Простые рефлексные агенты отвечают непосредственно на акты восприятия, тогда как рефлексные агенты, основанные на модели, поддерживают внутреннее состояние, прослеживая те аспекты, которые не наблюдаются в текущем акте восприятия. Агенты, действующие на основе цели, организуют свои действия так, чтобы достигнуть своих целей, а агенты, действующие с учетом полезности, пытаются максимизировать свою собственную ожидаемую «удовлетворенность».

В зависимости от ставящихся перед системой задач можно выделить два типа приложений: приложения, работающие с кратковременными (единичными) запросами и приложения для работы со сложными долговременными запросами. Работа с кратковременными запросами аналогична работе поисковых машин и заключается в предоставлении списка обработанных страниц интернет-магазинов с интересующим пользователя товаром и только. Работа со сложными запросами, растянутыми во времени, в полной мере использует все свойства интеллектуальных агентов. В данном случае мо-

жет потребоваться самостоятельная работа агента в течение дней, недель и даже более. Также здесь необходима возможность обучения и постоянного пополнения базы знаний для более эффективной работы и логика принятия некоторых решений для достижения необходимого результата.

Исходя из сложности архитектуры и масштаба проекта, можно выделить несколько типов приложений. Самым простым и доступным (с точки зрения затрат на разработку и вычислительной мощности компьютера пользователя) вариантом реализации программы является локальная архитектура приложения. Это Win32 (Java, WinForm для .Net и так далее) приложение, которое, используя пользовательский канал в Интернет, сканирует заданные Интернет-ресурсы и возвращает список страниц с результатами поиска. Этот вариант имеет недостаток в том, что накладывает определенные ограничения на скорость и гибкость поиска, так как в данном случае не учитывается история всей предыдущей работы агента, что может понадобиться при долговременном поиске, отслеживании изменений в интернет-магазинах и хранении информации о внутренней структуре этих веб-сайтов для облегчения последующего функционирования. Для устранения этих недостатков могут использоваться различные усовершенствования клиентских приложений, такие как: добавление базы данных для хранения и анализа результатов уже обработанных запросов; использование базы знаний для осуществления самостоятельного функционирования агента в сети и выполнения различного рода долговременных задач. В базе знаний уже могут храниться не только результаты выполненных запросов, а и собственная оценка агента уже обработанных страниц, а также вся важная информация, приведенная в удобную для хранения форму.

С точки зрения распределения модулей (поискового механизма приложения и клиентской части) существует также несколько способов реализации системы. Способом, лишенным недостатка в ограниченной пропускной способности пользовательского Интернет-канала, является использование технологии Клиент-Сервер, а особенно — использование Веб-сервисов XML. В этом случае интеллектуальная часть реализуется на сервере, доступном для клиентских подключений, однако, скорость поиска уже не будет зависеть от пропускной способности клиентского канала. Еще одним достоинством данного способа является то, что функциональную часть сервера могут одновременно использовать множество клиентов.

Для использования всех достоинств интеллектуальных агентов, приложение следует строить на программе рефлексного агента, основывающегося на модели с возможностью обучения. Такой интел-

лектуальный агент с успехом справится с долговременными задачами, функционируя даже в неизвестной среде. Его свойства будут способствовать работе с множеством интернет-магазинов, структура которых до сих пор была не известна. Для того чтобы повысить эффективность работы с множеством интернет-магазинов (а не с фиксированным их количеством), приложение необходимо связать с базой данных. Такой шаг позволит хранить в базе данных всю необходимую информацию о магазинах и их структуре, чтобы не тратить время на обучение каждый раз, когда возникнет необходимость в использовании агента.

База данных будет представлена в виде онтологии, которая представляет собой формализованное описание понятий и связей между ними. Применение онтологий позволяет представить предметную область в систематизированном виде, дает возможность оперировать абстрактными понятиями в проведении рассуждений.

Также имеет смысл вынести интеллектуальную часть на сервер, а клиентскую выполнить в виде WinForms-приложения. Работая в режиме веб-сервиса, интеллектуальный агент позволит разгрузить Интернет-канал пользователя и передавать на компьютер клиента лишь результаты выполнения запросов. Многопользовательская поддержка в таком случае также является плюсом, так как здесь практически не используются клиентские компьютеры в вычислениях, а вся нагрузка приходится на предназначенный под такие задачи сервер.

3. Обзор некоторых товарных агентов в Интернет

MySimon.com. В Web доступен интеллектуальный агент «MySimon.com», который производит быстрый поиск товаров в более чем 2 тыс. онлайн-магазинов, и сравнивает цены миллионов товаров. Это веб-сайт, поисковым движком которого и есть интеллектуальный агент.

В распоряжении пользователя имеется несколько базовых функциональных возможностей: поиск товара, ограничение по категориям в выпадающем списке рядом с формой поиска. Результатом выполнения поискового запроса будет список предложений из магазинов, доступных в Сети (в базе знаний агента «MySimon» их несколько тысяч). Также доступна сортировка, различные фильтры и возможность сравнения выбранных товаров вручную.

Наличие исключительно веб-интерфейса говорит о невозможности выполнения некоторых операций, таких как: многоразовый поиск, слежение за поступлениями, уведомление пользователя об определенных событиях. Также отсутствует определенная часть интеллектуальности, заключающейся в самостоятельном анализе характеристик

товара агентом, которая перекладывает плечи пользователя.

Достоинством технологии, по которой реализован MySimon.com, можно считать реализацию интеллектуальной части на сервере, за счет чего обеспечивается высокая скорость поиска и поддержка работы в многопользовательском режиме.

PricePirates. Windows-приложение, агент для поиска товаров в Сети. PricePirates представляет собой полноценное приложение, реализованное в виде набора нескольких, связанных между собой сервисов. Под сервисами понимаются: отдельная страница гибкого поиска по аукционам, страница поиска по веб-магазинам со сравнением, поиск непосредственно по магазину Amazon.com, поиск в Веб, страница ведения статистики, популярные в eBay. Работа с Интернет-магазинами (раздел «Price comparison») осуществляется путем удаленного взаимодействия с некоторыми (участвующими в проекте «PricePirates») сайтами из каталога веб-агента «Shopping.com». Этот факт можно считать минусом данной программы. Действительно, пользователю хотелось бы иметь в распоряжении максимум выборки всех веб-магазинов Сети, а не только те, в которых заинтересованы разработчики «PricePirates». Здесь (также как и при работе с аукционами) существует возможность гибкого поиска товара, но отсутствует функция отслеживания или периодического сканирования магазинов на предмет поступления временно отсутствующей позиции.

Shopping Assistant. Это приложение-агент для работы исключительно с Интернет-магазином Amazon.com. Программа распространяется shareware (то есть бесплатно), и требует для работы Microsoft .Net Framework 1.1. Для осуществления поиска по магазину «Shopping Assistant» сканирует категории товаров, существующие в Amazon.com, после чего можно свободно по ним перемещаться и просматривать их содержимое. Также есть возможность поиска по ключевым словам, доступного из меню «Search». Полезной особенностью данной программы является возможность работы с учетными записями, совершение покупок под определенной записью и так далее. Здесь отсутствует какая-либо система для многократного поиска, автоматизации поиска, система оповещений, статистики. Другими словами: «Shopping Assistant» полностью оправдывает своё название как удобного «помощника» в совершении покупок в Интернет, не более.

4. Формализация метода обработки web информации торговым агентом

Набор отношений, характеризующих базу данных торгового агента, представим при помощи формул логики первого порядка. Пусть существует

база данных KnowledgeBase – множество знаний, тогда

$$cat_1, \dots, cat_n \in Categories,$$

$$Categories \subset KnowledgeBase,$$

где Categories – множество категорий товаров в Internet, cat – элементы множества категорий, такие, что

$$\forall cat \ cat \in Categories \wedge Name(cat, name) \wedge \wedge Synonym(cat, syn_1) \wedge \dots \wedge Synonym(cat, syn_n),$$

где Name – отношение именованности, Synonym – отношение синонима, а множество Synonyms – множество синонимов, которое содержит словарь синонимов для категорий. Здесь каждая категория имеет название Name, и у каждой категории имеется syn_1, \dots, syn_n синоним.

Знания торгового агента об интернет-магазинах представляются таким образом:

$$store_1, \dots, store_n \in Stores,$$

$$Stores \subset KnowledgeBase,$$

где Stores – множество онлайн магазинов, store – элементы множества. При этом каждый элемент store имеет адрес главной страницы url, и имеет список категорий товаров cat, которые в нем можно приобрести:

$$\forall store, \exists cat \ store \in Stores \wedge cat \in Categories \wedge \wedge Url(store, page) \wedge Contain(store, cat),$$

где Url(store, page) – отношение, при котором каждому магазину store ставится в соответствие домашняя страница магазина page, а Contain(store, cat) – отношение содержания магазином store какой либо категории товаров cat.

В общем виде метод обработки информации торговым агентом описывается следующим образом:

$$GetOffers(query) \wedge CompareOffers(offers) \wedge \wedge ShowResult(result),$$

где query – запрос пользователя, offers – найденные предложения магазинов по запросу, result – релевантные предложения, выбранные торговым агентом. Здесь получение результата товарным агентом выполняется при совершении действий GetOffers (получить страницы различных магазинов с предложением товара), CompareOffers (сравнить найденный товар и выстроить градацию предложений в соответствии с определенными предпочтениями) и ShowResult (вывод результата в соответствии с градацией).

Ниже представлено описание получения предложений онлайн магазинов GetOffers(query) по запросу query, который состоит из действий GetStores(query) – получить список магазинов и

ряда действий $\text{GetRelevantPages}(\text{store}, \text{query})$ – получить список релевантных предложений (предложений, которые соответствуют пользовательскому запросу).

$$\begin{aligned} \text{GetOffers}(\text{query}) &= \text{GetStores}(\text{query}) \wedge \\ &\wedge \text{GetRelevantPages}(\text{store1}, \text{query}) \wedge \\ &\wedge \dots \wedge \text{GetRelevantPages}(\text{storeN}, \text{query}). \end{aligned}$$

Получение списка магазинов отличается от других аналогичных решений тем, что возможен поиск магазинов в Сети, которые не находятся в базе данных посредством поисковых систем:

$$\begin{aligned} \text{GetStores}(\text{query}) &= \\ &= \text{GetStoresFromBase}(\text{query}) \vee \\ &\vee \text{GetStoresFromWeb}(\text{query}), \end{aligned}$$

где действие $\text{GetStoresFromBase}(\text{query})$ – получение списка магазинов из базы данных, формально будет выглядеть как:

$$\begin{aligned} \text{GetStoresFromBase}(\text{query}) &= \\ &= \text{ParseQuery}(\text{query}) \wedge \\ &\wedge \text{LookForRelevantShops}(\text{pquery}), \end{aligned}$$

где действие $\text{ParseQuery}(\text{query})$ означает нормализацию пользовательского запроса в удобный для агента вид (pquery), а действие $\text{LookForRelevantShops}(\text{pquery})$ означает проведение поиска по базе данных нужных магазинов в соответствии с нормализованным запросом.

Действие по поиску магазинов в сети Интернет, если в базе данных не оказалось достаточного количества магазинов:

$$\begin{aligned} \text{GetStoresFromWeb}(\text{query}) &= \\ &= (\text{ParseQuery}(\text{query}) \wedge \\ &\wedge \text{SendQuery}(\text{pquery}) \wedge \\ &\wedge \text{ParseResultPage}(\text{page}) \wedge \\ &\wedge \text{LookForRelevantShops}(\text{pquery}) \wedge \\ &\wedge \text{UpdateKnowlageBase}(\text{shops})) \vee \\ &\vee \text{NotValidQuery}(\text{query}), \end{aligned}$$

где действие $\text{SendQuery}(\text{pquery})$ обозначает отправку запроса pquery поисковой системе, действие $\text{ParseResultPage}(\text{page})$ обозначает разбор страницы, которая была сформирована поисковой системой в ответ на запрос, $\text{UpdateKnowlageBase}(\text{shops})$ значит обновление базы данных новыми знаниями об онлайн магазинах, а действие $\text{NotValidQuery}(\text{query})$ будет совершаться в случае пустого результата.

Метод получения релевантных страниц $\text{GetRelevantPages}(\text{store}, \text{query})$:

$$\begin{aligned} \text{GetRelevantPages}(\text{store}, \text{pquery}) &= \\ &= \exists \text{page } \text{page} \in \text{Pages} \wedge \\ &\wedge \text{Relevant}(\text{page}, \text{pquery}) \wedge \\ &\wedge \text{Offer}(\text{page}), \end{aligned}$$

где существует хотя бы одна страница page , которая принадлежит множеству страниц Pages , является релевантной нормализованному запросу пользователя $\text{Relevant}(\text{page}, \text{pquery})$ pquery и которая предоставляет возможность совершить покупку $\text{Offer}(\text{page})$.

Алгоритм работы торгового агента представляется следующим образом:

- 1) Получение запроса пользователя.
- 2) Нормализация запроса – синтаксический разбор запроса и его представление с помощью определений, описанных в базе данных, как базовые.
- 3) Интеллектуальный поиск магазинов:
 - а. Обращение к базе данных, поиск магазинов соответствующей направленности в случае, если категория товара была определена агентом автоматически из запроса.
 - б. Расширение базы данных путем поиска магазинов соответствующей направленности в Сети с помощью поисковых систем, в случае, если знаний недостаточно.
 - в. Если агент не обладает дополнительными знаниями о товаре, осуществляется сразу поиск по страницам в магазинах, которые находятся в базе данных (функционал пункта 4).
 - д. Если страниц найдено не было – переход к поиску магазинов в Интернет с помощью поисковых систем.
 - е. В случае успеха – разбор найденных страниц и сохранение знаний о товаре и линейке товаров для ускорения последующего поиска.
- 4) Выбор страницы с предложением в найденных Интернет-магазинах.
- 5) Разбор страницы, выделение характеристик товара.
- 6) Сохранение полученных описаний товара в нормализованной форме.
- 7) Сравнение товара по критериям, очевидным для агента, либо по критериям, которые выбрал пользователь, сортировка.
- 8) Выдача отсортированного результата.

Преимуществами данного алгоритма перед алгоритмами других торговых агентов (*MySimon.com*, *PricePirates*, *ShoppingAssistant* и т.д.) и алгоритмами, которые предлагаются [1,4], являются: гибкий подход к выбору магазинов – самостоятельный поиск таковых в сети; интеллектуальный поиск, благодаря которому возможен не простой перебор страниц существующих онлайн магазинов, а выбор таких магазинов, в которых наличие данного товара наиболее вероятно.

Выводы

В результате исследований была разработана архитектура программы интеллектуального агента для осуществления покупок в Интернет. Также был предложен усовершенствованный алгоритм работы такого интеллектуального агента. Предло-

женное формальное описание с помощью алгебры предикатов первого порядка данного агента в виде метода обработки информации торговым агентом. Разработанный метод и алгоритм торгового агента отличается от аналогов [1,4] тем, что позволяет агенту более эффективно использовать вычислительные ресурсы, затрачивая на поиск меньше времени, благодаря усовершенствованному подходу в поиске магазинов. В данной программе агент вначале анализирует запрос и на основании своих знаний может спрогнозировать, в каких магазинах наиболее вероятно наличие такого товара. Если же такой прогноз невозможен, то агент возвращается к традиционному способу поиска товаров – к перебору страниц с предложениями по цепочке категорий. Еще одним нововведением можно считать возможность самостоятельного поиска агентом новых магазинов в сети Интернет посредством поисковых систем. Таким образом, расширяется база данных агента, что повышает эффективность его дальнейшей работы при повторном обращении к этим магазинам.

Список литературы: 1. *Рассел С., Норвиг П.* Искусственный интеллект: современный подход, 2-е изд.. Москва: Издательский дом «Вильямс», 2006. 1408 с. 2. *Люгер Д.Ф.* Искусственный интеллект: стратегии и методы решения сложных проблем, 4-е изд.. Москва: Издательский дом «Вильямс», 2003. 864 с. 3. *Тарасов В.Б.* От многоагентных систем к интеллектуальным организациям: философия, психология, информатика. – М.: УРСС, 2002.

4. *А.А. Морозов, Ю.В. Обухов.* Разработка математического аппарата логического программирования интеллектуальных агентов интернет // Журнал радиоэлектроники № 10, 2003. 5. *Ю.В. Рогушина* Использование онтологического описания предметной области для повышения релевантности информационного поиска // Проблемы программирования (4). 2003.

Поступила до редколегії 15.09.2008

УДК 519.6

Збір і обробка інформації в Інтернет з використанням інтелектуальних агентів / В.В. Дудар, М.В. Збітнева, А.Г. Дерев'янку // Біоніка інтелекту: наук.-техн. журнал – 2008. – № 2 (69). – С. 61-66.

Дана стаття присвячена розгляду питань розробки елементів агентної технології, таких як методи та алгоритми. Запропоновано удосконалений метод обробки інформації в Інтернет торговим агентом, реалізацію даного методу у вигляді алгоритму, архітектуру програмного застосування.

Бібліогр.: 5 найм.

УДК 519.6

Collection and treatment of information in internet with the use of the intellectual agents / V. Dudar, M. Zbitneva, A. Derevyanko // Bionics of Intelligence: Sci. Mag. – 2008. – № 2 (69). – P. 61-66.

This article is devoted to consideration of questions of development of elements of agent technologies, such as methods and algorithms. An improved method of information treatment by the commercial agent in Internet, realization of this method as an algorithm, architecture of program appendix are offered.

Ref.: 5 items.