Принцип ранжирования интернет-страниц поисковыми системами

Оглавление

Теоретическая модель

Модель интернета

Перед тем, как приступить к обсуждению принципов ранжирования интернет-страниц, формализуем и конкретизируем некоторые понятия.

Пусть *интернет* – система из N пронумерованных объектов – *страниц*. Каждая i-ая страница ($i \in \{1,2,\dots,N\}$) представляет собой множество W_i, состоящее из набора номеров страниц, на которые она ссылается. Всего на i-ой странице $n_i = |W_i|$ ссылок. Через L_i будем обозначать множество страниц, ссылающихся на i-ую страницу.

Для упрощения рассуждений будем считать, что с каждой страницы на другую может быть не более одной ссылки[1]. Таким образом, вероятность перехода с j-ой страницы на i-ую будет равна:

$$p_{ij} = \begin{cases} 0, j \notin L_i \\ \dfrac{1}{n_j}, j \in L_i \end{cases}$$

Ранг страницы

Задача поисковой машины – находить по запросу пользователя наиболее релевантные[2] страницы. Для этого ей необходимо некоторым образом ранжировать или, другими словами, отбирать, сортировать страницы. Введём понятие значимости или ***ранга*** страницы. Пусть x_i – ранг i-ой страницы (Langville, “PageRank”, 84-88):

$$x_i = \sum_{k \in L_i} (x_k * p_{ik}) = \sum_{k \in L_i} \frac{x_k}{n_k} \quad (1)$$

Пояснение: мы рассматриваем ранг страницы с номером i. Возьмём все страницы, которые на неё ссылаются – это и есть множество L_i. Для каждой $k \in L_i$ найдём вероятность перехода с этой страницы на интересующую нас i-ую и умножим её на ранг k-ой страницы. Затем просуммируем полученные результаты. Логика проста: чем выше ранг страницы, которая ссылается на интересующую нас страницу, тем «ценнее»

---

[1] Иными словами, мы строим **простой** ориентированный граф на N вершинах. По определению, кратные рёбра и петли запрещены, но две вершины могут соединяться двумя разнонаправленными дугами.

[2] Релева́нтность (лат. relevo — поднимать, облегчать) в информационном поиске — семантическое соответствие поискового запроса и поискового образа документа. В более общем смысле, одно из наиболее близких понятию качества «релевантности» — «адекватность», то есть не только оценка степени соответствия, но и степени практической применимости результата, а также степени социальной применимости варианта решения задачи. (Википедия)

i-ая страница. Чем больше других ссылок на *k*-ой странице, тем ниже вероятность перехода на *i*-ую. (Иванов, 72) В итоге, даже одна ссылка со страницы высокого ранга сильно увеличивает ранг нашей страницы. **Заметим, что ранг некоторой страницы невозможно найти, не зная рангов других страниц.**

Пример 1.

*П*редположим, что мы дилеры компьютерной техники. Что ценнее для нашего ресурса с точки зрения поисковика и клиентов – наличие ссылки на наш сайт на главной странице компании Apple (которая, безусловно, имеет высокий ранг) или ссылка с сайта никому неизвестного дяди Васи, продающего самодельные процессоры? А теперь представьте, что на главной странице Apple всего одна ссылка и она ведёт на сайт нашей компании – вероятность перехода = 100%!

Задача ранжирования

Матрица Маркова

Запишем теперь некоторый вектор $\bar{x}$, где x_i в столбце – ранг i-ой страницы[3]:

$$\bar{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix}.$$

Составим из элементов p_{ik} (вероятность перехода с k-ой страницы на i-ую) матрицу P и назовем её матрицей вероятностей. Как обычно, первая цифра индекса обозначает номер строки, а вторая – номер столбца. Таким образом, в i-ом столбце матрицы P представлены все вероятности перехода с i-ой страницы на каждую страницу множества W_i, а в k-ых строках – вероятности перехода на i-ую страницу со страниц из множества L_i. Весьма очевидно, что все элементы такой матрицы неотрицательны, а сумма всех элементов по столбцам равна 1 (по определению понятия вероятность). Такая матрица называется *матрицей Маркова* (Гантмахер, 381). Теперь формулу (1) можно записать следующим образом:

$$P * \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_N \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_N \end{pmatrix}$$

По сути, выполнив умножение, мы увидим, что ранг страницы равен сумме произведений ранга ссылающихся на неё страниц и вероятностей перехода с каждой из них на данную страницу. Таким образом, задача поиска ранга страницы сводится к нахождению собственного вектора со значением $\alpha = 1$.

Пример 2.

Пусть у нас есть две страницы в интернете. Обозначим их соответственно номерами 1 и 2. Пусть на 1-ой странице содержится 2 ссылки: на 1-ую и на 2-ую. А на 2-ой странице только одна ссылка – на 1-ую. Множества, соответственно равны: $W_1 = \{1,2\}$, $W_2 = \{1\}$, $L_1 = \{1,2\}$, $L_2 = \{1\}$,Найти ранги страниц.

1) Распишем вероятности и составим матрицу P:

[3] Интересное дополнение: если поделить все координаты данного вектора на их сумму, то мы получим вероятность нахождения на каждой из страниц.

p_{ik} – вероятность перехода с k на i

$$p_{11} = \frac{1}{2}, \qquad p_{12} = \frac{1}{1} = 1, \qquad p_{21} = \frac{1}{2}, \qquad p_{22} = 0$$

$$P = \begin{pmatrix} 0,5 & 1 \\ 0,5 & 0 \end{pmatrix}$$

2) Решим уравнение:

$$P\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

$$\begin{pmatrix} 0,5 & 1 \\ 0,5 & 0 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

$$\begin{cases} 0,5x_1 + x_2 = x_1 \\ 0,5x_1 = x_2 \end{cases}$$

$$x_2 = 0,5x_1$$

$$x_1 = 1, \qquad x_2 = 0,5$$

3) А теперь давайте рассудим логически: как же так, с каждой страницы на другую есть только по одной ссылке, но при этом ранг у них разный? В условии было сказано, что с первой страницы есть ссылка на неё саму. Разве это должно повышать ранг и учитываться при индексации? Конечно нет[4]. Поисковый робот при индексации страниц игнорирует элементы, удовлетворяющие условию: $w_i \in W_i = i$.

С учётом этого замечания получаем:

$$P = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

$$x_1 = x_2 = 1$$

Решение задачи ранжирования

Закономерным является вопрос существования собственного вектора со значением один, ведь по сути именно этот вектор и позволяет решить задачу ранжирования (Назин и др., 1).

Теорема Перрона-Фробениуса (Ланкастер, 260):

Для любой квадратной матрицы со строго положительными

[4] Кто прочитал сноску 1 мог увидеть, что из-за того, что мы строим простой ориентированный граф на N вершинах, петли запрещены

элементами (т.е. мы исключаем из рассмотрения матрицы на подобие P из примера 2) найдется вектор

$$x{:}\, Px = \alpha x, \alpha > 0$$

Определение (Гантмахер, 352):

Матрица называется *неразложимой*, если перестановками рядов ее нельзя привести к виду

$$A = \begin{pmatrix} B & 0 \\ C & D \end{pmatrix}, \text{где } B \text{ и } D - \text{квадратные матрицы}$$

Следствие из теоремы Перрона-Фробениуса (Ланкастер, 261):

Для положительной марковской матрицы $\alpha = 1$ – наибольшее характеристическое число, являющееся корнем характеристического уравнения, причём для неразложимой матрицы данное наибольшее значение будет единственным.

Мы определили класс матриц, для которых всегда будет существовать собственный вектор с собственным значением 1. С помощью теоремы Фробениуса-Перрона и следствия из нее мы получили, что все неразложимые марковские матрицы будут иметь искомый собственный вектор, а значит, задача ранжирования будет разрешима.

Для поиска собственного вектора компьютер использует степенной метод (Стренг, 332-335):

$$P * x^{k-1} = \alpha x^k, \qquad \alpha = 1,$$

k-ое приближение к искомому наибольшему собственному значению вычисляется по формуле отношения Релея (Стренг, 307):

$$\alpha_k = \frac{(x^{k+1}, x^k)}{(x_k, x_k)}$$

Таким образом, компьютер многократно выполняет операцию приближения к искомому вектору, начиная с некоторого нетривиального вектора и с каждым шагом умножения все более приближаясь к искомому вектору.

Вспомогательная теорема:

Если начальный вектор x^0 нетривиален, а α – единственное наибольшее собственное значение матрицы, то степенной метод сходится.

Важность этой теоремы заключается в том, что для неразложимой матрицы P при применении степенного метода, используемого компьютером, всегда найдется требуемый вектор, и задача ранжирования будет разрешима.

Проблемы и методы их решения

Проблема «висячих вершин»

Матрица, рассмотренная в примере 2, была марковской, но разложимой, т.к. содержала нулевой элемент. Заметим, что любая страница, не имеющая исходящих ссылок ($|W_i| = 0$), генерирует в матрице вероятностей нулевой столбец ($p_{ji} = 0 \; \forall \; j$). Значит, степенной метод может не сходиться, т.к. матрица будет разложимой (Langville, “Tiny Web”, 92-113).

Предпосылка для решения

Логично будет предположить, что, попадая на подобную страницу-ловушку, человек закрывает ее и равновероятно попадает на любую другую страницу. То есть для данной страницы:

$$n_i = N, p_{ji} = \frac{1}{N} \forall \; j$$

Введение данной предпосылки позволяет в нулевом столбце сделать замену: $0 \longrightarrow \frac{1}{N}$ и получить тем самым матрицу со строго положительными элементами (Langville, “The Fix”, 116-118).

Проблема разреженности матрицы

Отметим, что отсутствие нулевых столбцов не исключает существования нулевых элементов матрицы. И вправду, в интернете существуют миллиарды страниц, а значит, большое число элементов будут равными нулю (см. пример), и степенной метод не обязательно будет сходиться (Langville, “Nasty Problem”, 119-127).

Решение проблемы. Рецепт Google – «Page Rank» (Langville, “The Google Fix”, 128-130)

Очевидно, что необходимо сделать так, чтобы из матрицы пропали нулевые элементы. Рассмотрим некоторую матрицу:

$$M = \beta P + (1 - \beta) S$$

$$0 < \beta < 1, \qquad S - \text{матрица } N \times N, \qquad s_{ij} \equiv const$$

Без сомнения, S – неразложимая матрица Маркова, имеющая собственное значение $\alpha = 1$. Значит, степенной метод, основанный на этой матрице, будет сходиться.

Идея PageRank заключается в следующем: с определенными коэффициентами берутся две матрицы, первая – исходная матрица вероятностей, в которой некоторые элементы нулевые, вторая – матрица $N \times N$, которая по сути является матрицей вероятностей для сети, в которой каждая страница содержит ссылки на все остальные и на саму себя (вероятность любого перехода составляет $p_{ij} = \frac{1}{N}$).

В реальности используется $\beta = 0{,}85$, позволяющее получить достаточно близкий к x собственный вектор y[5]:

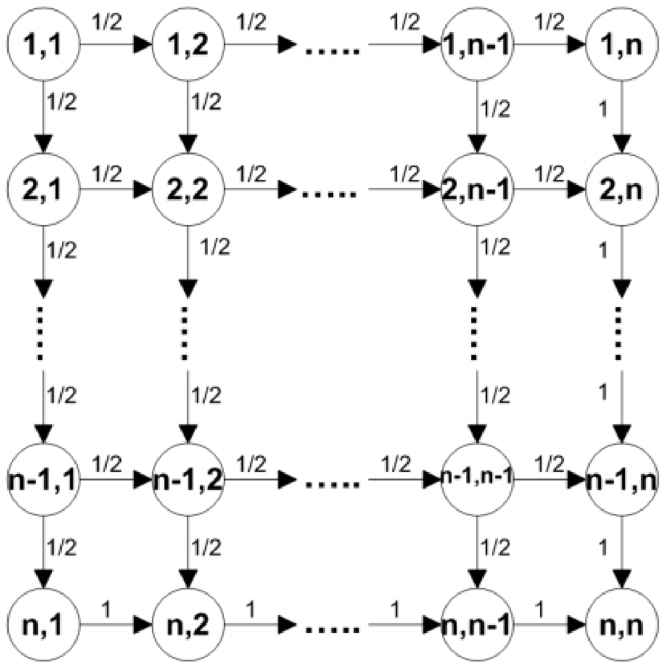
$$M * y^{k-1} = y^k$$

Вероятности перехода

Пример 3.

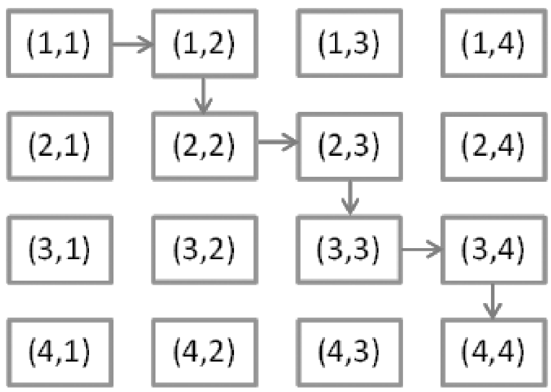
Рассмотрим некоторую сеть, в которой задействовано n^2 страниц. Пусть некоторая страница (n, n) ссылается на все страницы сети (все страницы обозначаются двумя координатами для удобства). Тогда вероятность перейти на любую из страниц или остаться составляет $\frac{1}{n^2}$. То есть, оказавшись на данной странице, в 1 из n^2 случаев пользователь перейдет на какую-то конкретную страницу. Пусть модель сети имеет вид:

[5] Почему используется именно такой показатель – отдельный вопрос, не рассматриваемый в данной работе, однако представляющий определённый интерес. Общий принцип основан на стремлении снизить потери точности при выборе параметра.



Если мы находимся на странице с номером $(1,1)$, то сделав $2(n-1)$ шагов, мы можем оказаться на странице с номером (n,n).

Например, для перехода из (1,1) в (4,4) необходимо $2(4-1)=6$ переходов:



Тогда с любой страницы (i,j) через $2n-i-j$ шагов можно попасть на страницу (n,n). Так, со страницы $(2,3)$ можно попасть на $(4{,}4)$ через

$8-2-3=3$ шага.

Для рассматриваемой нами модели сети марковская матрица будет иметь вид (рассмотрим теперь размерность 9×9):

$$P_{9\times 9} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{n^2} \\ \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{n^2} \\ 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{n^2} \\ \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{n^2} \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & \frac{1}{n^2} \\ 0 & 0 & 1 & 0 & \frac{1}{2} & 0 & 0 & 0 & \frac{1}{n^2} \\ 0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & \frac{1}{n^2} \\ 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & 1 & 0 & \frac{1}{n^2} \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & \frac{1}{n^2} \end{pmatrix}$$

Элементы данной матрицы как раз и есть вероятности перехода с некоторой конкретной страницы на другую страницу.

Практический аспект с точки зрения теории

Обман поисковых систем, накрутка ранга страниц

Существует множество способов улучшения индексируемости сайта. Большинство из них абсолютно естественны. Например, размещение уникального контента, оптимизация кода, повышение репрезентативности страниц и т.д. и т.п. Однако некоторые ресурсы стремятся обойти правила честной конкуренции и пробиться на верхушку поисковой выдачи путём обмана поисковых систем.

Одно из самых эффективных с точки зрения «начисления поисковой машиной баллов» способов – **искусственное** увеличение числа внешних ссылок на ресурс.

$$x_i = \sum_{k \in L_i} \frac{x_k}{n_k} \quad (1)$$

Безусловно, хотелось бы, чтобы ссылка на сайт была расположена на странице с высоким рангом какой-нибудь известной компании, однако, «ничего не делая», этого можно добиться лишь путём прямого вторжения или взлома, что в конечном счёте ни к чему не приведёт. Поэтому, приходится довольствоваться не высоким x_k, а большим $|L_i|$, т.е. увеличивать общее число ссылок на ресурс.

В связи с этим в интернете давно существуют сервисы, предоставляющие возможность вручную или автоматически нарастить «ссылочную массу» ($|L_i|$), – *каталоги* и *биржи ссылок*. Биржа ссылок – это рынок, где можно купить ссылку на ваш сайт или продать место под баннер на своей странице. Сама биржа берёт за это небольшой процент. Покупатели здесь, так называемые *оптимизаторы,* – владельцы сайтов, которые хотят повысить ранги своих страниц путём покупки ссылок. Продавцы, *веб-мастеры,* – люди, готовые разместить на своих ресурсах ссылки на сайты оптимизаторов с целью заработка. Тем не менее, этот способ перестал быть эффективным с тех пор как поисковики научились с высокой вероятностью отслеживать вебмастеров и операции на биржах и начали беспощадно санкционировать такого рода попытки обмана, порой навсегда вычёркивая домены оптимизаторов из базы индексации.

Каталог – большой структурированный сборник ссылок на разные сайты интернета. В отличие от биржи каталоги создаются с благородной целью сортировки информации в интернете и помощи пользователям в поиске сайтов, занимающихся продажей тех или иных продуктов или услуг. Существуют легальные (например, Яндекс.Каталог) ресурсы, которые при выполнении определённых требований размещают ссылку безвозмездно. Но существуют и «чёрные» каталоги, которые, во-первых, требуют от владельца сайта размещения ответной ссылки (таким образом, оптимизатор становится одновременно веб-мастером), во-вторых, не подбирают правильный раздел для вашего сайта, тем самым нарушая процесс структуризации. Постепенно поисковая машина «забанивает» чёрные каталоги. Подумайте сами, разве Яндексу понравится, что два сайта размещают друг на друга ссылки: каталог на сайт и сайт обратно на каталог? Риски, что Яндекс посчитает ресурс поисковым «спамером», сильно возрастают, а с ними возрастает и риск санкций со стороны поисковых систем.

Теперь хотелось бы остановиться на том, как одновременно разместить ссылку в чёрном каталоге и избежать наказания со стороны поисковой системы (“Как обмануть обманщиков поисковых систем”).

1) Оптимизатор может дать каталогу ссылку на страницу, где он разместил «плату» за услугу – обратную ссылку на каталог, а затем просто снять обратную ссылку на черный каталог через некоторое время. Но вопрос в том, как узнать, когда можно снять обратную ссылку? Возможно и вполне вероятно, что робот каталога или модератор проверяют наличие обратной ссылки с некоторой периодичностью. Поэтому таким образом обманывать чёрные каталоги бессмысленно
2) Как того и требует каталог, оптимизатор размещает ссылку на него на своём сайте, на специальной странице, где он собирает и все другие аналогичные ссылки. Теперь у него в некотором смысле появился свой собственный каталог каталогов. Специфика этой страницы заключается в том, что на неё с самого сайта оптимизатора не ведёт ни одна ссылка! Для того, чтобы поисковый робот попал на эту страницу, он должен перейти на неё по ссылке, но ссылки нигде нет и страница не может быть проиндексирована ($|L_i| = 0 \Rightarrow p_{ij} = 0\ \forall j$). С другой стороны, она существует и при регистрации сайта в каталоге оптимизатор указывает её как подтверждение того, что он разместил обратную ссылку на каталог. Недостаток этого способа в том, что определённый риск всё же существует. Дело в том, что программа проверки каталогом страницы с обратной ссылкой может учитывать ранг этой страницы, а так как страница не была проиндексирована[6], её ранг равен нулю.

[6] С точки зрения графа, построенного на вершинах страниц, она представляет собой изолированную вершину

3) Третий способ – не размещать ссылку на каталог на своем сайте. Например, в каталоге оптимизатор регистрирует сайт с адресом www.site.ru, а обратную ссылку на каталог размещает на странице, принадлежащей другому домену www.site2.ru/catalog. Последняя страница может быть при этом индексируемой и иметь $|L_i| > 0$ и $n_i > 0$. Перед поисковой системой оптимизатор чист, т.к. не обменивается ссылками, и перед каталогом он тоже чист, т.к. обратная ссылка размещена на странице с положительным рангом (пусть и небольшим). Главное – чтобы модератор каталога это принял. Такие методы со временем тоже вычисляются поисковыми машинами, которые постепенно «умнеют». Например, можно сверить данные о владельцах доменов site.ru и site2.ru.

Список литературы

Google's PageRank and Beyond: The Science of Search Engine Rankings [В Интернете] / авт. Langville Amy и Meyer Carl // Amy N. Langville's homepage. - http://langvillea.people.cofc.edu/Citadel.pdf.

The PageRank Citation Ranking: Bringing Order to the Web [Статья] / авт. Brin Sergey [и др.]. - 1998 г..

Как обмануть обманщиков поисковых систем [В Интернете] // Блог iforl.ru. - 16 мая 2012 г.. - http://www.iforl.ru/как-обмануть-обманщиков-поисковых-си/.

Линейная алгебра и ее применения [Книга] / авт. Стренг Гилберт. - Москва : "Мир", 1980.

Модель задачи ранжирования и её исследование [Статья] / авт. Тимонина Анна Валерьевна. - 2009 г..

Продвижение сайта в поисковых системах [В Интернете] / авт. Иванов Андрей и Ашманов Игорь // WEB-студия SEOmake.ru. - http://seomake.ru/page005.php.

Рандомизированный алгоритм нахождения собственного вектора стохастической матрицы с применением к задаче PageRank [Статья] / авт. Назин Александр Валерьевич и Поляк Борис Теодорович // Автоматика и телемеханика. - Москва : [б.н.], 2011 г..

Теория матриц [Книга] / авт. Гантмахер Феликс Рувимович. - Москва : "Наука", 1967.

Теория матриц [Книга] / авт. Ланкастер Питер. - Москва : "Наука", 1973.