

Яндекс

Офлайн-поиск на основе MapReduce

Сибиряков Александр
Разработчик отдела качества поиска

Я.Субботник, Екатеринбург, 2 июля 2011 года

MapReduce

Операции фреймворка

- Расчет IDF
- Построение малого индекса
- Поиск

Расчет IDF

- MAP: делим текст ссылки на слова, приводим к нормальным формам
- REDUCE: считаем умноживая по каждому термину и воз-во документов
- сохраняем в файли и раскладываем при поиске



Построение малого индекса

- фильтрация ссылок по словам запроса
- расчет векторных статистических факторов:
 - длина документа
 - общее количество слов
 - количество уникальных "слов" в документе

Делается за один MAP, меньше в 100-1000 раз полей базы.

Поиск

- по малому индексу
- для каждого ресурса
- двукратный проход по всем ссылкам
- токенизация, расчет факторов текстовой релевантности (TR)

Поиск

Проблемы создания ранжированного

- сложно собрать и анализировать
- много промежуточных программ, и много точек синхронизации
- дублирование
- данные расходятся, темпы сбоя...
- много запросов к базам данных
- вводится время на все вычисления



- ~1000 ссылок с главной страницы lenta.ru

Ru-net

- 350 млрд. хостов
- 1,5 трлн. ссылок
- ... и растут!

Производительность

- Время работы 15 минут
- 670 тыс. ресурсов *
- 267 Мб *

* - объем малого индекса



Profit

- разработка и тестирование новых факторов ранжирования
- отслеживание изменений в существующих
- оптимизация формулы
- ... и другие поисковые эксперименты



Александр Сибиряков
Разработчик отдела качества поиска
alexander@yandex.com



Лента.ru

- 1000 ссылок с главной
- 1000 ссылок с главной
- 1000 ссылок с главной
- 1000 ссылок с главной
- 1000 ссылок с главной
- 1000 ссылок с главной
- 1000 ссылок с главной
- 1000 ссылок с главной

Оценки от IDF

- 1000 ссылок с главной
- 1000 ссылок с главной
- 1000 ссылок с главной
- 1000 ссылок с главной
- 1000 ссылок с главной
- 1000 ссылок с главной
- 1000 ссылок с главной
- 1000 ссылок с главной

Итого: 670 тыс. ресурсов

- 670 тыс. ресурсов
- 670 тыс. ресурсов
- 670 тыс. ресурсов
- 670 тыс. ресурсов
- 670 тыс. ресурсов
- 670 тыс. ресурсов
- 670 тыс. ресурсов
- 670 тыс. ресурсов

```

import java.io.*;
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.*;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.util.StringUtils;
import org.apache.hadoop.mapreduce.Mapper.MapperContext;
import org.apache.hadoop.mapreduce.Reducer.ReduceContext;

public class MapReduce {
    public static void main(String[] args) throws Exception {
        Configuration conf = new Configuration();
        FileInputFormat.setInputPaths(conf, args[1]);
        FileOutputFormat.setOutputPath(conf, args[2]);
        MapReduce mapReduce = new MapReduce(conf);
        mapReduce.run();
    }
}

class MapReduce {
    private Configuration conf;

    MapReduce(Configuration conf) {
        this.conf = conf;
    }

    void run() throws Exception {
        MapReduceMapper mapper = new MapReduceMapper(conf);
        MapReduceReducer reducer = new MapReduceReducer(conf);
        Job job = Job.getInstance(conf, "MapReduce");
        job.setMapperClass(mapper);
        job.setReducerClass(reducer);
        FileInputFormat.addPathToJob(job, "input");
        FileOutputFormat.setOutputPath(job, "output");
        job.waitForCompletion(true);
    }
}

class MapReduceMapper extends Mapper<Text, Text> {
    private Configuration conf;

    MapReduceMapper(Configuration conf) {
        this.conf = conf;
    }

    @Override
    public void map(Text key, Text value, MapperContext context) throws IOException, InterruptedException {
        String[] words = value.toString().split(" ");
        for (String word : words) {
            context.write(new Text(word), new Text("1"));
        }
    }
}

class MapReduceReducer extends Reducer<Text, Text, Text, Text> {
    private Configuration conf;

    MapReduceReducer(Configuration conf) {
        this.conf = conf;
    }

    @Override
    public void reduce(Text key, Iterable<Text> values, ReduceContext context) throws IOException, InterruptedException {
        long sum = 0;
        for (Text value : values) {
            sum += Long.parseLong(value.toString());
        }
        context.write(key, new Text(sum + ""));
    }
}
    
```



File	Size	Count	Sum
1	1000	1000	1000
2	1000	1000	1000
3	1000	1000	1000
4	1000	1000	1000
5	1000	1000	1000
6	1000	1000	1000
7	1000	1000	1000
8	1000	1000	1000
9	1000	1000	1000
10	1000	1000	1000
11	1000	1000	1000
12	1000	1000	1000
13	1000	1000	1000
14	1000	1000	1000
15	1000	1000	1000
16	1000	1000	1000
17	1000	1000	1000
18	1000	1000	1000
19	1000	1000	1000
20	1000	1000	1000
21	1000	1000	1000
22	1000	1000	1000
23	1000	1000	1000
24	1000	1000	1000
25	1000	1000	1000
26	1000	1000	1000
27	1000	1000	1000
28	1000	1000	1000
29	1000	1000	1000
30	1000	1000	1000

Я
Александр Сибиряков
Разработчик отдела качества поиска
alexander@yandex.com

Яндекс

Офлайн-поиск на основе MapReduce

Сибиряков Александр

Разработчик отдела качества поиска

Я.Субботник, Екатеринбург, 2 июля 2011 года

MapReduce



map: $\text{list}(k1, v1) \longrightarrow \text{list}(k2, v2)$

reduce: $\text{list}(k2, \text{list}(v2)) \longrightarrow \text{list}(k3, v3)$

операции reduce всегда предшествует
сортировка по ключу

Отличия от SQL

- нет возможности обратиться к одной записи
- нет индексов
- нельзя хранить промежуточное состояние в рамках операции
- акцент на обработке всей таблицы целиком

Наши особенности

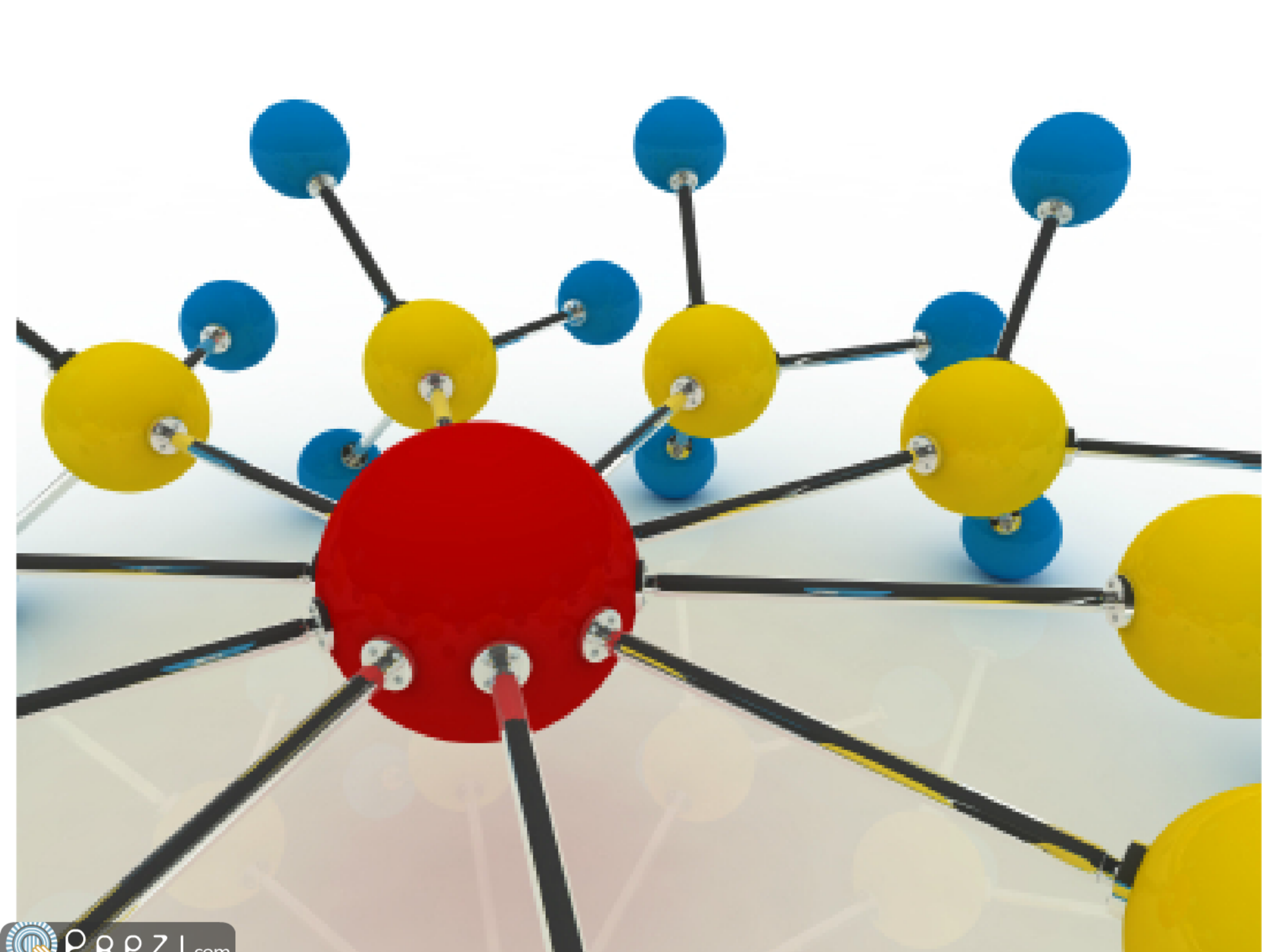
- собственная легкая реализация (C++, ~50 тыс. строк)
- 500 машин кластер
- 2,5 Пб данных
- 3X репликация
- транзакционность

см. доклад А. Дмитриева на YaC 2010

ПОИСК

Проблемы ссылочного ранжирования

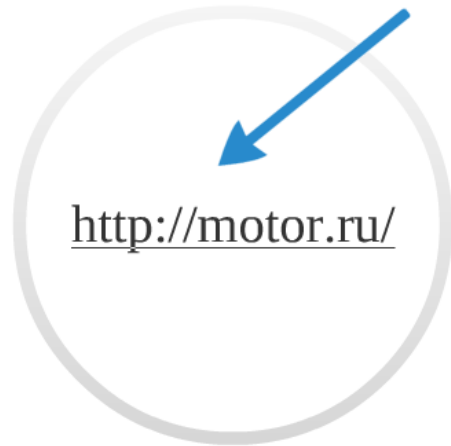
- сложно собирать поисковую базу
- много оптимизированных на производительность программ, которые сложно модифицировать:
 - дубликаты
 - pagerank
 - расчеты весов для спец. типов ссылок
 - замена локальных id на кластерные
 - подсчет весов слов в текстах ссылок



Url-источник



<http://lenta.ru/>



Url-получатель



<http://lenta.ru/news/2011/06/20/skypefire/>

Ресурс (Url-источника)

<http://www.lenta.ru>

- Статические: PR страницы, PR морды, кол-во страниц от морды ...
- Динамические: $TF*IDF$ по разным хитам, расстояния между словами, суммарный IDF слов ...
- Мета: URL, время появления, время послед. обхода, тип контента, длина документа, ...


```
▼<td class="glavnoe2" width="53%" rowspan="2">
  ▼<table class="top10" cellspacing="0" cellpadding="0" width="100%" border="0">
    ▼<tbody>
      ▼<tr valign="top">
        ▶<td class="last" width="38%">...</td>
        ▶<td class="top9" width="31%">...</td>
        ▼<td class="top9" width="31%">
          ▼<div>
            <span class="dt">24.06.06:55</span>
            ▼<span>
              "На Москве-реке баржа "
              <a href="/news/2011/06/24/ship/">врезалась в теплоход с выпускниками</a>
            </span>
          </div>
          ▶<div>...</div>
          ▶<div>...</div>
          ▶<div>...</div>
        </td>
      </tr>
    </tbody>
  </table>
</td>
▶<td class="pravaya" width="33%">...</td>
```

Описание ссылки

- Url-получателя (<http://lenta.ru/news/2011/06/20/skypefire/>)
- Время создания (2011-06-20 11:21)
- Url источника (<http://lenta.ru/>)
- Текст («Skype уволил нескольких топ-менеджеров»)
- Тип извлечения (PRE_LINK_TEXT)

- ~1000 ссылок с главной страницы lenta.ru

Ru-net

- 350 млрд. хостов
- 1.5 трлн. ссылок
- ... и растет!

Операции фреймворка

- Расчет IDF
- Построение малого индекса
- Поиск

Расчет IDF

- MAP: делим тексты ссылок на слова, приводим к начальным формам
- REDUCE: считаем упоминания по каждому термину и кол-во документов
- сохраняем в файл и раскладываем при поиске



ПОЛЕЗНЫЕ
ССЫЛКИ

Построение малого индекса

- фильтрация ссылок по словам запроса
- расчет некоторых статических факторов:
 - доля дубликатов
 - общее кол-во ссылок
 - кол-во уникальных "владельцев"
 - ...

Делается за один МАР, меньше в 100-1000 раз полной базы.

Поиск

- по малому индексу
- для каждого ресурса
- линейный проход по всем ссылкам
- токенизация, расчет факторов текстовой релевантности (TR)

[MROffline](#) >> [август - Конфиг 1](#) >> [август](#) >> [iphone \(2010-09-01 00:00:00, 213\)](#) >> [TTopCollector.ByRelevance](#)

[Prev](#) - [Next](#)

Results 1 - 10 of 10:

1. [iPhone. Новости, обзоры, тесты: Webter выпу...](#)
iPhone. Новости, обзоры, тесты: Webter выпустил официальное приложение - <http://www.iphones.ru>
[\[+\] show factors](#) | [\[+\] show citations](#) | [\[+\] show explanation](#)
2. [alexvolkov: Porsche для iPhone ...](#)
alexvolkov: Porsche для iPhone - http://vipclass.org/news/2010-05-13_ofitsialnoe-prilozhenie-porsche-dlya-iphone
[\[+\] show factors](#) | [\[+\] show citations](#) | [\[+\] show explanation](#) (0.387747)
3. [Porsche для iPhone ...](#)
Porsche для iPhone -dlya-iphone - http://vipclass.org/news/2010-05-13_ofitsialnoe-prilozhenie-
[\[+\] show factors](#) | [\[+\] show citations](#) | [\[+\] show explanation](#) (0.387747)
4. [Neznashka: Let's play We Rule on the iPhone! My gamename is...](#)
Neznashka: Let's play We Rule on the iPhone! My gamename is 'alexeifilonov'. - <http://phobos.apple.com/WebObjects/MZStore.woa/wa/viewSoftware?id=339274852>
[\[+\] show factors](#) | [\[+\] show citations](#) | [\[+\] show explanation](#)
5. [alexvolkov: Приложение для iPhone/iPad от BMW ...](#)
alexvolkov: Приложение для iPhone/iPad от BMW - http://vipclass.org/news/2010-05-28_bmw-v-vashem-ipad-ili-iphone
[\[+\] show factors](#) | [\[+\] show citations](#) | [\[+\] show explanation](#) (0.387747)
6. [DeepApple: Google переходит с W...](#)
DeepApple: Google переходит с Windows на Маки и Linux -



6. [DeepApple: Google переходит с W...](#)

DeepApple: Google переходит с Windows на Маки и Linux -

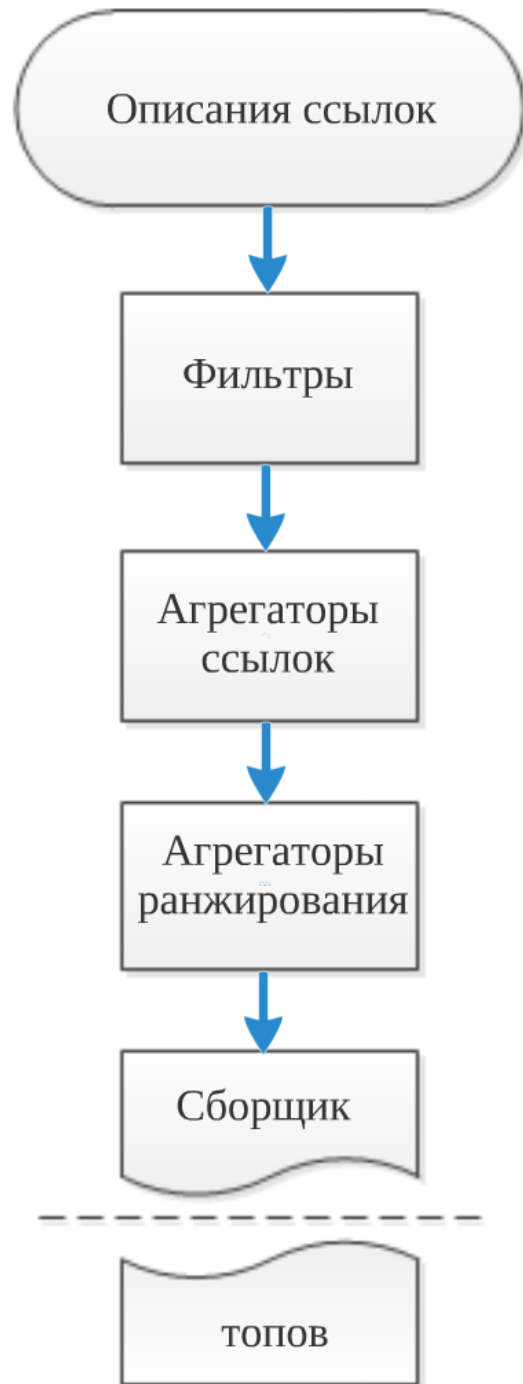
<http://deepapple.com>

[\[+\] show factors](#) | [\[+\] show citations](#) | [\[+\] show explanation](#)

SortFactor	RELEVANCE	TOTAL_QUOTES
0.033416	0.033416	697.000000

Date	Owner	Text	Type	Weight	Textmatch
20100402		DeepApple: Глюк Apple.com: в Купертино готовят релиз Linux-версии iTunes	Web	0.000900	0.055201
20100524		DeepApple: Первая статистика от Steam: на Мак-пользователей приходится 11 процентов продаж	Web	0.000900	0.052669
20100513		DeepApple: iPhone 4G всплыл во Вьетнаме	Web	0.000900	0.101464
20100421		DeepApple: DeepTip: кто слушает мои iTunes?	Web	0.000900	0.051672
20100421		DeepApple: Apple запускает программу бесплатной замены наушников у плееров iPod shuffle...	Web	0.000900	0.050138
20100426		DeepApple: «Хакси» возвращаются! Снова.	Web	0.000900	0.094631
20100428		DeepApple: DeepArment Store: На складе компьютеры MacBook Air, а также приводы...	Web	0.000900	0.041510
20100506		DeepApple: Распродажа «красивых» приложений под Мак	Web	0.000900	0.083770
20100429		DeepApple: Apple анонсирует дату начала работы WWDC 2010!	Web	0.000900	0.049310
20100525		DeepApple: Apple приступила к съемкам рекламы нового iPhone?	Web	0.000900	0.145358
20100604		DeepApple: Поехали! Apple начала развешивать банеры WWDC на здании Moscone Center!	Web	0.000900	0.045040
20100421		DeepApple: Очередной патент Apple может указывать на бюджетный настольный Мак	Web	0.000900	0.056214
20100410		DeepApple: Joy of Tech: вчерашнее главное	Web	0.000900	0.049583
20100607		DeepApple: Доля iPhone втрое превышает долю Android на рынке США	Web	0.000900	0.124700
20100507		DeepApple: Распродажи недели: билеты на WWDC и iPad 3G	Web	0.000900	0.070076

MAP



REDUCE

Агрегатор ссылок

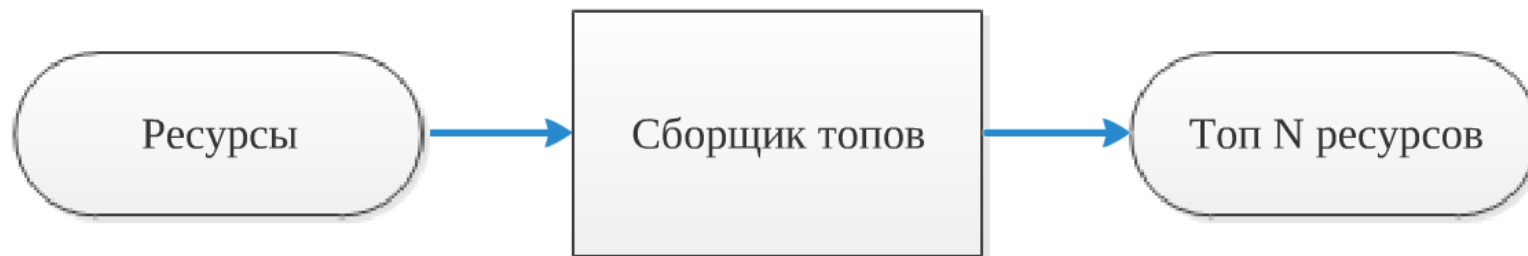


* - по ресурсу

Агрегатор ранжирования



Сборщик топов



Параметры:

- Ранжирующий фактор
- Кол-во документов (N)
- Факторы для вывода

<Experiment>

Name : march_april_2011
ReqFile : requests.txt
Tables : ^fresh_base/index\$
IndexName : 2011-05-23,20:35:0111
IndexAggregatorName : TPipeQuoteAggregator.CalcStaticFeatures
MorphoFixList : /place/home/mirzman/arcadia_tests_data/wizard/language/langdiscr.lst

<Filter>

Name : TSimpleFilter
OwnerType : Web
MatchType : allwords

</Filter>

<QuoteAggregator>

Name : TPipeQuoteAggregator.CalcRelevance
FilterName : TSimpleFilter.FilterWeb
RunScript : calc_relevance.awk
OutputFactors : NAIVE_RELEVANCE IDF_LIN_RELEVANCE IDF_COS_RELEVANCE IDF_BY_IDF_RELEVANCE PHRASE_MATCH

</QuoteAggregator>

<RankAggregator>

Name : TPipeRankAggregator.SampleRankAggregator
RunScript : rank_agg_sample.awk
InputFactors : IDF_BY_IDF_RELEVANCE NAIVE_RELEVANCE IDF_LIN_RELEVANCE IDF_COS_RELEVANCE PHRASE_MATCH
OutputFactors : FINISH_FACTOR

</RankAggregator>

<TopCollector>

Name : TTopCollector.ByRelevance
DocCount : 10
FactorName : FINISH_FACTOR
OutputFactors : NAIVE_RELEVANCE IDF_LIN_RELEVANCE IDF_COS_RELEVANCE IDF_BY_IDF_RELEVANCE PHRASE_MATCH

</TopCollector>

</Experiment>

Производительность

- Время работы 15 минут
- 670 тыс. ресурсов *
- 267 Мб *

* - объем малого индекса

Profit

- разработка и тестирование новых факторов ранжирования
- отслеживание изменений в существующих
- оптимизация формулы
- ... и другие поисковые эксперименты

Машина времени



Машина времени

- Храним snapshot-ы баз за период времени
- Можно делать поиск по любому snapshot-у
- Фильтрация ссылок по дате появления ссылки

Я

Сибиряков Александр

Разработчик отдела качества поиска

sibiryakov@yandex-team.ru