# DataparkSearch* at TREC-2005

**Maxim Zakharov**
OOO Datapark,
P.O.Box 35,
Sochi, 354000, Russia
maxime@datapark.ru

**Abstract**

This paper describes the experiments of the OOO Datapark in TREC-2005. We participated in the Genomics track and submitted official runs to the Adhoc retrieval task. Our goal is to compare two methods of relevance calculation uses in the DataparkSearch Engine.

## 1 Introduction

For TREC-2005 we participated in the Genomics track. Our Adhoc retrieval work used the DataparkSearch Engine, version 4.32. This is an open sources search engine released under the GNU General Public License and designed to organize search within a web site, group of web sites, intranet or local system. The DataparkSearch Engine can be build with one of two a little different methods of relevance calculation: a fast and a full method. We assume, that a full method can provide better results, while a fast method are works faster. Our goal is to compare these methods on a real retrieval tasks.

## 2 DataparkSearch methods of relevance calculation

In indexing, DataparkSearch divide every document onto sections. A section is any part of document, for example, for HTML documents this may be TITLE or META Description tags. For TREC-2005 genomics data we use all MEDLINE fields[1] as sections of document.

In addition to sections, some factors of document are counts also in relevance calculation: the average distance between query words, the number of query word occurrences in document, the position of first occurrence of a query word in document, the difference between the distribution of query word counts and the uniform distribution.

In searching, DataparkSearch compare every document found against an "ideal" document. The "ideal" document should have query words in every section defined and also should have the predefined values for addition factors.

*http://www.dataparksearch.org/

[1]http://www.ncbi.nlm.nih.gov/entrez/query/static/help/pmhelp.html#MEDLINEDisplayFormat

In our experiments, every query word has been automatically expanded using ispell data to produce all word forms, and using synonyms and acronyms and abbreviations lists[2]

### 2.1 A full method of relevance calculation

Let $x$ is weighted sum of all sections. In TREC-2005 experiments we assume all sections have weight equal to 1. Let $y$ is the weighted sum of differences between additional factors of document found and corresponding values of "ideal" document. And let $xy$ is the weighted sum of sections where at least one query word has been found. Then value of relevance of documents is calculated as:

$$\frac{1}{2} \times \frac{x + xy}{x + y}$$

### 2.2 A fast method of relevance calculation

Let $x$ is number of bits used for weighted values of all sections. In TREC-2005 experiments we assume all sections have weight equal to 1 Let $y$ is the weighted sum of differences between additional factors of document found and corresponding values of "ideal" document. And let $xy$ is the number of bits where weighted value of sections of "ideal" document are different to weighted value of sections of document found. Then value of relevance of documents is calculated as:

$$\frac{x - xy}{x + y}$$

## 3 Results and Analysis

We submitted two runs for the Adhoc retrieval task, dpsearch1 for a fast method of relevance calculation, and dpsearch2 for a full method of relevance calculation. The results for those runs shown in following table:

| | dpsearch1 | dpsearch2 |
|---|---|---|
| Ret | 16786 | 16786 |
| Ret_rel | 1342 | 1362 |
| Avg. Prec(10) | 0.2551 | 0.2633 |
| Avg. Prec(100) | 0.1182 | 0.1231 |

As expected, a full method of relevance calculation give better results, but difference is not so big, thus, a fast method may be used for large collections to archive better searching speed.

[2]these lists can be downloaded from the Download section at http://www.dataparksearch.org/