# Federated Search in the Wild

## The Combined Power of over a Hundred Search Engines

Dong Nguyen[1], Thomas Demeester[2] , Dolf Trieschnigg[1], Djoerd Hiemstra[1]
[1] University of Twente, The Netherlands
[2] Ghent University - IBBT, Belgium
d.nguyen@utwente.nl, thomas.demeester@intec.ugent.be, {trieschn, hiemstra}@cs.utwente.nl

## ABSTRACT

Federated search has the potential of improving web search: the user becomes less dependent on a single search provider and parts of the deep web become available through a unified interface, leading to a wider variety in the retrieved search results. However, a publicly available dataset for federated search reflecting an actual web environment has been absent. As a result, it has been difficult to assess whether proposed systems are suitable for the web setting. We introduce a new test collection containing the results from more than a hundred actual search engines, ranging from large general web search engines such as Google and Bing to small domain-specific engines. We discuss the design and analyze the effect of several sampling methods. For a set of test queries, we collected relevance judgements for the top 10 results of each search engine. The dataset is publicly available and is useful for researchers interested in resource selection for web search collections, result merging and size estimation of uncooperative resources.

## Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Miscellaneous

## Keywords

Federated search, distributed information retrieval, evaluation, dataset, test collection, web search

## 1. INTRODUCTION

Web search has become the most popular way for finding information on the web. The general web search engines use crawlers to populate their indices. However, a large part of the web, also called the *hidden* or *deep web*, is not easily crawlable [11]. Usually these pages are dynamic and only accessible through a search engine. A solution to this problem is *federated search*, also called *distributed information retrieval*. Queries are directly issued to search interfaces of collections, taking away the need to crawl these collections. Given a query, a broker selects suitable search engines. The query is then forwarded to these search engines. The broker gathers the results and creates a single ranked list. Examples of federated search on the web include vertical search, peer-to-peer networks and metasearch engines [12].

While federated search has been studied for many years and has many applications for the web, an appropriate dataset reflecting an actual web environment has been absent. So far, people have created artificial collections by dividing TREC collections [12], for example by topic or source. These collections are very different from actual search engines we find on the web, which have different retrieval methods, skewed sizes and heterogeneous content types (images, text, video etc.). As a result, it is not clear to what extent the findings so far in federated search hold in a web setting.

In this paper, we introduce a new dataset for federated search that we have made publicly available. The dataset contains result pages from 108 actual web search engines (such as Google, Yahoo, YouTube and Wikipedia). For each search engine, several query based samplings have been provided for resource selection. We also provide the responses and relevance judgements of their results for a set of queries. Results are annotated in two ways, by judging the snippet created by the engine and by judging the actual document. This dataset reflects an uncontrolled environment that can often be found in real federated search applications on the web. For example, the actual sizes of the resources, as well as the used retrieval algorithms are unknown. For researchers, the dataset is useful to evaluate resource selection, resource merging, and to experiment with size estimation of uncooperative resources. The dataset is available at http://www.snipdex.org/datasets.

We first discuss related work. We then describe the data collection and present analyses of the dataset. We conclude with a summary.

## 2. RELATED WORK

A federated search system presents three challenges: *resource description*, obtaining a representation of the resource, *resource selection*, selecting suitable collections for a query and *resource merging*, creating a single ranked list from the returned results [2]. In this section we will focus on resource description, because this is highly related to the construction of our dataset. In addition, we review existing test collections for federated search.

*CIKM'12,* October 29–November 2, 2012, Maui, HI, USA.
Copyright 2012 ACM 978-1-4503-1156-4/12/10 ...$10.00.

**Resource description**
The broker maintains a representation for each collection to facilitate the selection of suitable collections. In cooperative environments, in which each collection is willing to provide broker meta-information about its contents, these representations could contain the complete term statistics. In uncooperative environments, where collections do not provide meta-information about its contents or when this information cannot be trusted, the term statistics are typically approximated by sampling documents from the collection. Query-based sampling [3] involves selecting queries, sending them to the collection, and downloading the top $n$ documents for each query. These queries are usually single words sampled from the current representation. Another approach involves selecting queries from an external resource. This approach can give more representative samples, however it is less efficient since some queries might not return any results. It has been suggested that 300-500 documents are sufficient for sampling a collection [3]. Recently, snippets were used instead of documents, eliminating the need to download documents [16].

**Test collections**
The performance of resource selection methods was found to vary between test collections (e.g. [20], [14], [10]). Test collections for federated search have been constructed by reusing existing TREC datasets. For example, TREC disks 1 to 3 were divided into 100 collections based on source and publication date, and TREC4 was clustered using k-means into 100 collections. Recent datasets containing web pages were constructed by reusing the GOV2 and WT2G datasets and partitioning the pages by hosts. To simulate a more realistic environment, the datasets were modified. For example, to reflect collections with skewed sizes or relevant document distributions collections where collapsed together. To reflect collections with overlapping documents, sliding windows and grouping by queries were used [13]. Zhou et al. [19] created a test collection for aggregated search using the ClueWeb09, ImageCLEF and TRECVID collections. Thomas and Hawking [15] created a heterogeneous dataset for meta search, however the number of collections was small and each collection (regardless of size) had an equal amount of relevant documents.

Lack of a dataset reflecting a realistic web environment might have caused that only little research has been done on federated search for the web. In addition, the research done so far used data missing some important properties, making it unclear whether these results hold in a more realistic web environment. An exception is the work done by Arguello et al. [1]. However, their data was from a commercial search engine and not publicly available. In addition, they were interested in vertical search and therefore assumed a cooperative environment. Monroe et al. [8] examined query based sampling for web collections, by creating collections using pages in DMOZ. They found query based sampling to be effective for web collections. Craswell et al. [5] experimented with resource selection for the web. To simulate a web environment where search engines have different retrieval algorithms, resources used different retrieval methods, such as BM25 and Boolean ranking. Hawking and Thomas [6] evaluated resource selection methods in the GOV domain and proposed a hybrid approach combining distributed and central IR techniques. Although the work just discussed used actual web pages, the collections were relatively small compared to collections found on the web. For example, the largest collections were around 10 or 20 thousand documents [8, 6], or a couple of thousands [5]. Ipeirotis and Gravano [7] presented a new method for resource description and selection. One of their evaluations used 50 actual web search engines, however these were only shortly described and to our knowledge this dataset is not publicly available.

# 3. DATASET CREATION

In this section, we discuss collection of the dataset. We will motivate the new dataset, and describe how the resource sampling and annotation were performed.

## 3.1 Motivation

As outlined in the related work, most existing federated search collections were created using TREC collections. However, these do not sufficiently reflect real search engines on the web. We believe a dataset for federated web search should have the following properties:

- **Heterogeneous content**. To reflect the heterogeneity of content on the web, the dataset should contain different media types (text, images, videos). In addition, the text should be written in different styles, such as news, blogs, Q&A and general web pages.

- **Skewed sizes and relevance distributions**. Again reflecting the real web, the dataset should contain very small (e.g. very focused) as well as very large collections (e.g. general web search engines). There should also be a large variation in the average number of relevant documents available in collections.

- **Different retrieval algorithms**. Each search engine on the web has its own (usually hidden) retrieval algorithm. Preferably, we should not try to simulate web search retrieval algorithms, but instead use the existing ones.

- **Overlapping documents**. Once more, this reflects the nature of federated search on the web: a few large search engines have strongly overlapping indices, whereas a large number of smaller engines have little overlap.

## 3.2 Data Collection

### 3.2.1 Resources

To accommodate the requirements we selected 108 *real* search engines on the web for our dataset. A diverse selection was made, with varying sizes (from very large ones such as Google to small ones such as the website of a company), media types (web pages, images, videos) and domains (from general web search to specific domains like job search). The selection was made from search engines that are accessible through the OpenSearch[1] interface. To also include the major search engines that were not accessible through OpenSearch, an additional tool was used to scrape results [17]. A manual categorization of the resources is provided in Table 1.

[1] http://www.opensearch.org/

**Table 1: Categorization resources**

| Category | Count | Examples |
|---|---|---|
| General web search | 10 | Google, Yahoo, AOL, Bing, Baidu |
| Multimedia | 21 | Hulu, YouTube, Photobucket |
| Q & A | 2 | Yahoo Answers, Answers.com |
| Jobs | 7 | LinkedIn Jobs, Simply Hired |
| Academic | 16 | Nature, CiteSeerX, SpringerLink |
| News | 8 | Google News, ESPN |
| Shopping | 6 | Amazon, eBay, Discovery Channel Store |
| Encyclopedia/Dict | 6 | Wikipedia, Encyclopedia Britannica |
| Books & Libraries | 3 | Google Books, Columbus Library |
| Social & Social Sharing | 7 | Facebook, MySpace, Tumblr, Twitter |
| Blogs | 5 | Google Blogs, WordPress |
| Other | 17 | OER Commons, MSDN, Starbucks |

**Table 2: Example sample queries**

| Top | Zipf |
|---|---|
| travel | county |
| myspace | vineacity |
| videos | good |
| netflix | dureegauche |
| walmart.com | the |

### 3.2.2 Sampling

We used query based sampling [3] to obtain resource descriptions. The query-based samples and search results were collected between December 21, 2011 and January 21, 2012. We experimented with three methods to select queries for sampling: *Random*, *Top*, and *Zipf*. *Random* selects single terms randomly from the documents sampled so far. *Top* uses the most popular queries from a web search engine's query log [9]. *Zipf* uses single term queries taken evenly from the binned term distribution in ClueWeb09, where terms were binned on a log-scale of their document frequency ($df$) to ensure that there are queries from the complete frequency distribution. For each query, the top 10 results from each engine were retrieved. Sampling was done by collecting the snippets of 197 queries and downloading the actual documents of the first 40 queries. Example queries for *Top* and *Zipf*, which use the same queries for each resource, are presented in Table 2. *Random* uses different queries per resource.

### 3.2.3 Topics and search results

As test queries, we used 50 topics (Topics 51-100) from the 2010 TREC Web Track Ad Hoc task [4]. Every query was sent to each resource and the top 10 results were collected. An ID, URL, title, date, snippet and corresponding HTML page were stored. The results were stored in a standard format:

```
<snippet id="DT01-0115-1037-06">
  <origin pid="2e692a94a01e8c5dd3ec9cbb581798c6"/>
  <location cached="0115/random/1037-06.html">
    http://en.wiktionary.org/wiki/Asian_elephant
  </location>
  <title>Asian elephant</title>
  <found>2012-01-04 10:29:12</found>
  <summary>
    # An elephant, Elephas maximus, found in Asia.
  </summary>
</snippet>
```

### 3.2.4 Relevance judgements

The evaluation set was created using 10 judges, including IR researchers and outsiders, with the goal of obtaining relevance assessments both for the snippets and, independently, for the pages of the returned results. Many of the topics in the 2010 TREC Web Track are highly ambiguous. They come with a general information need (description) and several subtopics. All judgements were based only on the general information need (description) of a query. For example, for the ambiguous query *raffles*, the accompanied description is *Find the homepage of Raffles Hotel in Singapore.*

**Snippet judgements**
From potentially 54,000 results to be judged (if each search engine would have returned 10 results for all queries), there were in practice only just over 35,000 results. First, all snippets were judged. The snippets for one query were all judged by the same person. Most queries were judged by a single judge, but a few were judged by several, such that in total about 53,000 judgements were recorded. Per query, all gathered snippets were shown one by one in a random order, displaying their title, summary, preview (if present), and the page URL.

Given the snippet and information need for a query, the judges annotated the snippet with: *Junk*, *Non*, and *Unlikely*, *Maybe*, *Sure*, or *Answered*. The last category, where the particular information need had been answered by the snippet itself, appeared not applicable for the considered topics.

**Page judgements**
The process of judging pages turned out to be significantly slower than snippets, and therefore the page annotation task was organized as follows. We assume that snippets labeled *Junk* or *Non* do not represent relevant pages. Although we might miss some relevant documents due to this assumption, an actual user would not have clicked on the snippet anyway. Therefore, only pages for which the snippet was rated *Unlikely*, *Maybe*, or higher by at least one of the judges, were judged. That reduced the amount of required page judgements to only 28% of the total number of snippets.

Six levels of relevance were used, denoted in increasing order of relevance as *Junk*, *Non*, *Rel*, *HRel*, *Key*, and *Nav*, corresponding with the relevance levels from the Web TREC 2010 ad-hoc task [4]. The judgements were based on a snapshot from the page, taken simultaneously with the snippets. Additionally, the HTML data from the page had been crawled and was available to the judges. For each query, all pages for which a judgement was required, were shown and judged in a random order and by the same judge. The other pages were rated *Non*, by default[2].

Often, the same website appeared in the top 10 results from different search engines. Hence, in order to determine the reference judgement for each result, the URLs were first normalized to a standard form (e.g., omitting search engine specific additions, like *Sponsored*, or the query terms) and all judgements for the same normalized URL were considered together. The number of unique URLs amounts to 90.5% of the number of snippets. Moreover, for 11 out of 50 topics, we had all pages (and snippets) judged by at least two people. The different judgements corresponding to a specific URL were processed as follows. Each judgement received a score $s$, with value 0 (*Junk* or *Non*), 1 (*Rel*), 2 (*HRel*), 3 (*Key*), or 4 (*Nav*). The average $\bar{s}$ of these scores was determined. In this paper, we only use binary page relevance levels for evaluation. A page is considered relevant if the page is on average rated *Rel* or higher, otherwise non-relevant. An alternative, more demanding, criterion for relevance would be on average *HRel* or higher.

An extensive discussion of the reliability of test collections is found in [18], in which it is shown that the overall evaluation of retrieval systems is not very sensitive with respect to different annotators. The overlap between two different sets of relevance assessments, defined as the size of the intersection of the relevant document sets divided by the union of the relevant document sets for both judges, was found to lay between 0.42 and 0.49 for the TREC-4 collection used in [18].

For the current test collection, using the annotations by two different judges for 11 of the test topics, the average overlap is 0.43, quite similar to [18], confirming that this collection is equally apt for comparing retrieval systems. Note that, as in [18], the overlap varies significantly among the queries. This is due to the following reasons. First, we have instructed the judges to interpret the information need in a more general (*multimedia*) way, leading to different opinions on relevance for the results to some topics (e.g., when judging pictures). In addition, the background and interests of the judges have probably influenced their judgement. The worst case (overlap = 0.18) is topic 97, *south africa*, a highly ambiguous query, whereas the best annotator agreement (overlap = 0.69) was found for topic 74, *kiwi* (general information need: *'find information on kiwi fruit'*), being simple, hence less prone to subjective judgement.

### 3.2.5 Summary

As a summary, the dataset contains the following:

- 108 search engines, a diverse set varying in size and content type.
- Samplings for each resource using query based sampling. Queries were selected using three different methods (*Random, Top* and *Zipf*). For each method, snippets and documents were collected.
- Top 10 results of each resource for the 50 Web TREC 2010 queries.
- Relevance assessments on a six point scale for the top 10 results of each resource per TREC query.

Note that future systems will not be disadvantaged when being evaluated on this dataset, because we provide relevance judgements for all resources without pooling based on runs. We obtained search results over many billions of documents, undoubtedly much bigger than ClueWeb09. But, we do not need all actual documents to test federated search.

## 4. DATASET CHARACTERISTICS

We now discuss the characteristics of the dataset. All analyses are based on the relevance judgements using pages, assuming the relevance criterion *Rel* or higher.

### 4.1 Properties

We first discuss the dataset in the context of the desired properties outlined in the previous section.

**Heterogeneity** As shown in Table 1 the collections span a wide range of topics as well as content types. For example, there are 21 collections that focus on multimedia (e.g. video, sound, images, files in general), and the text genres range from news to blogs to Q & A.

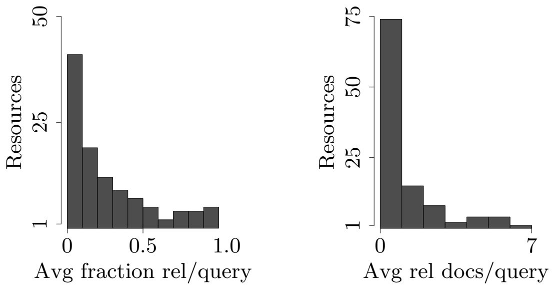
**Different retrieval algorithms** We do not need to simulate retrieval algorithms, but use the existing ones on the web. Most of them are (probably) specifically tailored to the specific collection they are searching.

**Relevance distribution** For each resource, we calculate the proportion of queries the resource has at least one relevant result for. A histogram is presented in Figure 1. The distribution is highly skewed, with many resources not containing any relevant document for most of the queries. The 10 top resources according to the proportion of queries the resources have at least one relevant result for, are exactly the 10 general web search engines as defined before. We observe the same trend in Figure 1b, which shows the average number of relevant documents per query. No resource has on average more than 6.1 (out of the first 10) relevant documents per query, suggesting that a good resource selection algorithm would be useful.

**Document overlap** Since we are dealing with actual search engines, we can only estimate the amount of overlap between them. We find that 34 collections had URL domains not shared with other collections in our samplings. These were collections that returned results from within their domain, such as CERN documents, CiteULike, Viddler and Wikispecies. Search engines with relatively high overlap are the ones in the category General web search (see Table 1).

**Skewed sizes** Although the actual collection sizes are unknown, it is clear that the collection sizes vary widely, with very larges ones such as Google, Bing and Mamma.com, and very small ones that are dedicated to a very specific topic.

[2] From those snippets that had been judged *Non* by one judge, but higher by at least one other judge, only 3% of the pages were judged *HRel*. The pages from those snippets are however expected to be more often relevant than the ones from snippets judged *Non* by each assessor. Therefore, it is expected that on average, less than 3% of the pages that were not judged, would correspond to a highly relevant page, which confirms that our assumption is acceptable.



(a) Avg fraction relevant/query (b) Avg number relevant docs

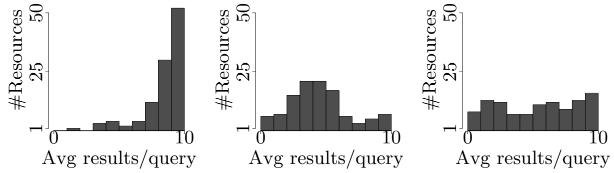
**Figure 1: Skewed relevance distributions**

## 4.2 Sampling analysis

In this section we analyze the query selection methods for query based sampling.

**Average number of results**

In Figure 2, histograms are shown of the average number of results returned for a query per resource. Using the *random* method, which selects queries depending on the resource, most queries have many results. However, for the *top* and *Zipf* methods, which select queries from an external resource, some of the resources return on average almost no results, resulting in a small amount of sampling documents from those resources.



(a) Random (b) Zipf (c) Top

**Figure 2: Histograms average number of results**

**Overlap between sampling queries and TREC queries**

We found no overlap in the *Zipf* and *top* queries with the test queries (Web TREC 2010). However, there were a total of 10 queries in *random* that matched exactly with one of the test queries such as *rice* (3x) or *iron* (2x). In addition there were more partial matches, for example *wall* (*the wall*), *sun* (*the sun*) etc. However, since the random queries are different for each resource, the total number of queries that matched is negligible (10 out of about 21,000 queries).

# 5. CONCLUSIONS

In this paper we presented a new test collection for federated search on the web, containing search results of over a hundred actual search engines including Google, YouTube, Wikipedia, Bing and many others. The dataset is publicly available and is useful for researchers interested in resource selection for web search collections, result merging and size estimation of uncooperative resources.

The construction of a second version of the data set is planned, with more samplings and more queries targeting specific collections or collection categories.

# 6. ACKNOWLEDGEMENTS

This research was partly supported by the Netherlands Organization for Scientific Research, NWO, grant 639.022.809 and the Folktales as Classifiable Texts (FACT) project, part of the CATCH programme, and partly by the IBBT (Interdisciplinary Institute for Broadband Technology) in Flanders.

# 7. REFERENCES

[1] J. Arguello, F. Diaz, J. Callan, and J.-F. Crespo. Sources of evidence for vertical selection. In *SIGIR 2009*, pages 315–322. ACM, 2009.

[2] J. Callan. *Advances in Information Retrieval.*, chapter Distributed information retrieval, pages 127–150. Kluwer Academic Publishers, 2000.

[3] J. Callan and M. Connell. Query-based sampling of text databases. *ACM Trans. Inf. Syst.*, 19:97–130, April 2001.

[4] C. L. A. Clarke, N. Craswell, I. Soboroff, and G. V. Cormack. Overview of the trec 2010 web track. In *TREC*, 2010.

[5] N. Craswell, P. Bailey, and D. Hawking. Server selection on the world wide web. In *Proceedings of the fifth ACM conference on Digital libraries*, DL '00, pages 37–46. ACM, 2000.

[6] D. Hawking and P. Thomas. Server selection methods in hybrid portal search. In *SIGIR 2005*, pages 75–82. ACM, 2005.

[7] P. G. Ipeirotis and L. Gravano. Distributed search over the hidden web: hierarchical database sampling and selection. In *VLDB 2002*, pages 394–405. VLDB Endowment, 2002.

[8] G. Monroe, J. French, and A. Powell. Obtaining language models of web collections using query-based sampling techniques. In *Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS'02)-Volume 3 - Volume 3*, HICSS '02, pages 67.2–, Washington, DC, USA, 2002. IEEE Computer Society.

[9] G. Pass, A. Chowdhury, and C. Torgeson. A picture of search. In *InfoScale 2006*. ACM, 2006.

[10] A. L. Powell and J. C. French. Comparing the performance of collection selection algorithms. *ACM Trans. Inf. Syst.*, 21(4):412–456, Oct. 2003.

[11] S. Raghavan and H. Garcia-Molina. Crawling the hidden web. In *VLDB 2001*. ACM, 2001.

[12] M. Shokouhi and L. Si. Federated search. *Foundations and Trends in Information Retrieval*, 5(1):1–102, 2011.

[13] M. Shokouhi and J. Zobel. Federated text retrieval from uncooperative overlapped collections. In *SIGIR 2007*, pages 495–502. ACM, 2007.

[14] L. Si and J. Callan. Relevant document distribution estimation method for resource selection. In *SIGIR 2003*, pages 298–305. ACM, 2003.

[15] P. Thomas and D. Hawking. Server selection methods in personal metasearch: a comparative empirical study. *Inf. Retr.*, 12:581–604, October 2009.

[16] A. S. Tigelaar and D. Hiemstra. Query-based sampling using snippets. In *Eighth Workshop on Large-Scale Distributed Systems for Information Retrieval, Geneva, Switzerland*, volume 630 of *CEUR Workshop Proceedings*, pages 9–14, Aachen, Germany, July 2010. CEUR-WS.

[17] R. B. Trieschnigg, K. T. T. E. Tjin-Kam-Jet, and D. Hiemstra. Ranking XPaths for extracting search result records. Technical Report TR-CTIT-12-08, Centre for Telematics and Information Technology, University of Twente, Enschede, March 2012.

[18] E. M. Voorhees. Variations in relevance judgments and the measurement of retrieval effectiveness. *Information Processing and Management*, 36:697–716, 2000.

[19] K. Zhou, R. Cummins, M. Lalmas, and J. Jose. Evaluating large-scale distributed vertical search. In *Proceedings of the 9th workshop on Large-scale and distributed informational retrieval*, LSDS-IR '11, pages 9–14. ACM, 2011.

[20] J. Zobel and J. A. Thom. Is CORI effective for collection selection? an exploration of parameters, queries, and data. In *in 'Proceedings of Australian Document Computing Symposium*, pages 41–46, 2004.