Введение в поисковые системы

Ю. Лифшиц*

3 ноября 2005 г.

План лекции

1. Архитектура поисковых систем
2. Алгоритмы поисковых систем
3. Поисковая оптимизация

1 Архитектура поисковых систем

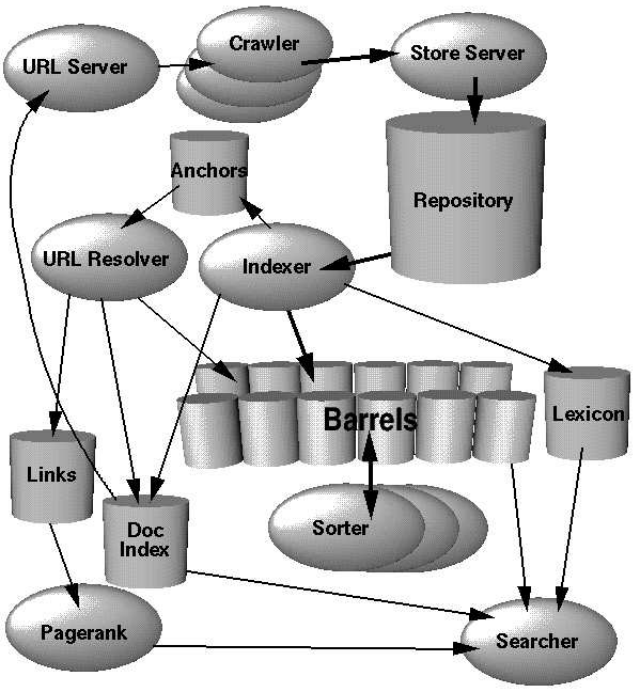
1.1 Анатомия поисковой системы

В любой поисковой системе можно выделить три базовых части:

1. Робот (краулер, спайдер, индексатор). Робот отвечает за сбор информации. То есть робот эмулирует работу пользователя, загружая страницы и сохраняя их в базе данных.
2. База данных. В базе данных хранится и сортируется собранная роботом информация.
3. Клиент. В этой части обрабатываются пользоватльские запросы. В действительности клиент может быть разнесен по нескольким физически несвязанным компьютерам. Однако, чтоит отметить, что все эти компьютеры должны иметь доступ к базе данных.

Рассмотрим описанную выше структуру на примере поисковой системы Google.

*Законспектировал Д. Кочелаев.



1. URL Server — список всех адресов.
2. Crawler — робот, который загружает страницы из списка адресов и передает в Store Server.
3. Store Server сохраняет страницы в Repository, чаще всего в виде HTML документа. При этом вся дополнительная информация, такая как картинки, flash-анимация и прочее, не сохраняется.
4. Indexer разбирает сохраненные в Repository HTML-документы в последовательности слов и сохраняет их в Barrles (база данных).
5. Lexicon — список всех слов. Чаще всего слова храняться в таблице с двумя полями “номер” и “слово”. Таким образом достигается экономия места в базе данных, так как длинные слова заменяются достаточно коротким номером.
6. Anchors — выделенные компонентом Indexer ссылки (URL).
7. URL Resolver — обработчик URL. Если находятся новые ссылки, то они передаются в URL Server.
8. Links определяет какие сайт на какие ссылаются и передает это в PageRank.

9. PageRank — определяет рейтинг сайта, основным критерием является количество ссылок на этот сайт (подробнее смотрите раздел про PageRank).

10. Searcher — клиент. Чаще всего клиент пользуется статической базой данных, которая обновляется примерно раз в сутки.

1.2 Прямой и обратный индекс

Рассмотрим варианты хранения записей в базе данных, то есть параметр, по которому они отсортированы, и какая дополнительная информация храниться для каждой записи. В связи с этим появляются два поняти прямой и обратный индексы.

В случае прямого индекса записи отсортированы по номеру документа. Для каждой записи хранится отсортированный по номеру список слов. Для каждого слова хранятся первые несколько (например восемь) позиций вхождения слова в докумен, количество вхождений и формат вхождения. Под форматом вхождения подразумевается вхождение слова в тексте ссылки, в описании к картинке, в загаловке и т.д., такие слова будут иметь приоритет при поиске. Прямой индекс обновляется постоянно при работе робота, соответственно возникает вопрос: “Как часто надо индексировать одну страницу?”. Для каждой страницы в базе данных харнится частота предполагаемого обновления, которая считается следующим образом: при очередном заходе робота на эту страницу в случае отсутствия обновлений частота увеличивается в два раза, а если страница за этот период времени менялась, то уменьшается. Также стоит отметить, что чаще всегоробот индексирует не все слова из документа (например только первую тясячу слов) и не все документы с одного сайта.

Обратный индекс используется клиентом при поиске. В этом случае для записи отсортированы по словам. Для каждой записи хранится номер слова, список документов, в которые входт это слово, и полная информация о вхождении. Отметим, что обратный индекс обновляется не так часто как прямой, а примерно раз в сутки.

1.3 Релевантность и позиция документа в поиске

Для начала дадим определение релевантности. Под релевантностью понимается то, насколько слово (совокупность слов) соответствует данному документу.

Рассмотрим теперь характеристики, которые могут влиять на позицию документа в списке ответов:

1. Наличие слов в документе. Очевидно, что если слова в документе не встречаются, то данный документ не подходит под условия поиска.

2. Частота вхождения слов. Чем чаще слова встречаются на странице, тем выше документ окажется в списке поиска.

3. Форматирование слов. То есть если в документе слова встречаются выделяющих тэгов, заголовков, описаний картинок, то такой документ будет иметь более высокий приоритет.

4. В случае набора слова значение также имеют расстояние между этиим словами в документе и их порядок.

5. В некоторых поисковых система (например Яндекс) значение имеет морфологическое вхождение слов, то есть падеж (род, лицо), в котором слово входит в документ.

6. Количество и уважаемость ссылок (подробнее смотрите раздел про PageRank).

7. Регистрация в каталоге поисковой системы. Это очень важная характеристика, так как каталоги составляются вручную и в них уже заданы разделы и тематика страницы.

1.4 Работа клиента

Сначала запрос разбивается на слова. Далее удаляются так называемые "стоп" слова — слова, которые встречаются почти во всех документах (предлоги, союзы). На следующем шаге каждому слову сопоставляется его номер из "лексикона". Для каждого слова из запроса находится в обратном индексе список документов, которые содержат это слово. Из этих списков создается новый, содержащий те и только те документы, которые входили в списки для всех слов. Затем, на основе характеристик обозначеных в предыдущем разделе для каждого документв вычисляется степень релевантности, и список сортируется по этому признаку. На этом шаге для всех документов создаются аннотации. Аннотацией может быть содержание тэга "description", контекст вхождения слов из запроса (наиболее близко стоящих или первое вхождение), первое предложение или заголовок документа.

1.5 Качество поиска

Имея несколько поисковых систем, хочется определить какая же из них лучшая. То есть оценить качество поиска. Как это сделать? На этот вопрос нет однозначного ответа. Однако, можно привести некоторые возможные критерии определения качества поиска:

1. Полота поиска, то есть насколька большая доля страниц, из удовлетворяющих условиям поиска, была найдена. Обычно у современных поисковых машин с этим критерием проблем не возникает. Хотя количество документов в базе данных у поисковых машин отличается, для сравнения база данных Google содержит около 8 миллиардов документов, а — Яндекса только около 10 миллионов. Однако, в базе данных Яндекса есть документы из русскоязычного интернета, отсутствующие в базе данных Google.

2. Точность поиска, то есть доля релевантных документов из всех найденных.

3. Benchmarks: показатели системы на контрольных запросах и специальных коллекциях документов. С этим критерием у некоторых поисковых систем возникают проблемы, например некоторое время назад из четырех крупных поисковых систем по своему имени себя находила только одна (остальные безусловно выдавали ссылку на себя, но далеко не самой первой). Также проводятся ежегодные соревнования поисковых алгоритмов.

4. Оценка экспертов.

Также стоит отметить, важность производительности поисковых алгоритмов, так например Google, при убыстрении своих поисковых алгоритмов на 1%, смог бы уменьшить свой парк компьютеров на 100 штук (на данный момент у них около 10 тысяч компьютеров).

2 PageRank

2.1 Постановка задачи

При выборе из списка документов, найденных по некоторому запросу, хотелось бы выбрать наиболее подходящие. Таким образом встает задача о вычислении "качества" документа. Первые идеи, которые могут прийти в голову:

1. Учет частоты обновления страницы.

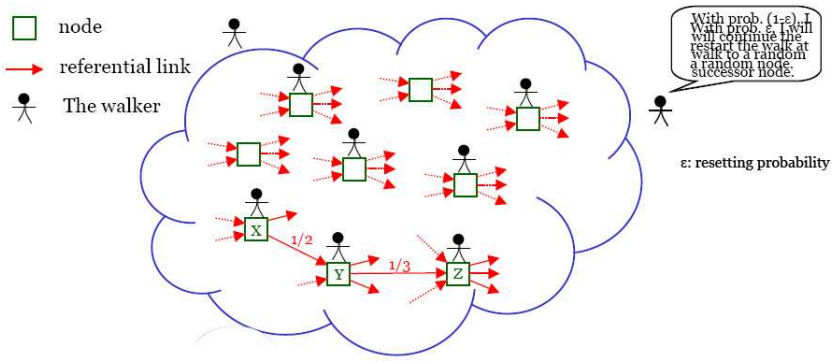
2. Учет количества посещений страницы.

3. Учет регистрации в каталоге данной поисковой системы.

К сожалению, эти характеристики легко подделать (пожалуй за исключением регистрации в каталоге). Рассмотрим выдвинутую в 1998 году Брином идею определения рейтинга страницы через количество ссылающихся страниц и их рейтинг. Как будет показано далее, "накрутить" данный показатель достаточно тяжело.

2.2 Модель случайного блуждания

Для определения PageRank для страниц промоделируем хождение пользователя по сети. Для этого воспользуемся моделью случайного блуждания. Представим себе сеть в виде ориентированного графа, где вершины являются докуентами, а ребра — ссылками. Пользователь начинает свое движение из случайной вершины. Далее на каждом шаге он с вероятностью ε (обычно ε берется примерно 0.15) переходит в случайную вершину и с вероятностью $1 - \varepsilon$ по одному из ребер ведущих из данной вершины.



Введем понятие предельной вероятности: $PR_k(i)$ — это вероятность оказаться в вершине i через k шагов.

Утверждение: $\lim_{k\to\infty} PR_k(i) = PR(i)$, то есть для каждой вершины существует предельная вероятность находиться в ней.

2.3 Основное уравнение PageRank

Введем некоторые обозначения: $T_1, \ldots, T_n$ — вершины, из которых идут ребра в i, $C(X)$ — количество ребер исходящих из вершины X. Тогда имеет место следующее утверждение.

Утверждение: $PR(i) = \varepsilon/N + (1-\varepsilon)\sum_{i=1}^{n} \frac{PR(T_i)}{C(T_i)}$

Доказательство.

По определению $PR_k(i)$ верно следующее:

$PR_0(i) = 1/N$

$PR_k(i) = \varepsilon/N + (1-\varepsilon)\sum_{i=1}^{n} \frac{PR_{k-1}(T_i)}{C(T_i)}$

Теперь переходим к пределу в обеих частях последнего равенства и получаем доказываемое утверждение.

Доказательство завершено.

Разберемся откуда берутся оба слагаемых в правой части равенства: ε/N — вероятность того, что мы начнем с этой вершины, а сумма это сумма вероятностей прихода с других страниц по ссылкам.

Практически предельный PageRank не используют, а вместо $PR(i)$ берут $PR_{50}(i)$. После расчета PageRank нормализуется и округляется так, чтобы это было целое число от 1 до 10.

2.4 Матричная интерпретация

Рассмотрим матрицу L для построенного в предыдущем пункте графа. Заполнять матрицу будем по следующему принципу: $l_{ij} = \varepsilon/N$, если из вершины i в вершину j нет ребра, и $l_{ij} = \varepsilon/N + (1-\varepsilon)1/C(i)$, если ребро есть. Введем следующие обозначения:

$\overline{PR_k} = (PR_k(1), \ldots, PR_k(N))$

$\overline{PR} = (PR(1), \ldots, PR(N))$

Тогда будут верны следующие соотношения:
$PR_k = L^k\, PR_0$
$PR = L\, PR$

Последнее равенство является свойством PR, но если при умножении матрицы на вектор получается исходный вектор, то он является собственным для данной матрицы. Таким образом, PR можно также считать как собственный вектор для матрицы L.

2.5 Векторная модель

Рассотрим коллекцию документов, каждый из которых является списком слов. Пусть TF_{ij} — частота терма, то есть относительная доля слова i в тексте j; IDF_i — обратная встречаемость в документах, то есть величина, обратная числу документов содержащих слово i. Составим матрицу M с элементами $m_{ij} = TF_{ij} \cdot IDF_i$. Элемент матрицы определяет на сколько слово подходит документу, а благодаря множителю IDF_i будут цениться редкие слова.

2.6 Сингулярное разложение матриц

Введем определение сингулярного разложения. Пусть A — матрица $m \times n$. Разложение $A = USV$ называется сингулярным, если U — ортогональная матрица $m \times m$, V — ортогональная матрица $n \times n$, а S — диагональная матрица $m \times n$. Отметим несколько фактов касательно сингулярного разложения.

1. Для каждой матрица A можно вычислить сингулярное разложение, причем числа, стоящие на диагонали S — корни из собственных чисел AA^T. Если же матрица A квадратная, то диагональ S — собственные числа A.

2. Если S' — матрица S, в которой оставили только k наибольших чисел, то $US'V$ — самое близкое приближение матрицы A имеющее ранг k.

2.7 Применение сингулярного разложения

Размер матрицы документы-слова имеет порядок $10^{10} \times 10^6$, однако, эта матрица очень разряжена, поэтому для нее можно посчитать приближение ранга 100, используя сингулярное разложение. В получившейся матрице выделим 100 линейно-независимых строк, назовем их обобщенными словами; и 100 линейно-независимых столбцов, назовем их обобщенными документами. Теперь любой документ можно представить как линейную комбинацию выбранных стобцов, тоже верно и для слов со строками. Таким образом теперь мы перешли от базица из 10^6 слов к базису. Обобщенные слова и документы также называюь “собственными смыслами”.

С помощью собственных смыслов можно решить следующие задачи:

- Поиск “похожих” слов и “похожих” документов, так как у них будет похожее разложение по базису.
- Учитывая предыдущий пункт, можно тематически классифицировать документы исходя из разложения. Данную классификацию можно использовать при подсчете PageRank.
- Также можно организовать фильтрацию при выводе результатов поиска, например введя поле “spam”.

3 Поисковая оптимизация

Поисковая оптимизация — это необходимый комплект действий над сайтом, направленный на улучшение нахождения сайта поисковиками. Цель поисковой оптимизации — вывод сайта в первую десяку ответов поисковых запросов по ключевым запросам. Отметим, что это не обязательно произойдет, а иногда оптимизационны действия могут быть воспирняты поисковиками как “накрутка” и приведут к обратному эффекту. Для крупных сайтов удаление из каталога (одно из возможных действий поисковика) может првести к огромным убыткам: например, для сайта по торговле недвижимостью невывод в перво тройке запросов может првести к убыткам измеряемым десятками тысяч долларов.

3.1 Этапы поисковой оптимизации

- Предвартельный этап. На этом этапе идет анализ тематического сегмента, сайта (обдумываем будующее содержание), поиск ниши позиционирования (как подать сайт), составление семантического ядра запросов (определение основных запросов, при которых хотим попадать в первую десятку; стоит внимательно отнестись к синонимам — это позволит улучшить результаты поиска).
- Оптимизация под аудиторию. Основную часть этого этапа составляет работа с содержанием. Во-первых, необходимо обртить внимание на тексты, так как индексация идет именно по словам. Во-вторых, наличие на сайте дополнительных сервисов привлечет дополнтельную аудиторию. В-третьих, можно привести к специальному виду — текст хорошо читается, но при этом содержит много слов из ключевых запросов (SEO копирайтинг).
- Оптимизация под поисковые запросы. На данном этапе сайт подготовка сайта к индексации. Перечислим некоторые пункты.
  - Запрет индексации некоторых страниц. Это делается для того, чтобы сайт не находился по ненужным словам. В качестве примера можно приведем большой сайт-каталог, в котором в том числе было описание телефона Nokia, этот сайт находился именно

по запросу про этот телефон. Что явно не предусматривалось создателями сайта.

- Корректировка архитектуры сайта с учетом юзабилити. Например, добиться того, чтобы с главной страницы до любой можно было дойти за три–четыре клика, причем было бы легко догадаться, куда надо кликать.
- Работа над внутренними факторами. Заполнить тэги “description”, “keywords”, “title”. В последнем тэге полезной может оказаться структура: “название сатйа” - “название раздела” - “название страницы”.
- Визуальное и архитектурное (то есть выделение “тэгами”) выделение ключевых слов. Напомним, что при поиске слова выделенные тэгами имеют более высокий приоритет по сравнению с обычными.
- Корректировка текстов с точки зрения наличия ключевых слов и читаемости. С одной стороны, нельзя допустить, чтобы тест из-за обилия ключевых слов стал нечитаемым. С другой, каким бы замечательным с точи зрения литературы не был текст, но если в нем отсутствуют ключевые слова, то проиндексируется он плохо.
- Работа над описанием сайта для каталогов, обмена ссылками и прочего.
- Работа над внешними факторами. Сюда входят: регистрация в посковиках, каталогах, повышение тематической авторитетности сайта (например ссылка из новостной ленты сайта lenta.ru будет высоко цениться поисковиками при подсчете PageRank).

- Поддержка сайта. На этом этапе идет анализ достигнутых результатов и дальнейшая корректировка сайта. Также имеет смысл разместить рекламу. При ее размещении стоит учесть аудиторию сайтов, где реклама будет размещена. Ведь если ваш сайт посвящен “искусству бонсай”, то вряд ли имеет смысл размещать рекламу на сайтах посвященных “спорту”, то есть тематика сайтов должна быть схожа. Интересным примером продвижения сайта является премия учрежденная А. Лебедевым, когда он еще был никому не известен. Он организовал премию и раздал ее всем крупнейшим сайтам вместе с баннером, на котором была надпись “Лауреат премии лучший сайт”, и ссылочкой на сайт Лебедева. Как результат огромное количество ссылок на сайт со страниц с высоким PageRank.

4 Ссылки по SEO

Где учиться:

- SearchEngines.Ru

- SearchEngineWatch.Com
- http://seotext.ru
- SEO in Wikipedia.Org
- Ralph Wilson Checklist
- Энциклопедия Интернет Рекламы

Эффективный поиск в интернете:

- http://logic.pdmi.ras.ru/~yura/search.html