# РАЗВИТИЕ КОРПОРАТИВНОЙ ПОИСКОВОЙ СИСТЕМЫ В КОНТЕКСТЕ ЭВОЛЮЦИИ СОВРЕМЕННЫХ ТЕХНОЛОГИЙ ПОИСКА ИНФОРМАЦИИ

**Потапенко В.А.**
кафедра ЭВМ ДонГТУ
eline@cs.dgtu.donetsk.ua

***Abstract***

***Potapenko V. The development of an internal corporate search system and evolution of modern searcli technologies.*** *The technologies of development modern search systems were examined. The investigations in area ofproductivity and functionality of the search system ofDonSTU were carried out.*

***Введение***

В статье на базе анализа современных поисковых технологий и текущего их развития формулируются основные требования к построению корпоративной поисковой системы и рассматриваются некоторые особенности ее реализации на примере внутренней поисковой системы ДонГТУ.

Современные поисковые системы постепенно переходят ту границу, когда при поиске информации осуществлялось примитивное посимвольное сравнение искомой комбинации слов с текстом документа. При больших объемах информации (а сейчас крупнейшие поисковые службы насчитывают более 1 миллиарда уникальных ссылок) такой метод поиска оказывается чрезвычайно неэффективным. Кроме того, уровень структуризации информации в Интернет очень низок, а частота обновления достаточно высока. Проблема наполнения, стоявшая вчера, трансформировалась в проблему поиска открытых и бесплатных источников, погребенных в недрах колоссальной, запутанной гипертекстовой среды. Традиционные поисковые механизмы не справляются с задачей индексирования и даже не в состоянии представить имеющиеся данные в упорядоченном виде. За последнее время появилось несколько новых поисковых технологий, использующих различные методики для повышения релевантности выдаваемых результатов. Рассмотрим принципы организации и работы некоторых из них.

***Обзор и анализ существующих поисковых технологий***

DirectHit. Логика работы данного поискового механизма основана на анализе предпочтений пользователей, ранее осуществивших выбор той или иной ссылки на подобный запрос. Служба анализирует поведение миллионов людей, ежедневно обращающихся к различным поисковым узлам, и для каждого запроса фиксирует наиболее часто используемые ссылки. Учитывается также количество времени, проведенное человеком за изучением содержимого Web-страниц, скрывающихся за ссылками. Чем оно больше, тем выше становится значение релевантности ресурса. Имеется функция отслеживания корреляции и связи между различными запросами, так что, сформулировав запрос, посетитель получает набор связных тем, которые он тоже может просмотреть, расширив ареал поиска.

разных социальных групп. Пользователь, подписавшись на услуги Personalized Search, сначала заполняет анкету, в которой указывает свой пол, место проживания, род занятий и прочие сведения. Теперь система сможет предложить ему ссылки, заинтересовавшие других людей с похожими анкетными данными. Например, для жителя Европы слово "motorsport" ассоциируется с чемпионатом Formula-l, адля жителя США с серией CART и соревнованиями NASCAR.

Индексная база службы пополняется с помощью робота под названием Grabber. При обновлении применяется избирательная стратегия, согласно которой сайты, получившие наибольшее количество переходов с результатов поиска, просматриваются чаще (раз в неделю), чем все остальные ресурсы (полное обновление раз в месяц). Производительность аппаратного кластера, поддерживающего функционирование робота, позволяет индексировать до 10 миллионов сайтов в день.

Google. Данная система представляет собой новое поколение в области построения мощных распределенных поисковых механизмов. Система оказалась настолько мощной, эффективной и, главное, достаточно дешевой в реализации, что даже такой гигант, как Yahoo подписал соглашение о сотрудничестве с Google. Суть работы поискового механизма заключается в следующем. Специалисты взяли на вооружение общеизвестную систему "оценки ценности" статей, принятую в мировом научном сообществе: рейтинг статьи есть производная от количества сделанных цитат и ссылок на нее в других научных публикациях. Google высчитывает релевантность документа, попавшего в результаты поиска, в соответствии с количеством ссылающихся на него других Web-страниц. "Старинные" бумажные принципы оказались действенными и в Internet.

Цитируемость документа выводится Google с использованием системы PageRank. Значение PageRank любого документа учитывает количество ссылок на него во всех прочих проиндексированных источниках и вычисляется по формуле:

$$PR(A) = (1-d) + d\,(PR(T1)/C(T1) + ... + PR(Tn)/C(Tn)), \quad (1)$$

где A - оцениваемый документ,
PR(A) - рейтинг документа A,
C(A) - общее количество ссылок со страницы A,
T1. ..Tn — документы, ссылающиеся на A,
d - некий фактор случайности, описывающий поведение посетителей.

Итак, PR(A) представляет собой вероятность попадания хаотически путешествующего по Internet пользователя на страницу A. Величина d, которую разработчики Google установили равной 0,85, характеризует вероятность того, что, находясь на странице, участвующей в оценочной формуле, пользователь решит перейти на произвольную страницу в Internet, путем набора URL прямо в соответствующем поле броузера. Как видно из формулы, "рекомендация" от страницы, имеющей высокую "репутацию", обладает большим весом, что позволяет правильно оценивать значимость непопулярных, но качественных сайтов. В эту схему хорошо укладываются запросы, состоящие из одного слова, а в случае нескольких заданных терминов приходится учитывать и другие факторы. Например, оценка близости искомых слов в документе выбирается из десяти дискретных значений, начиная от совпадения фразы и заканчивая "очень далеко". Порядок слов в запросе не играет роли для Google. Система также активно использует индексирование по ссылкам.

Сбором данных в системе занимаются несколько независимых роботов, получающих задание от URL-сервера, коллекционирующего ссылки. Найденные документы архивируются и помещаются в репозиторий, далее формируется три индекса страниц: по словам, документам и ссылкам.

Clever. Система разрабатывается в исследовательском центре Almaden корпорации IBM. Clever, как и Google, в своей работе основывается на ссылках и рейтингах, но подходит к задаче совсем по-другому. Если в Google сначала вычисляются коэффициенты PageRank для всех индексированных документов, а потом просто учитывает их при сортировке результатов, то поисковая система IBM оценивает страницы на ходу. Сначала выполняется обыкновенный поиск по терминам заданного запроса. Отобранные страницы просматриваются, по ссылкам выделяется новая порция документов. Их тоже просматривают на предмет связей. И так далее итерация за итерацией. Согласно последним исследованиям центра, 96% документов, связанных по ссылкам, имеют сходную тематику. После того как определенная часть структуры выявлена, Clever высчитывает рейтинг для каждой из найденных страниц на основании количества ссылающихся на нее документов. Система различает два типа сайтов: "первоисточники" (authorities) и "хабы" (hubs). Ценность первых - контент, вторых - ссылки на многочисленные "первоисточники". Сайты - хабы часто оказываются более полезными, чем непосредственно поставщики контента, поскольку зачастую предлагают более широкий взгляд на тему поиска. Действительно, сегодня пользователь зажат в рамках собственного запроса: обобщенные термины дают слишком много результатов, а узкоспециализированные - слишком мало. В отличие от Google, ориентированной именно на узлы "первоисточники", Clever отдает должное "хабам". Благодаря разветвлению поиска "вширь" удается выявлять тематические сообщества сайтов, число которых достигает 100 тыс. Тут просматриваются интересные аналогии с Internet - каталогами, редактируемыми людьми. Ни один из них не может справиться с экспоненциальным ростом Internet, и даже Yahoo индексирует всего около 1 миллиона страниц. Таким образом, Clever сочетает преимущества традиционных поисковых машин и каталогов.

InfraSearch. Девизом создателей данной поисковой системы является «динамический поиск в режиме реального времени». Чтобы понять механизм работы данной системы, следует подробнее рассмотреть проблему нахождения нужной информации с точки зрения владельца сайта. К примеру, для крупного информационного ресурса, опубликовавшего статью, скажем, о визите Папы Римского, чрезвычайно желательно, чтобы при поиске по ключевому слову "Папа Римский" читатель получал ссылку на недавно опубликованную статью, а не на некий материал, размещенный на сайте полгода назад, однако содержащий большее количество ключевых слов и поэтому появляющийся в списке ссылок первым. Аналогично продавцу автомобилей хотелось бы, чтобы клиент, ищущий информацию о новой модели, попадал непременно на страницу, где, по мнению автора сайта, представлена искомая информация. InfraSearch для подобных целей предлагает владельцу сайта специальный продукт, с помощью которого поиск можно было бы "направлять" вручную, причем делать это в режиме реального времени, т. е. нужную информацию клиенту выдавать сразу после его запроса к поисковой системе.

В случае InfraSearch релевантность (которая настраивается "вручную") дополняется динамичностью контента, что особенно важно для быстро обновляемых сайтов - пользователь в течение дня может получить различные

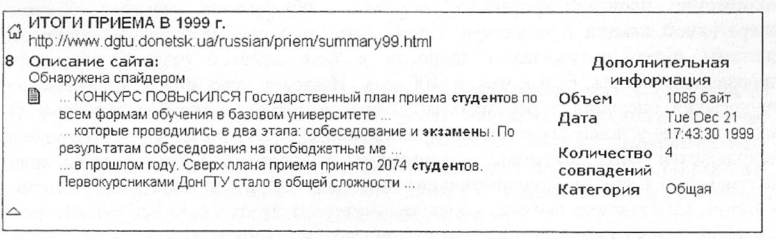
результаты на один и тот же запрос, в зависимости от того, как скоро появляется информация в данной сфере.

### *Разработка и исследование поисковой системы внутренней сети ДонГТУ*

Разрабатываемая поисковая система для внутренней сети ДонГТУ относится к классу небольших систем с количеством индексированных страниц не более 10 тысяч. В связи с этим использование даже упрощенной методики поиска дает хорошие результаты. Тем не менее, система обладает рядом отличий от систем подобного уровня. Обычно при поиске страниц с заданным набором ключевых слов поисковые системы анализируют на совпадение искомое ключевое слово со словом из базы проиндексированных страниц. При этом обязательньм является условие, чтобы до и после данного слова стоял разделитель (пробел, запятая и т.д.). Данный метод является достаточно неэффективным. Например, если пользователь захочет найти документ со словом «студент», то система не выдаст ссылки на документы, в которых слово «студент» имеет окончание. Кроме того, очень часто ограничен набор логических конструкций, при помощи которых можно точнее конкретизировать запрос. Разрабатываемая поисковая система лишена перечисленных выше недостатков. Среди ее возможностей можно выделить следующие:

- полнотекстовая индексация - в базе данных документ хранится полностью;
- поддержка HTTP proxy;
- поддержка базы данных MySQL;
- поддержка логических конструкций при поиске;
- автоматическая индексация новых ресурсов при помощи робота.

Сильной стороной данной поисковой системы является возможность поиска по корням искомых слов. Например, при запросе «студент экзамен» система выдаст ответ следующего вида:




Возможность задания логики поиска «И» или «ИЛИ», а также способность игнорировать регистр символов при поиске способствуют получению более точных результатов. Если пользователь при поиске хочет указать строгую последовательность ключевых слов, достаточно заключить в двойные кавычки искомое выражение. Поддерживаются даже конструкции следующего вида: слово_1 «слово_2 слово_3» слово_4 «слово_5 слово б», т.е. имеется возможность обрабатывать запросы, состоящие из вложенных конструкций из двойных кавычек. Таким образом, можно составить достаточно сложный запрос, который даст желаемый результат.

Одной из основных проблем при построении поисковой системы является проблема производительности. На этапе разработки система работала на базе персонального компьютера под управлением ОС Linux (частота процессора 333 MHz, объем оперативной памяти 32Mb). Для сравнения: система Google построена на базе 4000 персональных ЭВМ с ОС Linux, каждый из которых имеет 256Mb оперативной памяти. Именно объем ОЗУ является одним из определяющих факторов производительности системы. Основным потребителем системных ресурсов является система управления базой данных (СУБД). В качестве таковой была выбрана MySQL, как одна из самых быстрых и наименее ресурсоемких СУБД. На рисунке 1 показана полученная экспериментальным путем зависимость времени работы поисковой системы от объема проиндексированных страниц.

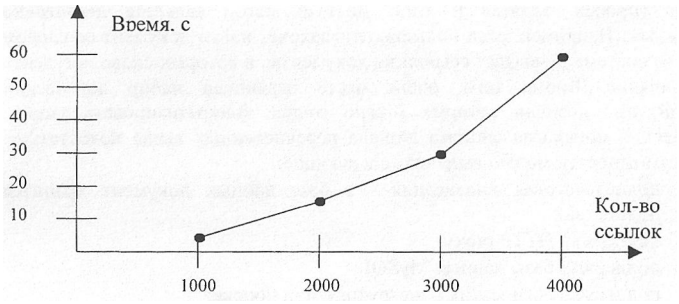



Рисунок 1. - Зависимость времени поиска от количества проиндексированных ссылок

Большое время работы поисковой системы при относительно небольшом количестве проиндексированных ссылок объясняется малым объемом оперативной памяти компьютера. Кроме того, во время разработки поисковой системы путем оптимизации запросов к базе данных удалось увеличить производительность более чем в 100 раз. Известно, что одной из наиболее трудоемких операций в работе базы данных является выборка с условием. В нашем случае условие было заменено хеш массивами, в которых в качестве ключа используется URL документа, а значением в одном массиве является тело документа, а в другом дополнительная информация (размер, дата и т.д.). Таким образом, работа с «условиями» перешла к интерпретатору языка Perl, на котором написана поисковая система, что привело к значительному повышению производительности. Еще одним способом повышения скорости работы базы данных, а, следовательно, и всей поисковой системы в целом является использование индексов таблиц. Для того чтобы понять, каким образом это влияет на время выборки, рассмотрим, как работает типичная реляционная база данных. Чтобы получить ответ на любой запрос из таблицы, не имеющей индекса, СУБД вынуждена сканировать таблицу, т.е. считывать в ней каждую строку. Очевидно, что в случае больших таблиц такой процесс слишком расточителен для сервера. Индексы обеспечивают механизм указателей на требуемые данные. Индекс в базе данных работает аналогично индексу справочника. Как и в книге, индекс в базе данных представляет собой список «важных» значений, которым соответствуют страницы в таблице базы данных. Для выборки информации

данных представляет собой список «важных» значений, которым соответствуют страницы в таблице базы данных. Для выборки информации считывается обычно меньший, чем вся таблица, список страниц индекса, которые в свою очередь указывают на данные, необходимые для ответа на любой запрос [1].

Индекс является механизмом организации данных в таблице для их оптимальной выборки («оптимальная» в данном случае означает «максимально быстрая»). Индексы представляют собой наборы уникальных для данной таблицы значений и соответствующий им список указателей на страницы данных, где эти значения находятся в таблице физически.

Индексы позволяют ускорить выборку из таблиц записанной в базе данных информации. Они являются объектами базы данных и так же, как и таблицы, нуждаются в месте для хранения. Подобно таблицам, требующим страниц для хранения своих строк, индексы требуют страниц, чтобы сохранить свою информацию. Достоинством индексов является то, что, как правило, они позволяют уменьшить количество запросов ввода/вывода, необходимых для выборки данных из таблицы.

### *Перспективы развития поисковой системыДонГТУ*

Следующим этапом в построении поисковой системы ДонГТУ будет дальнейшее увеличение ее производительности и повышение релевантности выдаваемых результатов. Механизмом для этого будет служить реализация идеи, заложенной в поисковую систему Google. Как уже было сказано выше, Google использует PageRank для определения «веса» конкретной ссылки, вычисленной по формуле 1. Планируется использовать несколько упрощенный вариант данной системы. Суть его заключается в следующем. При очередном индексировании БД для каждой страницы формируется список ссылок, которые ссылаются на данную страницу. Таким образом, эти ссылки как бы отдают «свой голос» за данный ресурс. На первом этапе каждый «голос» будет иметь одинаковое фиксированное значение, и результаты поиска будут формироваться с учетом количества ссылок, проголосовавших за данный ресурс. В итоге планируется в несколько раз увеличить релевантность выдаваемых результатов.

### *Выводы*

В данной статье выполнен обзор и анализ современных технологий поиска информации в Интернет. Были рассмотрены основные тенденции развития поисковых технологий и проблемы, стоящие перед разработчиками высокотехнологичных поисковых систем. В этом контексте описана разработка поисковой системы для внутренней сети ДонГТУ. Отмечены ее достоинства, недостатки и пути ее дальнейшего усовершенствования.

### *Литература*


1. Тихомиров Ю.В. Microsoft SQL Server 7.0. СПб.: БХВ, 1999. - 720 с.
2. Yahoo! предпочла Google. http://www.itc.kiev.ua/article.phtml?ID=2883
3. Поиск в Internet: новые методики. http://www.itc.kiev.ua/article.phtml?ID=2081
4. Google: путь к вершине. http://www.itc.kiev.ua/article.phtml?ID=3424
5. Google: определение релевантности по рейтингу популярности, http://www.alldollars.bizland.com/searchengine/articles/google_lp.htm.