Статистический подход к решению проблемы определения страниц soft 404

© С.С. Чирков

Санкт-Петербургский государственный университет

sergechircov@yandex.ru

Аннотация

Проблема распознавания страниц soft 404 является актуальной проблемой современных поисковых машин. Прежде всего, это связано с тем, что страницы soft 404 нельзя определить по коду протокола HTTP, как в случае с обычными страницами 404. Ранее предложенные решения [1, 2] поиска страниц soft 404 не смогли в полной мере решить данную проблему. В данной статье представлен новый подход к распознаванию страниц soft 404, основанный на представлении страниц в виде наборов слов (выражений) и использовании алгоритмов машинного обучения для оценки близости этих наборов.

1 Введение

Ранее было произведено много исследований, связанных со сломанными страницами. Сломанная страница – это страница, ссылка на которую сломана, или страница являющаяся soft 404. Страница soft 404, если переходить к понятиям кодов протокола HTTP, – это страница, которую нужно было возвратить с HTTP кодом 404 (т. е. страница не существует), но разработчики сайта решили возвратить её с HTTP кодом 200 (т. е. запрос клиента обработан успешно, и сервер возвращает хорошую страницу).

В ходе исследования [3], проведенного среди 150 популярных сайтов из доменов .com, .gov, .edu, .org, .net и .mil, было установлено, что через полгода 50% ссылок на сайтах было сломано.

В ходе исследования [4] было установлено, что процент сломанных страниц внутри одного домена верхнего уровня приблизительно равен проценту сломанных страниц во всем интернете.

В работе [5] было установлено, что количество сломанных страниц во всем интернете в 1997 году составляло 5 – 8%. Сейчас эта цифра выше из-за появления страниц soft 404.

В работе [6] было установлено, что страницы soft 404 составляют 25% от общего числа сломанных страниц, т. е. если общее число сломанных страниц равно 5 – 8%, то число страниц soft 404 составляет 1,5 – 2%.

Разработчики сайтов могут использовать страницы soft 404:

- в качестве страниц, на которых можно поместить информацию о том, почему страница не доступна в данный момент, и предложить способы решения проблемы;
- в качестве «припаркованных сайтов», т. е. сайтов, которые исчезли и были перерегистрированы для раскрутки другого сайта, например, порно-сайтов; в работе [7] рассматривается сайт, для раскрутки которого использовалось более 4000 «припаркованных сайтов».

Чем плохи страницы soft 404? Во-первых, поисковой системе приходится индексировать такие страницы, что требует определенного места для хранения информации. Во-вторых, страницы soft 404 могут привести к некорректной работе алгоритмов поисковой системы. Одним из таких алгоритмов является алгоритм ранжирования, самый важный алгоритмом поисковой системы. Для примера рассмотрим алгоритм ранжирования PageRank [8]. Очевидно, что «припаркованные сайты» будут оказывать влияние на работу этого алгоритма ранжирования, завышая реальные позиции сайтов, на которые ссылаются «припаркованные сайты», среди результатов, найденных по запросу пользователя.

Рассмотрим предложенные решения поиска страниц soft 404 [1, 2]. Первое решение рассмотрено в работе [1]. В ней авторы представили алгоритм определения сломанных страниц, также представили некоторую меру, при помощи которой можно подсчитать степень разложения интернета. Коротко алгоритм определения сломанных страниц можно записать так:

1. Посылаем два запроса на сервер. Первый запрос – на интересующую нас страницу. Второй запрос – на страницу, которой, вероятно, не существует на сервере.
2. Сравниваем количество перенаправлений сервера для обоих запросов.
3. Затем сравниваем содержание страниц, используя метод шинглов [9].

Труды 12[й] Всероссийской научной конференции «Электронные библиотеки: перспективные методы и технологии, электронные коллекции» – RCDL'2010, Казань, Россия, 2010

ROMIP BY.Web 2007		
Состав	Размер	Предоставлена
страницы домена .by из индекса Яндекс (май 2007) (на глубину 3 ссылки от стартовой,) процент ссылок внутрь коллекции ~25%	1 524 676 док 8 Гб	Яндекс

На основании данных о количестве перенаправлений сервера и подсчитанных шинглах, делаем вывод о том, является ли рассматриваемая страница страницей soft 404 или нет. Недостатком алгоритма является то, что автор считал, что главная страница web-сайта не является soft 404. Для его исследования этого было достаточно, но, как видно на практике, дело обстоит иначе (достаточно вспомнить «припаркованные сайты», которые состоят всего из одной страницы).

Авторы работы [2] разработали инструмент Walden's Paths Path Manager, который помогает пользователям в исследовании изменений, связанных с ресурсами, ссылки на которые присутствуют на странице. В работе отмечено, что незначительные изменения в содержании страницы оставляют страницу релевантной, а глобальные изменения приводят к нерелевантности. При работе программа производит сравнение данной страницы со страницей, ранее сохраненной в кэше. При сравнении акцент делается на структуру документа, заголовки, ссылки и ключевые слова. Полученный инструмент вполне можно применять для создания алгоритма распознавания страниц soft 404. Недостатком этого алгоритма является то, что первый раз все страницы необходимо просмотреть вручную, чтобы исключить уже существующие сломанные страницы.

Недостатком алгоритма, описанного в [1], является его направленность на статические сайты, т. к. принятие решения о том, является ли страница soft 404 или нет, делается на основании равенства шинглов и равенства количества перенаправлений сервера. А для динамических сайтов значения шинглов для одной и той же страницы могут различаться, т. к. при каждом новом посещении страницы содержание будет меняться, хотя ее структура может оставаться неизменной, например, http://www.vse-putem.com/djkdjk.html.

В связи с этим актуальна задача создания алгоритма, не зависящего от окружения, т. е. ссылок, присутствующих на рассматриваемой странице или других страниц сайта.

2 Описание подхода к решению

Решение включало в себя следующие этапы:

1) Составление выборки документов, состоящей из страниц soft 404 и хороших страниц.

2) Составление словаря, состоящего из слов, часто встречающихся на страницах soft 404 или относящихся к теме soft 404.

3) Представление страниц коллекции в виде признаков. Признаку соответствует слово из словаря.

4) Реализация алгоритма классификации по построенным векторам.

Для составления выборки документов использовалась коллекция документов ROMIP BY.Web 2007 [10] (см. таблицу выше). Коллекция состоит из большого количества документов, и искать страницы вручную во всей коллекции не представлялось возможным.

Следующий итеративный алгоритм (1) позволил сократить количество страниц, среди которых нужно было искать страницы soft 404:

1) Вручную составить словарь слов, чаще всего встречающихся на страницах soft 404.

2) Найти страницы, на которых встречаются слова из словаря.

3) Разметить вручную найденные страницы. Если количество найденных страниц достаточно для дальнейшей работы, то перейти к пункту 5, иначе – к пункту 4.

4) Дополнить словарь новыми словами, встречающимися на найденных страницах. Далее перейти к пункту 2.

5) Получена выборка размеченных страниц.

Основные сложности, которые возникают при работе по этому алгоритму, – это разметка страниц, которую приходится выполнять асессору. Для того чтобы облегчить работу асессора, будем полагаться на следующие наблюдения о страницах soft 404:

1) Имеет значение, в каком HTML-тэге встретилось слово. Например, если фраза «error 404» встретилась в тэге <h1>, то эта страница с большей вероятностью является soft 404, чем страница, в которой эта же фраза встретилась в тэге <p>.

2) Некоторые HTML-тэги можно объединить в группы, тогда 1 верно и для групп. Примеры: {<h1>, <h2>, <h3>} – группа тэгов или {<p>, , <font>, } – группа тэгов. Такое объединение позволит сократить количество признаков, описывающих страницу.

3) Существуют тэг T и слово W (фраза F) такие, что если W(F) встречается в предложении, находящемся в тэге T, то с некоторой вероятностью страницу можно отнести к одному из классов: soft 404 или хорошая страница. Например, если в тэге <title> встречается «about error», то рассматриваемая страница с 90%-ой вероятностью будет хорошей.

В пределах одного предложения фраза, состоящая из нескольких слов, имеет большую значимость, чем слова фразы по отдельности и в разных предложениях, принадлежащих одному тэгу. Например, если в тэге <title> встретились два слова «about» и «error», и они находятся в двух разных предложениях, то мы не можем утверждать, что

данная страница с вероятностью 90% является хорошей, а если в одном предложении встречается «about error» – то можем.

Учитывая наблюдения, перечисленные выше, приходим к следующему «слабому» (из-за ограниченного количества фраз в словаре, порядка 115 фраз) алгоритму определения страниц soft 404:

1) Составляем тренировочное множество, состоящее из soft 404 и хороших страниц.

2) Все HTML-тэги разбиваем на группы.

3) Для каждой группы тэгов составляем по 2 словаря фраз, используя страницы из тренировочного множества. Один из словарей, относящихся к одной группе тэгов, может оказаться пустым.

4) Считаем частоты для фраз по следующей формуле:

$$P(f_i) = \frac{\#(f_i)}{\sum\limits_{\forall j} \#(f_j)},$$

где $\#(f)$ – частота появления фразы на всех страницах тренировочного множества.

5) Для каждой страницы составляем вектор параметров $A = (a_1, a_2, \cdots, a_m)$, где

$a_i = -\sum\limits_{\forall j,k} \log(c_{jk})$, i=1..m,

$$c_{jk} = \begin{cases} P(f_k), & \text{если } f_k \in s_j, \\ 1 & \text{иначе}, \end{cases}$$

f_k – из i-го словаря, s_j – из множества предложений содержащихся в тэгах $[(i+1)/2]$-ой группы тэгов. Размерность вектора A определяется количеством групп тэгов, m = 2·(кол-во групп тэгов).

6) Для классификации по построенным векторам воспользуемся алгоритмом k-NN.

Для тренировки и тестирования представленного выше алгоритма использовались множества, состоящие из 250 страниц. Выборки осуществлялись из страниц, присутствующих в выдаче поисковой системы Google по запросам «error 404», «page not found», «ошибка 404» и т. д. Все тэги были разбиты на 3 группы:

1) <title>;

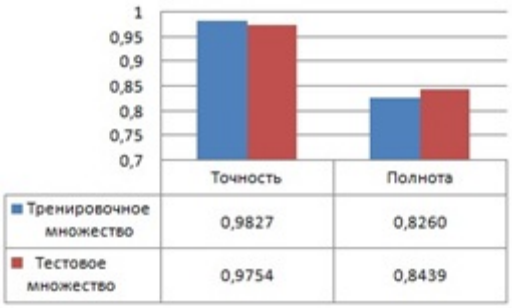
2) <p>, , , <font>, ,
, <b>, <pre> и <i>;

3) <h1>, <h2>, <h3>, <h4>, <h5>, <h6> и <body> (для случаев, когда фраза не находилась ни в одном из тэгов, а находилась в <body>).

Для классификации страниц использовали 10 ближайших соседей как число, доставляющее максимальную точность алгоритму. Точность и полнота алгоритма указаны на рис. 1.

Использование этого алгоритма позволило значительно уменьшить время на разметку страниц. Для поиска 2800 страниц soft 404 во всей коллекции потребовало бы 280 чел.·часов, если предположить, что:

- страницы soft 404 распределены равномерно,
- количество страниц soft 404 составляет ~1% от общего числа всех страниц,
- на разметку 1000 страниц требуется 1 чел.·час.



	Точность	Полнота
Тренировочное множество	0,9827	0,8260
Тестовое множество	0,9754	0,8439

Рис. 1

При использовании «слабого» алгоритма это время составило 5 чел.·часов (без учета времени написания и тренировки самого алгоритма). С использованием словаря из 145 слов в алгоритме (1) в исходной коллекции было найдено 95913 страниц. На первой итерации в алгоритме (1) использовался словарь из 71-го слова. Из выборки, полученной на втором шаге алгоритма (1), применяя, описанный выше алгоритм, удалось выделить 3031 страницу, потенциально являющуюся soft 404. После разметки получили результат, представленный на рис. 2. Разметка производилась одним асессором.

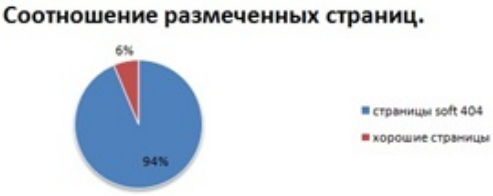


Рис. 2

После 7-ми итераций алгоритма (1) получили результат, представленный на рис. 3.

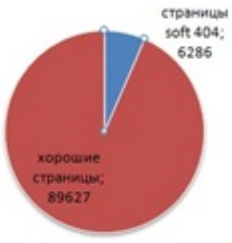


Рис. 3

Среди хороших страниц осталось приблизительно 2 – 3% страниц soft 404 (оценка получена методом бутстраппина [11]).

Словарь, полученный после работы по алгоритму (1), был расширен словами, встречающимися более чем на 1% страниц выборки. Страницы выборки были представлены в виде векторов TF слов, встречающихся в расширенном словаре. В качестве алгоритмов машинного обучения были выбраны алгоритмы решающих деревьев: Random Forest [12] и Id3 [13], а также алгоритм, основанный на

методе ближайших соседей. Для того чтобы использовать алгоритмы, основанные на решающих деревьях, необходимо было перевести непрерывные величины TF в величины, принимающие конечное количество значений. Перевод производится по следующему правилу:

$$TF = 0 \Rightarrow 0,$$

$$TF \in (0, t_1] \Rightarrow 1,$$

...

$$TF \in (t_n, 1] \Rightarrow n+1.$$

Неизвестными остались только точки $t_1, t_2, \ldots, t_n$ деления отрезка [0,1]. Для каждого документа выборки составим множество, состоящее из величин TF для каждого признака. Затем объединим все множества в одно. Точки полученного множества отсортируем по возрастанию значений TF и пронумеруем. На графике (рис. 4), приведенном ниже, видно, как расположены эти точки.

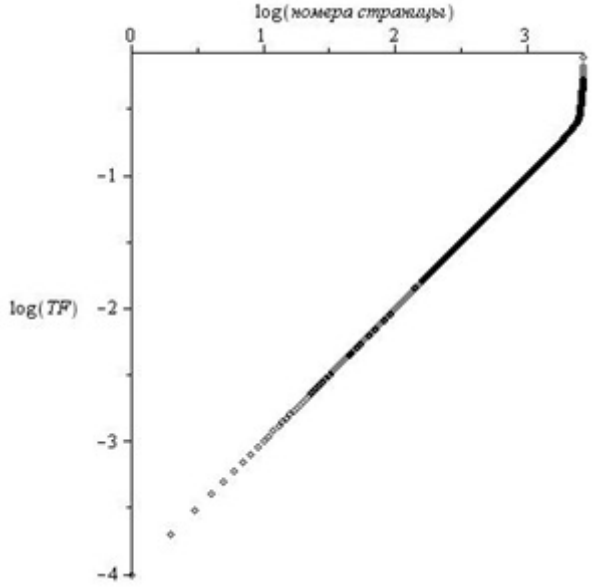


Рис. 4

На графике видно, что на отрезке [0, 0.2] точки расположены равномерно, а на отрезке [0.2, 1] плотность точек такая же, как плотность экспоненциального распределения. Этот факт можно использовать при выборе точек деления $t_1, t_2, \ldots, t_n$, например:

$$t_i = \begin{cases} [(\alpha^{-k} - 0)/T] \cdot i, & i = 1, \ldots, T-1, \\ \alpha^{i-T-k}, & i = T, \ldots, T+k-1, \end{cases},$$

где $\alpha^{-k} \approx 0.2$, $k \in \mathrm{N}$, $1 < \alpha \le 5$, $T \in \mathrm{N}$, $T \ge 2$.

Параметры α, k и T выбираются таким образом, чтобы обеспечить алгоритмам Random Forest и Id3 максимальную точность и минимальность получаемой модели. В данной работе при настройке алгоритмов было рассмотрены несколько наборов параметров, оптимальным оказался набор (α=1.5, k=4, T=36).

3 Результаты

Обучение алгоритмов производилось на выборке из 95913 страниц: 6286 страниц soft 404, 89627 хороших страниц. Начнем с рассмотрения алгоритма Random Forest. Количество деревьев Random Tree, используемых для создания Random Forest, было 5 (из-за ограничения на оперативную память 1,5 Гб). Тренировочное множество для каждого отдельного Random Tree строилось по методу «бэггинга» Брэймана [14]. Для тестирования Random Forest использовалась вся выборка. Ниже представлены результаты работы Random Forest.

Если 50% деревьев голосуют за soft 404:

порог[i]	точность	полнота
0.5	0.8957	0.9845
0.6	0.8976	0.9832
0.7	0.9007	0.9791
0.8	0.9614	0.7416
0.9	0.9608	0.6439

Если 80% деревьев голосуют за soft 404:

порог	точность	полнота
0.5	0.9702	0.9481
0.6	0.9712	0.9443
0.7	0.9741	0.9368
0.8	0.9962	0.5927
0.9	0.9967	0.5381

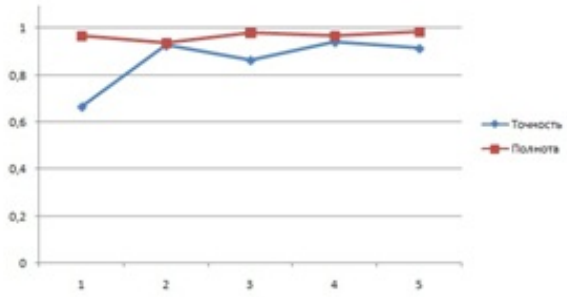


Рис. 5

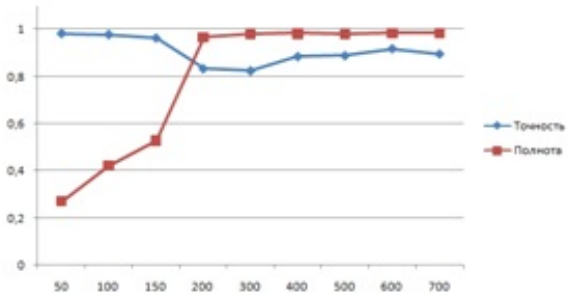


Рис. 6

Кривая обучения Random Forest в зависимости от количества деревьев (количество деревьев комитета) представлена на рис. 5. Кривая обучения Random Forest из 5-ти деревьев в зависимости от высоты деревьев представлена на рис. 6.

Рассмотрим алгоритм Id3. Выборку разделили на два множества: тренировочное – 100·(1-θ)% от выборки, тестовое – оставшиеся 100·θ% от выборки, где $\theta \in (0,1)$. Как показали результаты тестирования, алгоритм подвержен переобучению, связано это с высокой детализацией дерева. На рис. 7 изображена кривая обучения для тренировочного и тестового множеств. На рис. 8 представлена кривая обучения алгоритма k-NN в зависимости от k.

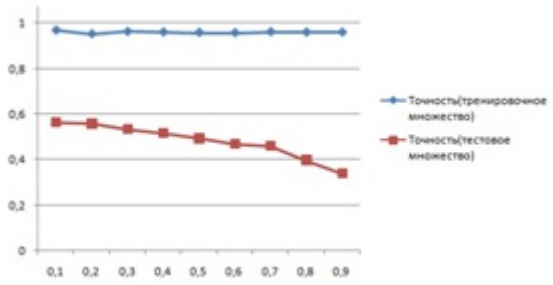


Рис. 7

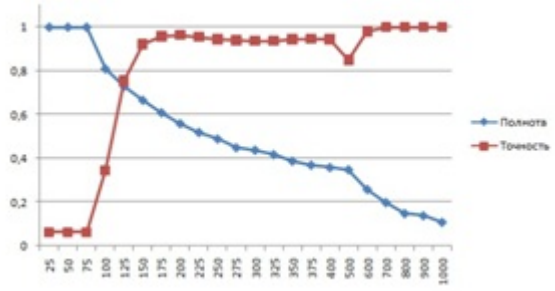


Рис. 8

Результаты сравнения различных алгоритмов машинного обучения представлены в таблице ниже. Как видно, алгоритм Random Forest имеет высокие показатели полноты и точности в отличие от других сравниваемых алгоритмов, у которых только один из показателей близок к показателям Random Forest. Следовательно, Random Forest можно использовать на практике как классификатор страниц soft 404.

Алгоритм	точность	полнота
“Слабый” алгоритм + k-NN (k=10)	0,94	0,45
k-NN(k=150)	0,92	0,66
Id3	0,56	0,90
Random Forest (порог = 0.5, 50% голосующих за soft 404 деревьев)	0,89	0,98
Random Forest (порог = 0.5, 80% голосующих за soft 404 деревьев)	0,97	0,94

4 Заключение

В работе предложен новый подход к проблеме определения страниц soft 404. Подход состоит в представлении страницы в виде вектора параметров слов, встречающихся на странице и присутствующих в специальном словаре. Словарь содержит слова из русского и английского языков, наиболее часто встречающиеся на странице soft 404. Показано, что применение алгоритма Random Forest позволяет получить алгоритм классификации страниц soft 404 с высокими показателями полноты и точности, что позволяет использовать данный алгоритм в поисковой системе на стадии скачивания страниц перед стадией индексирования и для распознавания страниц soft 404.

В дальнейшем планируется провести сравнительный анализ предложенных алгоритмов и алгоритма, рассмотренного в работе [1]. В данной работе этого не удалось сделать, потому что коллекция ROMIP BY.Web была составлена в 2007 году, с того времени состояние страниц изменилось. А для оценки производительности алгоритма [1] требуется постановка эксперимента, позволяющего скачивать страницы из интернета.

В данной работе обучение проводилось на векторах TF, в будущем планируется провести обучение предложенных алгоритмов на векторах BM25 [15]. Также планируется уменьшить размер получаемой модели Random Forest (1,5 Гб оперативной памяти) за счет корректировки используемых признаков.

Также возможно создание словарей слов, относящихся к soft 404, на языках, отличных от русского и английского.

Литература

[1] Bar-Yossef Z., Kumar R., Broder A.Z., Tomkins A. Sic transit gloria telae: towards an understanding of the Web’s decay// Proc. of the 13th Int. WWW Conf., 2004.

[2] Francisco-Revilla L., Shipman F., Furuta R., Karadkar U., Arora A.. Managing change on the web// JCDL’01: Proc. of the 1st ACM/IEEE-CS Joint Conf. on Digital libraries, 2001. – P. 67-76.

[3] Ntoulas A., Cho J., Olston C. What’s new on the web? The evolution of the web from a search engine perspective// Proc. of the 13th Int. World Wide Web Conf., 2004.

[4] Baeza-Yates R., Castillo C., Efthimiadis E. Characterization of national web domains//ACM TOIT, 2006.

[5] Pitkow J.E. 1999. Summary of WWW characterizations// World Wide Web. – 1999. – V. 2, No 1-2. – P. 3-13.

[6] Bar-Yossef Z., Keidar I., Schonfeld U. Do not crawl in the dust//WWW, 2006.

[7] Edelman B. Domains reregistered for distribution of unrelated content: A case study of “Tina’s Free Live Webcam”. – http://cyber.law.harvard.edu/people/edelman/renewals/, 2002.

[8] Brin S., Page L. The anatomy of a large-scale hypertextual web search engine, 1998. – http://infolab.stanford.edu/~backrub/google.html.

[9] Broder A., Glassman S., Manasse M., Zweig G. Syntactic clustering of the web// Proc. of the Sixth

Int. World Wide Web Conf., 1997. – P. 391-404. – http://www.std.org/~msm/common/clustering.html.
[10] Описание коллекции ROMIP BY.Web 2007. – http://romip.ru/ru/collections/by.web-2007.html.
[11] Bootstrap Sampling Numerical Example. – http://people.revoledu.com/kardi/tutorial/Bootstrap/examples.htm.
[12] Breiman L., Cutler A. Random forests. – http://stat-www.berkeley.edu/users/breiman/ RandomForests/cc_home.htm.
[13] Mitchell T.M. The basic decision tree learning algorithm// Machine Learning. – McGraw-Hill Science/ Engineering/Math press, 1997. – P. 55-60.
[14] Breiman L. Bagging predictors//Technical Report No. 421, 1994.
[15] BM25. – http://xapian.org/docs/bm25.html.

A statistical approach to solving the problem of identification of soft 404 pages

S.S. Chirkov

The problem of identification soft 404 pages is actual problem of modern search engines. First of all, this is due to the fact that the 404 page soft cannot be determined by the code of the protocol HTTP, as is the case with the usual 404 pages. Previously proposed solutions [1, 2] of finding soft 404 pages could not completely solve this problem. This article presents a new approach to identification soft 404 pages based on the presentation of pages in the form of sets of words (phrases) and the use of machine learning algorithms to assess the proximity of these sets.

[i] Порогом для Random Forest является величина, при достижении которой, каждое отдельное дерево классифицирует страницу как soft 404. В данном случае эта величина является вероятностью