# Subwebs for Specialized Search

Raman Chandrasekar, Harr Chen, Simon Corston-Oliver, Eric Brill
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052, USA
{ramanc, harrc, simonco, brill}@microsoft.com

## ABSTRACT

We describe a method to define and use *subwebs*, user-defined neighborhoods of the Internet. Subwebs help improve search performance by inducing a topic-specific page relevance bias over a collection of documents. Subwebs may be automatically identified using a simple algorithm we describe, and used to provide highly-relevant topic-specific information retrieval. Using subwebs in a Help and Support topic, we see marked improvements in precision compared to generic search engine results.

## Categories and Subject Descriptors

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

## General Terms

Algorithms

## Keywords

information retrieval, customized search, subwebs

## 1. INTRODUCTION

General purpose web search engines are designed to provide broad coverage in their results. To scope a search to a particular topic, we could get more specific results using search sites specialized for that topic. But then we have to decide where to search, and how to rank results from diverse search sites. Ideally, we want search to utilize the breadth of the web, while providing the specificity of specialized sites.

We propose an approach based on the notion of subwebs. We conceptually partition the internet into small topic-specific neighborhoods, and restrict our search to one or more of these neighborhoods. In addition, we learn which parts of the neighborhood to prefer, and boost the ranks of results from these parts. For example, a generic search on the query *virus* would provide results for medical and computing senses. Subweb search could distinguish these senses.

Specifically, we view the World Wide Web as a collection of topic-specific subwebs, each of which is a collection of documents relevant to a given topic, e.g. cars, real estate, or celebrities. Subwebs may overlap, or one subweb may wholly contain another. A subweb can be described by a collection of domains (such as *http://microsoft.com*) or site paths (such as *http://www.geocities.com/mcsefreesite*), or an arbitrary collection of documents. The paths that describe the members of the subweb are weighted according to their relevance to the topic area.

We have a two step method of scoping search to specific topics: creating subweb definitions and using these definitions to rerank results from a generic search system. We define subwebs based on the intuition that the more often a path appears in web search results for topic-specific queries, the more relevant the path is to that topic. This approach identifies the frequent paths in the result sets of queries from a topic-specific log along with paths from the neighborhood of these results. The algorithm used to define a subweb is described here.

1. **Compute Result Path Distribution.** Queries from topic-specific query logs are sent to a search engine, and the frequency of paths in the search result set is computed. A set of seed URLs/paths, keyphrases extracted from relevant documents or even keyphrases specific to individual users may be used instead of query logs.

2. **Compute Neighborhood Path Distribution.** The one-link path neighborhood of the result set is computed from sites that are one link away: the URLs that the result sites point to ('outlinks'), and the URLs that point to the result set sites ('inlinks'). With some thresholding on the number of inlinks and outlinks used, the frequency of paths in the result neighborhood is computed. More generally, this could be computed for an N-link neighborhood.

3. **Compute Net Path Distribution.** The net distribution is determined by adding the frequency of path neighborhood to the result frequency, with some differential weighting.

4. **Normalize Against Baseline.** A normalized distribution is obtained by computing the net path distribution for a topic-specific query log, and subtracting the net path distribution for a baseline (random) set of queries. This normalization reduces the noise due to highly linked sites, and some advertising sites.

*SIGIR'04,* July 25–29, 2004, Sheffield, South Yorkshire, UK.
ACM 1-58113-881-4/04/0007.

The subweb definition, which is essentially a set of paths with weights proportional to their significance to the topic, is derived from the normalized distribution.

We assume that users will specify a query and the topic of their query. Alternatively, the topic of the query may be implicity inferred from the search entry point. The subweb definition corresponding to the search topic is used to rerank the search results obtained from a search engine. The reranking function we use is of the form
*NewRank(site) = f(SearchEngineRank(site), SubwebWeights(Paths(site))*
This function boosts sites based on the weights assigned to the paths subsumed in their URL, and maintains the original relative ranking of sites whose sub-paths are not in the subweb definition. In our initial experiments, *NewRank(site)* first ranked by subweb weights, and then by original search engine rank. We can display all results including boosted results, or display only those in the subweb definition.

## 2. EVALUATION

Intuitively, scoping search using additional information should improve result accuracy. We compared the relevance of results returned from three search engines: (1) MSN Search (http://search.msn.com), (2) MSN Search augmented with an HSC subweb ('Subwebs MSN'), and (3) Google (http://www.google.com).

A Help subweb was built from 450 queries from a Help and Support Center (HSC) query log, and normalized against a set of 1000 queries randomly sampled from an MSN Search query log. The HSC query log was obtained from end-user queries sent to the Microsoft Windows XP Help and Support Center. We used the top 20 results from MSN Search to build both the Help topic and baseline path distributions, and weighted the result distributions five times higher than the neighborhood distributions.

510 queries drawn from a held-out mix of frequent and random HSC queries were used to test this subweb. For each query, we got the top results from each of these search providers, and merged and deduplicated these to get 17,741 unique documents. These results were then presented in a random order to independent annotators in a double-blind manner. The annotators evaluated the documents from the perspective of Help and Support using a three-way relevance scale: {'Good', 'OK', 'Bad'}. These manual annotations formed the gold standard against which we report results.

The results clearly favor the use of subwebs. Subwebs MSN for this topic beat both regular MSN Search and Google on every measure of relevance by a substantial margin.

Regular MSN Search had an overall Mean Reciprocal Rank (MRR) of 0.22 for this domain. That is, on average, the first relevant result was returned in positions 4 or 5. Google had a similar MRR of 0.21. In contrast, Subwebs MSN had an overall MRR of 0.39, indicating that the first relevant result on average was in the 2nd or 3rd position, an improvement of 2 positions on average.

We computed Top N precision for N=1, and N=10, and the Mean Average Precision (MAP) for N=10. For multiple result lists, the Top N precision is the average of the Top N precisions of each result list. MAP is computed over relevant results; irrelevant results do not contribute to the average precision of a result list. The MAP value depends on the ranks of relevant results, while TopN precision depends only on the number of relevant results in a given set of results.

| | Regular MSN | Google | Subwebs MSN |
|---|---|---|---|
| Top 1 Prec./Good+Ok | 23.92% | 24.31% | 51.18% |
| Top 1 Prec./Good | 14.85% | 14.51% | 32.16% |
| Top 10 Prec./Good+Ok | 19.37% | 18.90% | 28.13% |
| Top 10 Prec./Good | 9.47% | 9.35% | 13.21% |
| Top 10 MAP/Good+Ok | 6.35% | 6.32% | 11.14% |
| Top 10 MAP/Good | 3.48% | 3.39% | 6.11% |

**Table 1: Relevance numbers**

In all these cases, we considered the case where a site marked either Good or OK was considered acceptable, and the case where only Good was acceptable. The results are presented in Table 1. The results for MSN and Google are comparable, but Subwebs MSN more than doubled the Top 1 precision, and improved the Top 10 Precision by over 40%, and the top 10 Mean Average Precision by over 75%.

## 3. DISCUSSION

We have described a simple and automated method to specialize search, and shown that it substantially improves result relevance. While generic search engines such as MSN Search are good in general, subweb search is better for specific topics. Subweb search helps the most with queries that use words or phrases whose ambiguity can be resolved with context, but offers less benefit for extremely specific queries.

Previous research has attempted to improve search results by identifying web communities [3] on the basis of link analysis [5]. [1] improves upon this, using topics generated from single queries while [2] uses user relevance feedback to augment the process. Another approach has been to use query expansion and structural methods to learn about specific domains [4]. Our work starts with a large set of queries, which makes the resulting subweb robust. In addition, the normalization that we perform against the baseline corrects for some of the problems of web-graph based methods.

## 4. REFERENCES

[1] K. Bharat and M. R. Henzinger. Improved algorithms for topic distillation in a hyperlinked environment. In *Proceedings of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval*, pages 104–111, Melbourne, AU, 1998.

[2] H. Chang, D. Cohn, and A. K. McCallum. Learning to create customized authority lists. In *Proc. 17th International Conf. on Machine Learning*, pages 127–134. Morgan Kaufmann, San Francisco, CA, 2000.

[3] D. Gibson, J. M. Kleinberg, and P. Raghavan. Inferring web communities from link topology. In *UK Conference on Hypertext*, pages 225–234, 1998.

[4] E. Glover, G. Flake, S. Lawrence, W. P. Birmingham, A. Kruger, C. L. Giles, and D. Pennock. Improving category specific web search by learning query modifications. In *Symposium on Applications and the Internet, SAINT*, pages 23–31, San Diego, CA, January 8–12 2001. IEEE Computer Society, Los Alamitos, CA.

[5] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.