ИСПОЛЬЗОВАНИЕ АССОЦИАТИВНОГО МЕТОДА ПОИСКА СЛОВ В ПРОЦЕССЕ ДИСТАНЦИОННОГО ОБУЧЕНИЯ

А.А. Маталыга, О.В. Герман

Белорусский государственный университет информатики и радиоэлектроники, Минск, Беларусь, kadosha1@rambler.ru

Abstract. This article presents an original method of finding information, using methods and associative binary search. The basic methods and algorithms for information retrieval are described.

Конец XX - начало XXI века характеризуется стремительно растущим потоком разнообразной информации, представляющей интерес для самых широких слоев социума. Более того, Интернет - технологии и программно-технические средства, также доступные большинству людей, позволяют осуществлять процесс поиска информации в любое время, практически в любом месте, по любым запросам.

Поиск – это процесс, в ходе которого в той или иной последовательности производится соотнесение найденного объекта с хранящимся в памяти массива эталоном. Цель любого поиска заключается в необходимости или желании найти ту или иную информацию, сведения, справку и т.д. для повышения собственного профессионального, культурного или любого другого уровня; создания новой информации или формирования новых знаний; принятия управленческих решений и т.п.

Обычно пользователи не имеют исчерпывающих знаний об информационном содержании ресурса, в котором проводят поиск. Пользователи нередко ошибаются, набирая поисковый запрос. Например, пропускают или добавляют лишние буквы, пишут слова с орфографическими ошибками, пишут слова разговорным языком (включая слэнг), пишут слова не переключив клавиатуру на нужный язык.

Таким образом, следует учитывать при разработке алгоритма поиска типичные ошибки в поисковых запросах.

Рассмотрим основные методы поиска информации в Интернете, используемые по отдельности или в комбинации друг с другом.

Прежде всего, это использование поисковых машин. В результате поиска появляется список ресурсов Интернета, который необходимо детально рассмотреть. Метод поисковых машин основан на использовании ключевых слов, передающихся поисковым серверам в качестве аргументов поиска.

В качестве метода рассматривается и непосредственный поиск с использованием гипертекстовых ссылок. На основании того, что все сайты в пространстве Всемирной паутины фактически связаны между собой, поиск информации может быть произведен путем последовательного просмотра связанных страниц с помощью браузера. Этот способ Web-страниц часто оказывается единственно возможным на заключительных этапах информационного поиска, когда механическое исследование уступает место более глубокому анализу.

Поиск с применением специальных средств – это полностью автоматизированный метод, весьма эффективный для проведения первичного поиска. Данный метод заключается в применении специализированных программ – спайдеров, которые в автоматическом режиме просматривают Web-страницы, отыскивая на них искомую информацию с помощью гипертекстовых ссылок. Этот метод является особо

эффективным в том случае, если использование поисковых машин не дает необходимых результатов в силу нестандартности запроса, либо других причин.

Еще один метод – анализ новых ресурсов поиска наиболее свежей информации, либо анализ тенденций развития объекта исследования в динамике. Большинство поисковых машин обновляет свои индексы со значительной задержкой, вызванной гигантскими объемами обрабатываемых данных, и это упущение наиболее четко прослеживается по наименее популярным темам.

Очевидно, что поиск информации в Интернете является больше процессом решения поисковой задачи, стоящей перед пользователем, а не просто нахождением релевантной запросу информации [1].

Как мы знаем, в ассоциативной памяти поиск реализуется аппаратно путем параллельного сравнения слова-эталона (ключевого слова) со всеми записанными словами [2]. Для этого каждый набор элементов хранения программных объектов дополняется схемами сравнения. При схемной реализации ассоциативной памяти доступ к данным осуществляется очень быстро, их поиск по любому фрагменту не представляет труда, если этот фрагмент точно определен. Но, если фрагмент был видоизменен (написан с ошибкой или опечаткой), то ассоциативный поиск в таком случае будет безрезультатен. В этой связи будет целесообразно использовать возможности бинарного поиска, который может осуществлять поиск не точно заданного ключа.

На основании выше изложенного анализа можно предположить, что оптимальным будет поиск, который соединит в себе ассоциативный поиск и поиск по дереву. Дерево дает возможность использования ASCII кода. Буквы кодируются в соответствии с кодовым набором ASCII, для их представления и поиска используются 5 младших бит кода [3].

В предлагаемом нами методе используется аналог метода динамики средних (идея нивелирования влияния случайных отклонений при ошибках в записи ключей), полученный список поиска дает вероятностные результаты (определяемый документ не обязательно тот, поиск которого задумал клиент сайта).

Рассмотрим более детально алгоритм поиска. Сначала подсчитывается среднее значение ASCII-кода запроса. Далее сравнивается среднее значение запроса с элементом массива, т.е. сравнивается значение массива с диапазоном среднего значения ключа. Первая запись входной последовательности сопоставляется с диапазоном значений корня дерева.

Для каждой следующей записи ключ сначала сравнивается с диапазоном значений ключа корня, т.е. входит ли ключ записи в диапазон значений ключа корня дерева. Если он меньше чем диапазон значений ключа корня, то далее он сравнивается с диапазоном значений ключа правого потомка и т.д. до тех пор, пока потомок не будет отсутствовать. Место отсутствующего потомка занимает новая вершина, с которой сопоставляется очередная запись.

Данные действия повторяются до тех пор, пока не будет просмотрена вся входная последовательность записей.

Поиск считается успешно завершенным, если ключ искомого элемента входит в диапазон значений узла. Если поиск завершается неудачей, т.е. ключ не вошел в диапазоны, приписанные узлам дерева, то выбираем тот узел, где степень близости значений оказалась наибольшей.

Рассмотрим алгоритм поиска на примере (рисунок 1). В строку запроса введено слово «грепп». В начале осуществляется поиск введенного слова (ключевого слова) путем параллельного сравнения со всеми хранимыми в памяти словами. Поиск по

ключу оказался безрезультатным, далее поиск автоматически продолжается по дереву. Слово «грепп» имеет ключ 1174, среднее значение ASCII-кода запроса - 234,8. Сравнивается среднее значение запроса с элементом массива, т.е. идет сравнение значения массива с диапазоном среднего значения ключа. Соответственно производится поиск диапазона значений ключа по дереву. Первая запись входной последовательности сопоставляется с диапазоном значений корня дерева.

Для каждой следующей записи ключ сначала сравнивается с диапазоном значений ключа корня, т.е. входит ли ключ записи в диапазон значений ключа корня дерева. Если он меньше чем диапазон значений ключа корня, то далее ключ сравнивается с диапазоном значений ключа правого потомка и т.д. до тех пор, пока потомок не будет отсутствовать. Место отсутствующего потомка занимает новая вершина, с которой сопоставляется очередная запись.

В рассматриваемом примере ответом будет узел p4 с диапазонам значений (230; 240) в который попадает ключ искомого слова.

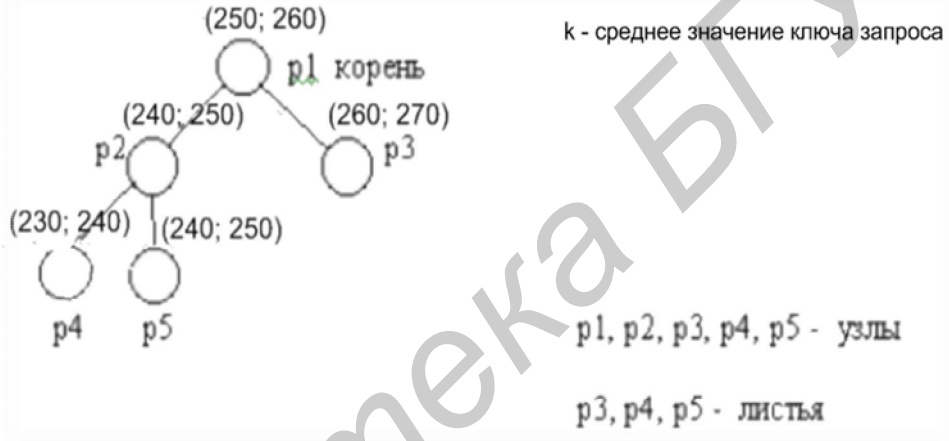

Рисунок 1 – Пример алгоритма поиска

Таким образом, соединение ассоциативного поиска с поиском по дереву позволит улучшить алгоритм ассоциативного поиска, что, в свою очередь, приведет к уменьшению затрат времени на получение необходимого пользователю объема информации.

Представленный нами программный продукт может быть использован в системе дистанционного образования БГУИР. Например, при проведении интерактивных консультаций со студентами дистанционной формы обучения.

Литература

1. Современные методы поиска информации Электронный ресурс]. – Электронные данные. – Режим доступа: http://poisk.swsu.ru/opis-poisk/problem/63-sovremen-metod.html, свободный.

2. Кохонен, Т. Ассоциативная память / Т. Кохонен – М.: Мир, 1980. – 240 с.

3. Прохождение и поиск по бинарным деревьям [Электронный ресурс]. – Электронные данные. – Режим доступа: http://rk6.bmstu.ru/electronic_book/posapr/zadanpo/bintree.htm, свободный.