How Large Is the World Wide Web?*

Adrian Dobra[1] and Stephen E. Fienberg[2]

[1] Institute of Statistics and Decision Sciences, Duke University, Durham NC 27708, USA, adobra@stat.duke.edu

[2] Department of Statistics and Center for Automated Learning and Discovery, Carnegie Mellon University, Pittsburgh PA 15213, USA, fienberg@stat.cmu.edu

Summary. There are many metrics one could consider for estimating the size of the World Wide Web, and in the present chapter we focus on size in terms of the number N of Web pages. Since a database with all the valid URLs on the Web cannot be constructed and maintained, determining N by counting is impossible. For the same reasons, estimating N by directly sampling from the Web is also infeasible. Instead of studying the Web as a whole, one can try to assess the size of the *publicly indexable Web*, which is the part of the Web that is considered for indexing by the major search engines.

Several groups of researchers have invested considerable efforts to develop sound sampling schemes that involve submitting a number of queries to several major search engines. Lawrence and Giles [8] developed a procedure for sampling Web documents by submitting various queries to a number of search engines. We contrast their study with the one performed by Bharat and Broder [2] in November 1997. Although both experiments took place almost in the same period of time, their estimates are significantly different.

In this chapter we review how the size of the indexable Web was estimated by three groups of researchers using three different statistical models: Lawrence and Giles [8, 9], Bharat and Broder [2] and Bradlow and Schmittlein [3]. Then we present a statistical framework for the analysis of data sets collected by query-based sampling, utilizing a hierarchical Bayes formulation of the Rasch model for multiple list population estimation developed in [6]. We explain why this approach seems to be in reasonable accord with the real-world constraints and thus allows us to make credible inferences about the size of the Web. We give two different methods that lead to credible estimates of the size of the Web in a reasonable amount of time and are also consistent with the real-world constraints.

1 Introduction

The World Wide Web (henceforth the Web) has become an extremely valuable resource for a wide segment of the world's population. Conventional sources of information (e.g. libraries) have been available to the public for centuries, but the Web has made possible what seemed to be only a researcher's dream: instantaneous access to journals, articles, technical report archives, and other scientific publications. Since

* This chapter is an edited version of a paper presented at Interface 2001 [4].

almost anybody can create and "publish" Web pages, the Web has no coherent structure and consequently it is not easy to establish how much information is available. Evaluation of the size and extent of the Web is difficult not only because of its sheer size, but also because of its dynamic nature. We have to take into account how fast the Web is growing in order to obtain credible estimates of its size. *Growth* in this context is "an amalgam of new Web pages being created, existing Web pages being removed, and existing Web pages being changed" [11]. As a result, any estimate of the size of the Web will be time dependent.

The Web consists of text files written in HyperText Markup Language (HTML). A HTML file contains special fields called *anchor tags*, which allow an author to create a *hyperlink* to another document on the Web. When the user clicks on one of these fields, the Web browser loads the URL specified in the hyperlink, and thus the Web can be seen as a directed graph, $\mathcal{G}$, with HTML pages as vertices and hyperlinks as edges. Albert et al. [1] claim that the diameter of $\mathcal{G}$, defined as the mean of the number of URLs on the shortest path between any two documents on the Web, can be expressed as a linear function of the number of vertices N of the graph $\mathcal{G}$ on a logarithmic scale. Using the value of N found by Lawrence and Giles [9], they concluded that "two randomly chosen documents on the Web are on average 19 clicks away from each other". Unfortunately, very little is known about the underlying structure of this highly connected graph. As a consequence, there is no direct method of estimating N. The dimensions of a database with all possible URLs on the Web will be huge, and, even if we could construct a URL database, we cannot determine which URLs correspond to valid Web documents. Sampling directly from the Web is infeasible: without a list of URLs, known in sample surveys as a *frame*, either implicit or explicit, it is impossible to take a valid probability sample. Alternative methods are also problematic, e.g. the length of the random walks required to generate a distribution over a subset of the Web that is close to the uniform may be extremely large [2].

If we cannot study the Web as a whole, we can try to assess the size of the *publicly indexable Web*. The indexable Web [11] is defined as "the part of the Web which is considered for indexing by the major engines, which excludes pages hidden behind search forms, pages with authorization requirements, etc.". Search engines such as Northern Light, AltaVista, or HotBot might give the impression that it is very easy to locate any piece of information on the Web. Since "several search engines consistently rank among the top ten sites accessed on the Web" [9], it should be obvious that the search services are used by millions of people daily. However, studies show that the search engines cover "fewer than half the pages available on the Web" [8], and as time goes by, they increasingly fail to keep up with the expanding nature of the Web. Several estimates of the total number of pages [8] indicate that because of the rapid growth of the Web, the fraction of all the valid sites indexed by the search engines continues to decrease.

Search engines have the best Web crawlers, therefore it seems natural that we should try to exploit them. If we want to estimate the portion of the Web covered by the existent search engines, why shouldn't we use the search engines themselves? Lawrence and Giles [8] developed a procedure for sampling Web documents by

submitting various queries to a number of search engines. We contrast their study with the one performed by Bharat and Broder [2] in November 1997. Although both experiments took place almost in the same period of time, their estimates disagree. In Sect. 2 we show how the size of the Web was estimated by three groups of researchers: Lawrence and Giles [8, 9], Bharat and Broder [2] and Bradlow and Schmittlein [3]. In addition, we explain the discrepancy between the results they obtained. Sect. 3 outlines our approach for calculating a lower bound on the size of the Web based on the data collected by Lawrence and Giles in December 1997. Our objective is to develop a procedure that could be applied in real time, allowing us in the future to monitor the growth of the Web by calculating estimates at several points in time. In the last section we present two different methods that give credible estimates of the size of the Web in a reasonable amount of time and are also consistent with the real-world constraints.

2 Web Evaluation

We defined the indexable Web as that part of the Web that is *considered* for indexing by the major engines. We have to make an unequivocal distinction between the indexable Web and the union of the indices of all existent search engines. There are Web documents which might be indexed by a search engine, but, at a fixed time t_0, were not included in any index. Furthermore, pages with authorization requirements, pages hidden behind search forms, etc., are not compatible with the general architecture of the search engines, and it is unlikely that they will be indexed by any search engine in the near future. We can summarize these ideas as follows:

$$\boxed{\textbf{Web pages indexed at } t_0} \subset \boxed{\textbf{Indexable Web at } t_0} \subset \boxed{\textbf{Entire Web}}$$

where the above inclusions are strict. Based on the inferences we make about the size of the portion of the indexable Web covered by several popular search engines, our goal is to produce an estimate of the size of the whole indexable Web.

Although there is no direct method of counting the number of documents on the Web, the search services disclose the number of documents they have indexed. Unfortunately, counts as reported by the services themselves are not necessarily trustworthy. It is not clear the extent to which duplicate pages, aliased URLs, or pages which no longer exist, are included in the reported counts. Despite these problems, we can still use the self-reported counts as an approximate order of magnitude of the search engines' indices (cf. [8]).

If every single search service has a narrow coverage, the size of any index might offer only a very limited insight about the dimension of the indexable Web. Because any engine has some inherent contribution, the combined coverage of all of the existent engines would allow us to make better inferences about the size of the indexable Web.

2.1 Lawrence and Giles Study

Lawrence and Giles [8, 9] developed a procedure for sampling Web pages that does not necessitate access to any confidential database and that can be implemented with fairly modest computational resources using only public query interfaces. This procedure consists of running a number of queries on several search engines and counting the number of results returned by each engine.

Lawrence and Giles [8] studied six major, widely available full-text search engines, namely AltaVista, Infoseek, Excite, HotBot, Lycos and Northern Light. The experimenters selected 575 queries issued by scientists at the NEC Research Institute between 15 and 17 December 1997 and submitted those queries to the six search services. They retrieved all of the results accessible through every engine and for each document they recorded the search engines which were able to locate it. Lawrence and Giles implemented a number of reliability assessments because the data obtained in this way cannot be used as-is due to several experimental sources of bias.

Since search engines do not update their databases frequently enough, they often return URLs relating to Web documents that no longer exist or that may have changed and are no longer relevant to the query. Lawrence and Giles retained only those documents that could be downloaded, and then they removed duplicates including identical pages with different URLs. Some engines are case-sensitive and some are not, hence Lawrence and Giles did not use queries that contained uppercase characters.

Another potential problem is that the six search engines use various ranking algorithms to assess relevance. The Web documents served up by an engine should be perceived as the "best" matches as determined by the ranking procedure. Since relevance is difficult to determine without actually viewing the pages, ranking algorithms might seriously bias our findings. To prevent this from happening, Lawrence and Giles retained only the queries for which they were able to examine the entire set of results. Queries returning more than 600 documents (from all engines combined after the removal of duplicates) were discarded for the purposes of the analysis.

Lawrence and Giles provided us with data in the form of a 575×63 matrix. Each row contains the counts for an individual query. The first six columns are the number of pages found by AltaVista, Infoseek, Excite, etc. The next columns are the number of pages found by any two engines, then the pages found by any three engines, and so on. The last column contains the number of pages found by all six engines. Using the principle of inclusion and exclusion, we transformed the "raw" data into 575 cross-classifying tables of dimension $2^6 - 1$. Let $X_1, X_2, \ldots, X_6$ be categorical variables corresponding to AltaVista, Infoseek, Excite, HotBot, Lycos and Northern Light, respectively. Each variable has two levels: "1" stands for "found page" and "0" stands for "not found". Let $\mathcal{D}$ denote the set of all binary vectors of length 6. The contingency table for query k ($1 \leq k \leq 575$) can be expressed as $\mathcal{S}_k = \{x_s^k | s \in \mathcal{D}\}$. For a given query, we do not know how many pages were *not* found by all six engines, therefore all 575 tables have a missing cell, which can be interpreted as the difference between the "real" number of pages existing on the Web and the number of pages actually found by the six engines for the queries in question.

Table 1. Multiple list data for query 140 (S. Lawrence and C.L. Giles, private communication)

						Northern Light							
						yes				no			
						Lycos				Lycos			
						yes		no		yes		no	
						HotBot		HotBot		HotBot		HotBot	
						yes	no	yes	no	yes	no	yes	no
AltaVista	yes	Infoseek	yes	Excite	yes	1	0	2	0	0	0	1	0
					no	2	0	3	2	0	0	0	2
			no	Excite	yes	1	0	2	1	0	0	3	4
					no	1	3	0	8	2	0	3	19
	no	Infoseek	yes	Excite	yes	0	0	0	1	0	0	0	0
					no	0	0	1	1	0	0	5	4
			no	Excite	yes	0	0	0	1	0	0	4	22
					no	0	0	7	17	2	3	31	?

The quality of the analysis we want to perform depends on other factors which might or might not turn out to be significant. Web documents were added, removed, edited, and modified while the experimenters collected the data, hence the search results might also change. Search engines first look for the best matches within the segment of their index loaded in the main memory and only if the matches they found are not satisfactory, they expand the search to the rest of the database. This means that if we would submit the same query to the same search service at different times of the day, the set of results fetched might not be the same. Under some (nearly) improbable circumstances, a search engine might not return any documents present in other search engines' databases, in which case we would not be able to estimate the overlap between indices.

We examined the intersections between the sets of pages corresponding to the 575 queries aggregated over the six search engines and concluded that all the interactions between two queries appear to be significantly smaller than any set of pages matching a query. The segments of the Web defined by the 575 queries are for all practical purposes disjoint, hence we can view our data as a seven-way contingency table $\mathcal{S}$ in which "query" is a multilevel stratifying variable.

We determined which engine performs better than the others. The *relative coverage* [11] of a search engine is defined by the number of references returned by a search service divided by the total number of distinct references returned. Notice that we can compute this ratio without having to estimate the missing cell. Table 2 shows our calculations of the relative coverage for the six search services considered. HotBot appears to have the largest coverage, followed by AltaVista and Northern Light. Although relative coverage can express the quality of a search service with respect to the others, it cannot be used if we want to find a way to measure the *combined* coverage of the six search engines with respect to the entire indexable Web.

By analyzing the overlap between pairs of search engines, one can easily calculate the fraction of the indexable Web covered by any of the six search engines. Since HotBot had reportedly indexed 110 million pages as of December 1997, Lawrence and Giles estimated that the absolute size of the indexable Web should be roughly 320 million pages. We discuss in detail the validity of their estimate in Sect. 2.4 as part of our reanalysis of their data. In February 1999, Lawrence and Giles [9] repeated

Table 2. Estimated relative coverage of the six search engines employed

Service	Coverage
HotBot	52.02%
AltaVista	37.23%
Northern Light	26.39%
Excite	19.11%
InfoSeek	13.24%
Lycos	4.15%

their experiment. The number of search engines was increased to 11 (AltaVista, Euroseek, Excite, Google, HotBot, Infoseek, Lycos, Microsoft, Northern Light, Snap and Yahoo) and the number of queries was expanded to $1,050$, hence the data this time consists of a $1,050 \times (2^{11} - 1)$ array. The experimenters did not make clear whether the 575 queries used for the first study were among the $1,050$ queries used for the second one. Northern Light had indexed 128 million pages at the time of the experiments, hence Lawrence and Giles approximated that there were 800 million pages on the indexable Web. The estimation method was similar to the one employed in the previous study. Unfortunately, their analysis was done dynamically and the new data were not retained for possible reanalyses.

2.2 Bharat and Broder Study

In November 1997, Bharat and Broder [2] performed an analysis analogous in many respects to the one carried out by Lawrence and Giles. They employed only four engines, i.e. AltaVista, Excite, Infoseek and HotBot. Instead of measuring directly the sizes and overlaps of the four search services, their approach involved generating random URLs from the database of a particular search engine and checking whether these pages were also indexed by the other search services.

The experimenters approximated sampling and checking through queries. Rather than choosing queries made by real users, Bharat and Broder randomly generated their own queries. The queries were derived from a lexicon of about 400,000 words built from 300,000 documents existing in the Yahoo! hierarchy. The artificially generated queries were presented to one search service and the search results were retrieved. Since it is very hard to get a hold of the entire set of results, Bharat and Broder picked a URL at random from the top 100 matches that were found for every query, hence the results were heavily dependent on the ranking algorithm used by every search

engine and also on the particular choice of lexicon. Both the ranking strategy and the lexicon can introduce serious experimental bias, that is, some documents will have better chances of being included in the sample than others.

For every query selected from one of the four indices, Bharat and Broder created a *strong query* intended to uniquely identify that particular page. They built the strong query by picking the most significant terms on the page and submitted it to the other search services. An engine $\mathcal{E}$ had indexed page $\mathcal{P}$ if $\mathcal{P}$ was present in the set of results fetched from $\mathcal{E}$. Because there is so much duplication on the Web, the set of results obtained might contain more than one document. It is not clear whether $\mathcal{E}$ would have found page $\mathcal{P}$ if the original query that generated $\mathcal{P}$ had been submitted to $\mathcal{E}$.

Bharat and Broder performed two series of experiments: trials 1 (10,000 disjunctive queries) and 2 (5,000 conjunctive queries) in mid-1997, and trials 3 (10,000 disjunctive queries) and 4 (10,000 conjunctive queries) in November 1997. We can see that the set of queries employed was considerably larger than the set used by Lawrence and Giles [8].

A more elaborate method than the one used by Lawrence and Giles [8, 9] was employed to assess what fraction of the indexable Web was covered by an individual search engine involved in the study. The experimenters calculated engine size estimates by minimizing the sum of the squared differences of the estimated overlaps between pairs of search engines. Since AltaVista reportedly indexed 100 million pages, Bharat and Broder concluded that the indexable Web had roughly 160 million pages in November 1997. We will come back with a detailed discussion of the validity of these results in Sect. 2.4.

2.3 Bradlow and Schmittlein Study

Another attempt to evaluate the Web was carried out by Bradlow and Schmittlein [3] during October 1998. They tried to assess the capability of six search engines (the very same engines employed in Lawrence and Giles [8]) to find marketing and managerial information using query-based sampling. Twenty phrases were chosen to be submitted to the search engines. The phrases had to be representative of the marketing world and also adequately precise (any number of pages could be relevant for an ambiguous query, hence our inferences could be adversely biased if too many relevant pages were found).

The six search engines combined returned $1,588$ different pages. For each of these pages, the experimenters recorded the binary pattern of length 6 describing what engines successfully detected the page (as before, "1" stands for "found page" and "0" for "not found"), the number of page links ($0-5$, $6-10$, or $10+$) and the domain type, indicating whether the site where the page was located was commercial (.com), academic (.edu), an organization (.org) or some other type of site ("other"). In addition, two phrase characteristics were also recorded – newer versus older and academic versus managerial – for more details see Bradlow and Schmittlein [3].

The originality of this approach comes in the way Bradlow and Schmittlein analyzed the data they collected. Each search engine and Web page are assumed to lie in a D-dimensional space. The probability that a given engine will capture some page is

a decreasing function of the distance between the engine and the page, hence a search engine is more likely to capture pages located in its immediate vicinity than pages that are situated at some considerable distance.

In the first model they proposed, they placed all the engines in the origin of a one-dimensional space ($D = 1$). Search engines tend to find the same Web pages, and consequently the "less resourceful" engines index only a subset of the pages indexed by the "more powerful" engines. The second model studied differs from the first one only with respect to the number of hypothesized dimensions of the underlying space; they took $D = 2$ to be a reasonable choice. Their third model is more flexible than the previous two because it allows the engine locations to vary in a two-dimensional space.

To be more specific, let p_{ijk} be the probability that the kth URL for the jth phrase is found by engine i. Moreover, d_{ijk} denotes a squared Mahalanobis distance between the location of the ith engine and the location of the kth URL for phrase j in the D-dimensional space. If u is "the rate at which the probability an engine finds a given URL drops off", we can express p_{ijk} as a function of d_{ijk} by

$$p_{ijk} = \frac{1}{1 + d_{ijk}^u}. \tag{1}$$

Bradlow and Schmittlein fit all three models using a Markov chain Monte Carlo sampler. The first two models were invalidated by the data, while the third seems to fit their data reasonably well. This is a clear indication that every search engine "carves out" its own location in the URL space.

Bradlow and Schmittlein [3] conclude that, for marketing/managerial queries, "the reader should feel confident that the search engines cover about 90% of what exists to be found for these kind of phrases". Although the authors argue that their modeling technique is superior to any other study performed and that "these kinds of marketing/management documents are relatively easy to locate", the result they came up with appears to conflict with what we know about search engine behavior. There are elements of their model and analyses, however, that would be worth further investigation as elaborations of the approach suggested in this chapter.

2.4 The Size of the Indexable Web

Here we describe explicitly the statistical models and inherent assumptions that underlie the estimates of Lawrence and Giles [8, 9], and Bharat and Broder [2].

Let $\mathcal{E}_1$ and $\mathcal{E}_2$ be two search engines with indices E_1 and E_2, respectively. Denote by Ω the complete set of documents available on the indexable Web. We make two major assumptions:

(A1) The indices E_1 and E_2 are samples drawn from a uniform distribution over Ω.

(A2) E_1 and E_2 are independent.

Denote by $|A|$ the number of elements of the set A. The first assumption says that

$$|\Omega| = \frac{|E_1|}{P(E_1)}, \tag{2}$$

while (A2) implies

$$P(E_1) = P(E_1 \cap E_2 | E_2). \tag{3}$$

We can estimate $P(E_1 \cap E_2|E_2)$ from our data by

$$\widehat{P}(E_1 \cap E_2|E_2) = \frac{|A_1 \cap A_2|}{|A_2|}, \tag{4}$$

where A_1 and A_2 are the sets of pages returned when all the queries utilized in a study were submitted to engines $\mathcal{E}_1$ and $\mathcal{E}_2$. As a result, the size of the Web can be estimated by

$$\widehat{|\Omega|} = \left\lfloor \frac{|E_1| \cdot |A_2|}{|A_1 \cap A_2|} \right\rfloor, \tag{5}$$

where $\lfloor x \rfloor$ is the largest integer smaller than or equal to x.

Formula (5) gives us a way to extrapolate the size of the indexable Web based on the published size of the index of an engine $\mathcal{E}_1$ and on the estimated overlap between $\mathcal{E}_1$ and another engine $\mathcal{E}_2$. But assumptions (A1)–(A2) are not necessarily satisfied. The search engines do not index Web documents at random. They employ two major techniques to detect new pages: user registration and following (hyper)links [9]. On one hand, people who publish on the Web have the tendency to register their pages with as many services as possible. On the other hand, popular pages that have more links to them will have greater chances to be indexed than new (hence unlinked) pages. We infer that search engines will be more inclined to index several well-defined fractions of the indexable Web, which will induce a positive or negative correlation between any two search engines indices. Since the probability of a page being indexed is not constant, a search engine's index will represent a biased sample from the entire population of Web documents.

The estimate in Lawrence and Giles [8] was based on the overlap between AltaVista and HotBot. Since they were the engines with the largest (relative) coverage at the time of the tests (among the six engines studied), their indices will have "lower dependence because they can index more pages other than the pages the users register and they can index more of the less popular pages on the Web" [8]. The reported size of HotBot was 110 million pages, hence Lawrence and Giles found 320 million pages to be an estimate of the size of the indexable Web in December 1997. Bharat and Broder argue that the indexable Web should have about 160 million pages as of November 1997, since AltaVista had reportedly indexed 100 million pages at that time and "had indexed an estimated 62% of the combined set of URLs" [2]. There is a clear discrepancy between the two estimates. Since the queries used by Lawrence and Giles were issued by researchers, they relate to topics few users search for. Search engines are oriented toward finding information the average user wants, thus Lawrence

and Giles might have underestimated the overlap between indices. On the other hand, Bharat and Broder might have overestimated the overlap since the engines have a tendency to locate content-rich documents and these are the documents the randomly generated queries are inclined to match. As a consequence, it appears that Lawrence and Giles overestimated the size of the indexable Web, whereas Bharat and Broder underestimated it.

Although the Web is a dynamic environment, it can be assumed that the population of Web documents is *closed* at a fixed time t_0, i.e. "there are no changes in the size of the population due to birth, death, emigration or immigration from one sample to the next" [5]. This definition translates in our framework to: *no Web pages were added, deleted or modified while the data was collected.* Since the entire indexable Web can be considered closed at a fixed time t_0, the subpopulation of pages that would match query k, $1 \leq k \leq 575$, will also be closed at time t_0. Our goal is to assess the size N_k of the population of pages defined by query k at time t_0 (i.e., December 1997) using the standard multiple-recapture approach to population estimation. In the capture–recapture terminology, Web pages are referred to as *individuals* or *objects* and search engines as *lists*.

More precisely, we have six samples L_1^k,..., L_6^k, where L_i^k, $1 \leq i \leq 6$, represents the best matches for query k found by engine i. Following Fienberg [5], let n_{1+}^k and n_{+1}^k be the number of individuals in the samples L_1^k and L_2^k, respectively, and n_{11}^k be the number of individuals in both lists. The classical capture–recapture estimate for N_k based on the first two lists is

$$\hat{N}_k = \left\lfloor \frac{n_{1+}^k \cdot n_{+1}^k}{n_{11}^k} \right\rfloor, \tag{6}$$

i.e. the traditional "Petersen" estimate. We can compute the Petersen estimates for N_k based on all pairs of the six available lists. The Petersen estimate assumes the objects are heterogeneous (A1) and the lists are pairwise independent (A2), hence Eq. (5) and the Petersen estimate (6) are built on the same suppositions. Moreover, we can see that the estimate Lawrence and Giles found for the indexable Web is nothing more than a Petersen estimate scaled up by a factor, namely the number of pages HotBot had reportedly indexed divided by the total number of pages found by HotBot for the 575 queries.

We considered the seven-way table $\mathcal{S}$ collapsed across queries and computed the traditional capture–recapture estimates for the number $\mathcal{N}$ of Web pages matching at least one of the queries used. Only 7 out of 15 were above the observed number of objects in the six lists, which represents a lower bound for the "real" number of pages $\mathcal{N}$. In Fig. 1, we give the proportion $\mathcal{T}_k$ of Petersen estimates smaller than the observed number of pages n_k for every six-way contingency table $\mathcal{S}_k$. Proportion $\mathcal{T}_k$ is bigger than 50% for almost half the queries. This is clear evidence that the assumptions (A1)–(A2) do not hold. These calculations also suggest that there is positive and also negative dependence between pairs of search engines across the 575 queries. Since the six search engines attempt to maintain full-text indices of the entire

indexable Web, the interactions we observed are the result of the bias introduced by the query-based sampling.

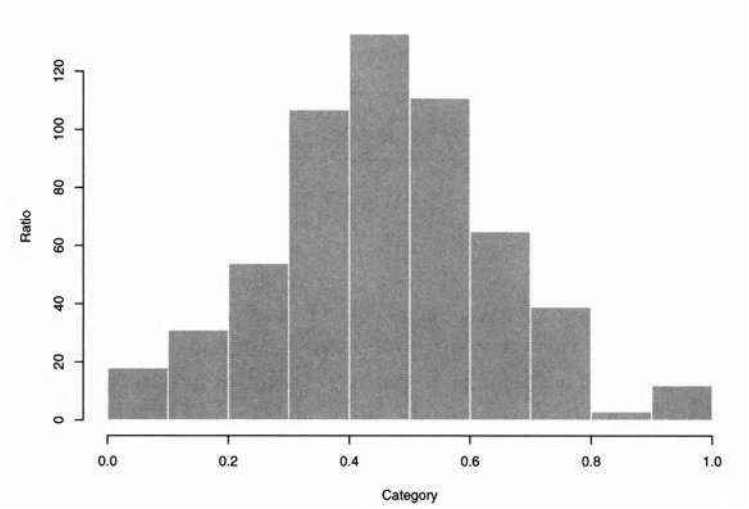

Fig. 1. Proportion of Petersen estimates smaller than n_k

3 Our Approach for Estimating the Size of the Web

The 575 queries of Lawrence and Giles [8] define a population of Web documents, while the union of the indices of the search engines employed define another population of pages. We observed the intersection between the two populations and summarized it in a seven-way contingency table S with missing entries. If we were able to approximate the number of pages *not* found by all the search engines used for every query, we could draw inferences about the dimension of the population of pages relevant to the 575 queries. Based on this estimate and on the published size of the index of one of the six search engines, we could extrapolate the number of Web documents contained in the dotted rectangle in Fig. 2. This approach provides a lower bound of the size of the indexable Web as of December 1997, and in the following sections we propose one possible implementation of it using an approach suggested in Fienberg at al. [6].

3.1 The Rasch Model

The subpopulation of Web documents matching query k, $1 \leq k \leq 575$, is a closed population at a given point in time. Our objective is to estimate its unknown size

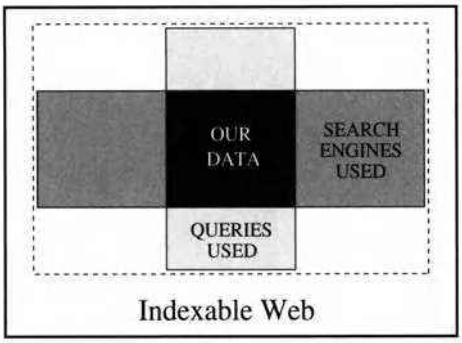

Fig. 2. The two populations of Web pages that define the observed data

N_k using multiple lists or sources. We make use of the kth contingency table $\mathcal{S}_k$, that cross-classifies individuals (Web pages) based on which search engines (or lists) were able to locate them. This is the usual setting for the multiple-recapture population estimation problem, which originated in estimating wildlife and fish populations.

Let $i = 1, \ldots, N_k$ index the individuals and $j = 1, \ldots, J = 6$ index the lists. Define

$$X_{ij} = \begin{cases} 1, & \text{if engine } j \text{ located page } i, \\ 0, & \text{otherwise.} \end{cases}$$

In other words, $X_{ij} = 1$ if individual i appears on list j. Let p_{ij} be the probability of this event. The number of Web pages identified by at least one search engine for query k is n_k. Clearly, estimating N_k is equivalent to estimating $N_k - n_k$. We require a model that allows for

1. **Heterogeneity of capture probabilities**: The probability of a page being indexed by a search engine is not constant. Pages with more links to them are more likely to be located by a search service [9].
2. **List dependencies**: Search engines are more inclined to index certain fractions of the Web, hence the search results they return will be correlated.
3. **Heterogeneity among search services**: Each engine has a specific built-in searching mechanism and because this mechanism is different from one engine to the other, the set of Web documents indexed by every service will also be different.

Rasch [10] introduced a simple mixed-effects generalized linear model that allows for object heterogeneity and list heterogeneity. The multiple-recapture model can be expressed as

$$\log\left\{\frac{p_{ij}}{1-p_{ij}}\right\} = \theta_i + \beta_j; \quad i = 1, \ldots, N_k; \quad j = 1, \ldots, J; \tag{7}$$

where θ_i is the random catchability effect for the ith Web page, and β_j is the fixed effect for the penetration of engine j into the target population represented by all indexable Web documents relevant for the kth query. The heterogeneity of capture probabilities across objects depends on the distribution F_Θ of $\theta = (\theta_1, ..., \theta_{N_k})$. Note that, if we set the θ_i in Eq. (7) equal to zero, the log-odds of inclusion of object i on list j depends only on the list, and thus the Rasch model reduces to the traditional capture-recapture model with independent lists. When the θ_i's are different from zero and we treat them as random effects, this model is multilevel, with lists at one level and individuals at another.

Fienberg et al. [6] showed how to analyze one query using a Bayesian approach for estimating the parameters of the Rasch model. They employed the following full Bayesian specification:

$$\begin{cases} X_{ij} & \sim & Bern(p_j|\theta_i);\ i = 1, ..., N_k;\ j = 1, ..., J; \\ \theta_i & \sim & F_\Theta(\theta_i); \quad i = 1, ..., N_k; \\ \beta_j & \sim & G_\beta(\beta_j); \quad j = 1, ..., J. \end{cases} \tag{8}$$

This permits us to describe all of the model components, and alter precisely those parts that need adjustment to reflect the dependency in the data. We use an extension of the Markov chain Monte Carlo technique for fitting item-response models, as described in Johnson et al. [7]. Following Fienberg et al. [6], we assume that the vector of list parameters $\beta = (\beta_1, ..., \beta_6)$ is distributed $\mathbf{N}_6(\mathbf{0}, 10 \cdot \mathbf{I}_6)$ and is independent of (θ, σ^2, N_k). The catchability parameter vector is distributed $\theta|\sigma^2, N_k \sim \mathbf{N}_{N_k}(\mathbf{0}, \sigma^2 \cdot \mathbf{I})$ and $\sigma^2 \sim \Gamma^{-1}(1, 1)$. This distribution is proper but presumes that we have little knowledge about the search engines indices and about their underlying indexing algorithms. As the prior distribution of N_k, we use a variation of the Jeffrey's prior

$$f_{\mathbf{N}_k}(N_k) \propto \frac{1}{N_k} \cdot I_{\{n_k < N_k < N_k^{\max}\}}. \tag{9}$$

This specification is robust to the choice of $N_k^{\max}$ and can be as small as $10,000$ or as large as $1,500,000$. The latter threshold was used when fitting the Rasch model for the table collapsed across queries.

To illustrate the use of this model, we consider query 140, which has $n_{140} = 159$ URLs, and we compare the results obtained by fitting the Bayesian Rasch model with the classical Petersen estimates. The posterior distribution of N_{140} is skewed, while the median (639) is not very close to the mode (481; Fig. 3). The 95% confidence interval for N_{140} is $[330, 1691]$. The 95% *highest posterior density* (HPD henceforth) interval is $[268, 1450]$. This 95% HPD interval for the Bayesian Rasch model is an equal-tailed probability interval [6]. The lower end of the HPD interval is only slightly smaller than the lower end of the 95% confidence interval, whereas the upper ends of the two intervals are a lot farther apart. We are not surprised by this fact since the posterior distribution has a long right tail. Both the mean (165) and the maximum (322) of the Petersen estimates are bigger than the observed number of pages. The Petersen estimates suggest that the expected number of pages is only twice as large as

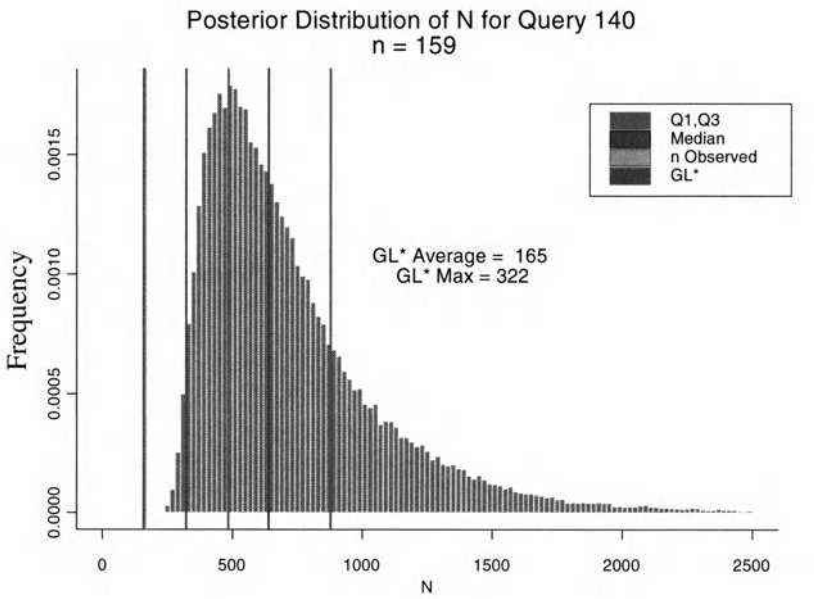

Fig. 3. Posterior distribution of the projected number of Web pages N_k for query $\mathcal{Q}_{140}$

n_{140}, as compared with the Rasch model estimate, which is at least four times larger than n_{140}.

As the observed number of pages n_k increases, the posterior distribution of the projected number of pages N_k moves toward a symmetric distribution. The Petersen estimator constantly underestimates N_k when compared with the inferences we draw through the Rasch model. Since the assumptions the Rasch model is built on are a lot closer to reality than the assumptions (A1)–(A2) we make when using the Petersen estimator, we are inclined to give more credit to the Rasch model. Moreover, the Lawrence and Giles approximation of the size of the indexable Web is of the same order of magnitude as the Petersen estimator.

3.2 Collapsing versus Regression

Estimating the total number of documents $\mathcal{N}$ on the indexable Web relevant to at least one of the 575 queries is a key step in our analysis. Each of the 575 queries defines approximately disjoint segments of the Web. Since the Rasch model provides a good estimate of the size $\hat{N}_k$ of each subpopulation, we would be tempted to approximate $\mathcal{N}$ as

$$\hat{\mathcal{N}} = \sum_{k=1}^{575} \hat{N}_k. \tag{10}$$

Although simple and appealing, it is not easy to make use of Eq. (10) in practice since it requires fitting a Rasch model for every query we work with. An alternative solution is to fit the Rasch model for the contingency table $\mathcal{S}_0$ derived from the seven-way table $\mathcal{S}$ by collapsing across queries. We are aware that the different queries induce heterogeneous populations of pages, hence building our reasoning solely on the six-way cross-classification $\mathcal{S}_0$ might seriously bias our findings. On the other hand, the heterogeneity effect might not be as strong as we expect and so it might be adequate to make use of $\mathcal{S}_0$.

To account for the possible heterogeneity effect, we sampled without replacement 128 queries from the 575 queries (about 20%) contained in the data set, and we used the Rasch model to estimate the number of relevant pages that were not found by any of the search engines. We estimated N_k for the queries not selected in the sample by employing simple linear regression. We modeled the posterior mean, median and mode of N_k as a function of the observed number of pages for every query in the sample and ended up with the following models:

$$\begin{aligned}
&\text{(M1)} \quad \log(\hat{N}_k^{\text{mean}}) &&= 4.67 + 19.2 \cdot \log(n_k), \\
&\text{(M2)} \quad \log(\hat{N}_k^{\text{median}}) &&= 130.6 \cdot \log(n_k), \\
&\text{(M3)} \quad \log(\hat{N}_k^{\text{mode}}) &&= -2.88 + 41.94 \cdot \log(n_k).
\end{aligned} \tag{11}$$

The coefficients of determination for models (M1), (M2) and (M3) are 75%, 99% and 93% respectively. Care should be taken when interpreting the coefficient of determination R^2 for (M2), since the intercept is not present in the model. The plot of observed versus fitted values (Fig. 4) confirms the validity of the models we proposed. It appears that the projected number of pages germane to query $\mathcal{Q}_k$ is directly proportional on a logarithmic scale to n_k, that is, the total number of Web pages identified by the six search engines combined. The models (M2) and (M3), which are the "best" regressions, can be employed to predict N_k for the queries for which we did not fit the Rasch models. The six search engines employed by Lawrence and Giles [8] identified 49,416 pages on the Web relevant to at least one of the 575 queries. The predicted number of relevant pages is 167,298 if we use model (M2) and 125,368 if we use model (M3). Therefore these regression-based projections suggest that there exist *at least* twice as many relevant pages on the Web that were not found by any search engine.

In Fig. 5, we present the posterior distribution of $\mathcal{N}$ from fitting the Bayesian Rasch model for table $\mathcal{S}_0$. This distribution is symmetric and unimodal, with a posterior median equal to $N_0 = 184,160$. The 95% HPD interval for $\mathcal{N}$ is [173427, 199939]. The mean 50,440 of the Petersen estimates is only slightly bigger than the total number of pages $n_0 = 49,416$ captured by the combined search engines for all the queries, whereas the maximum is 75,130. Consequently, the projected number of pages $\widehat{\mathcal{N}}$ using the Petersen estimator is not even twice as large as n_0, while using the Rasch model the same quantity would be approximated to be almost four times as large as n_0.

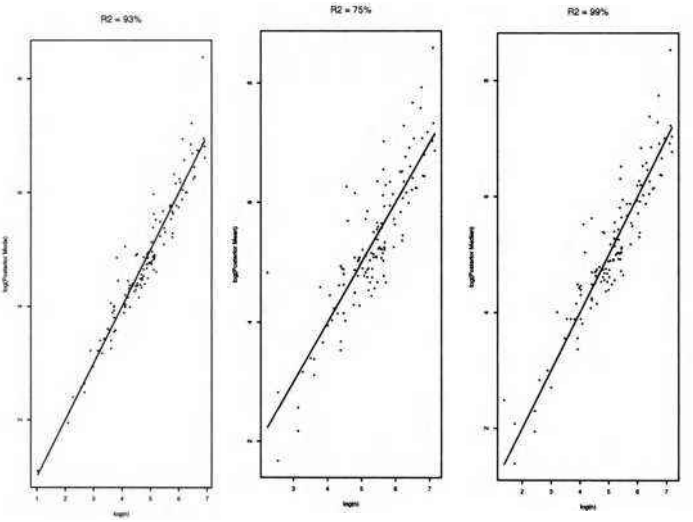

Fig. 4. Regression of the posterior mode, mean and median of N_k on the observed number of pages

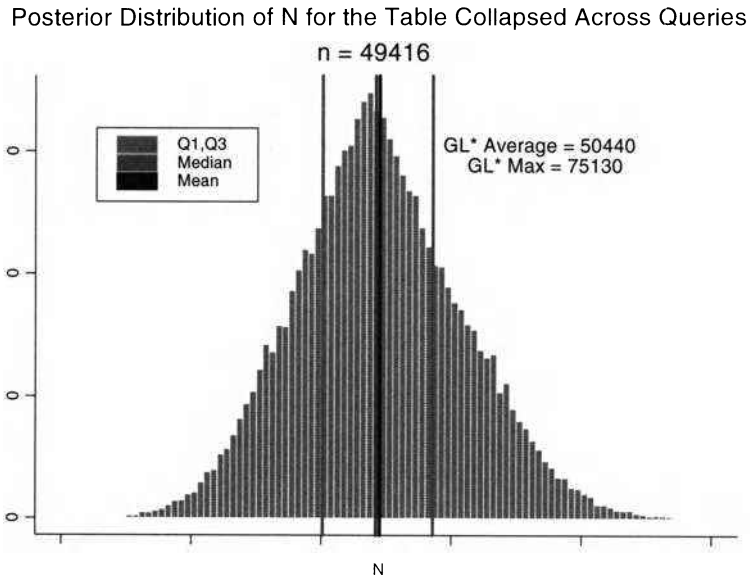

Fig. 5. Posterior distribution of $\mathcal{N}$ for the seven-way table $\mathcal{S}$ aggregated across queries

Recall that β_j, $1 \leq j \leq 6$, is the fixed effect for the penetration of engine j into the target population. Figure 6 portrays the catch effort of AltaVista across all of the 128 sample queries. We plotted several summary statistics based on the posterior distribution of β_1 from the samples we generated using the Rasch model. The overall catch effort β_1^0 of AltaVista is taken to be the posterior median of the Rasch model for table $\mathcal{S}_0$. Figure 6 offers unmistakable evidence that the performance of AltaVista remains stable across the 575 queries used, since β_1^0 stays within the 95% confidence limits for almost all 128 queries. The posterior distributions of those β_1's for which β_1^0 lies outside the 95% confidence intervals might not be well approximated due to insufficient information – few Web pages observed for the corresponding queries. The rest of the search engines exhibit the same unvarying behavior. When interpreting Fig. 6, we have to keep in mind that the 575 queries have no "natural" order; they were labeled with "1", "2", ..., "575" in the same order in which Lawrence and Giles [8] included them in the initial 575×63 matrix they provided us with. This means that the curves in Fig. 6 have no intrinsic meaning. However, Fig. 6 is useful for observing the tightness of the quantiles of the estimated catch efforts of the queries about the overall catch effort: only 3 queries of the 128 deviate significantly from the collapsed value.

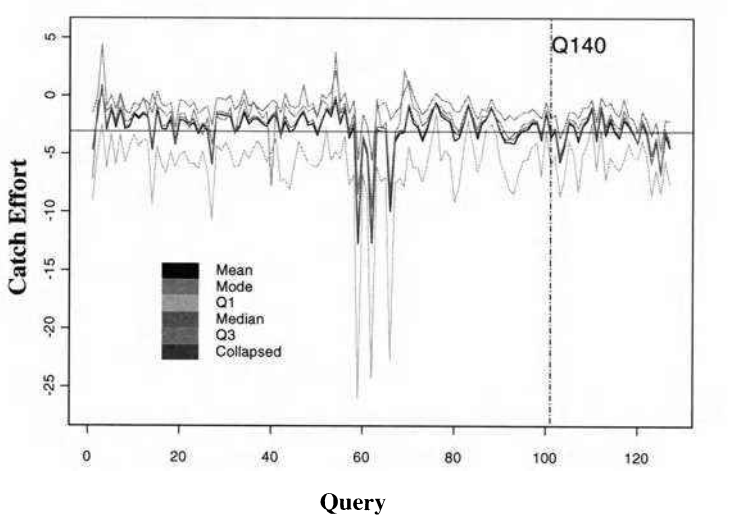

Fig. 6. Catchability effect of AltaVista across the queries selected in the sample. The vertical axis represents the number value of β_1 in model (7) and the curves connect the quantiles of the 128 sample queries displayed along the horizontal axis in an arbitrary order

4 Scaling Up to the Web

We are now in a position to provide estimates for $\mathcal{N}$ based on the analyses described in the preceding section:

- **Method 1**: Select a sample from the set of queries, fit the Rasch model for every sample query and extend the results to the rest of the queries via regression.
- **Method 2**: Find a direct estimate for $\mathcal{N}$ by fitting the Rasch model for the seven-way table $\mathcal{S}$ collapsed across queries.

For the Lawrence and Giles data, method 1 gives $\widehat{\mathcal{N}}_1 = 167,298$ as an estimate for $\mathcal{N}$ if model (M2) is employed, while using method 2 we obtain a slightly larger value, namely $\widehat{\mathcal{N}}_2 = 184,160$. Thus both techniques return results within the same order of magnitude. However, Method 2 fully overlooks the heterogeneity existent among queries and although this method is less expensive to implement, in some particular circumstances we might favor method 1.

Table 3 gives the estimates of the absolute coverage of the six search engines we obtained by employing methods 1 and 2. We contrast our findings with the coverage estimates of Lawrence and Giles [8]. Our estimates suggest that HotBot, the engine with the largest coverage in December 1997, indexed only about 15% of the indexable Web, rather than 34% as calculated by Lawrence and Giles. In addition, our combined coverage of the six search engines is approximately equal to the coverage of AltaVista estimated by Lawrence and Giles!

Table 3. Estimated coverage of the search engines used relative to the indexable Web as of December 1997 (Percentages)

	Estimates based on		Lawrence and Giles [8]
	Method 1	Method 2	estimates
Combined coverage of engines used	29.54	27.00	–
AltaVista	11.00	10.00	28.00
Infoseek	3.91	3.60	10.00
Excite	5.65	5.12	14.00
HotBot	15.37	14.00	34.00
Lycos	1.23	1.11	3.00
Northern Light	7.80	7.00	20.00
Common coverage of engines used	0.06	0.03	–

We cannot make inferences about the size of the indexable Web based on our data alone. Consider a search engine $\mathcal{E}_1$ with index E_1. The relationship in Eq. (2) tells us that the number of documents available on the indexable Web can be estimated by

$$\left\lfloor \frac{|E_1|}{P(E_1)} \right\rfloor . \tag{12}$$

We approximate $P(E_1)$ as the ratio between the total number of pages located by $\mathcal{E}_1$ for all queries used and the estimate for $\mathcal{N}$ we employed. Currently, we have no choice but to rely on the size of the index of $\mathcal{E}_1$ as reported by the engine itself. Since these published estimates are not reliable, we used Eq. (12) for several search engines and compared the results we obtained (Table 4). Lawrence and Giles argued that HotBot had reportedly indexed 110 million pages as of December 1997, and consequently they based their estimates on this value. On the other hand, Bharat and Broder [2] claimed that "Search Engine Watch reported the following search engines sizes (as of November 5, 1997): AltaVista = 100 million pages, HotBot = 80 million, Excite = 55 million, and Infoseek = 30 million pages". The first row uses $\widehat{\mathcal{N}}_1$, while all the other values use $\widehat{\mathcal{N}}_2$ as an estimate of $\mathcal{N}$.

Table 4. Absolute estimates for the size of the Web as of December 1997 (millions of pages)

	Reported Sizes				
	HotBot (80)	HotBot* (110)	Infoseek (30)	AltaVista (100)	Excite (55)
Our Web size (Method 1)	520.63	715.87	767.30	909.29	974.21
Combined coverage of engines used (collapsed)	153.78	211.45	226.46	268.57	287.76
AltaVista (collapsed)	57.26	78.73	84.39	100.00	107.15
Infoseek (collapsed)	20.36	27.99	30.00	35.55	38.09
Excite (collapsed)	29.39	40.42	43.32	51.33	55.00
HotBot (collapsed)	80.00	110.00	117.90	139.71	149.70
HotBot (collapsed)	80.00	110.00	117.90	139.71	149.70
Lycos (collapsed)	6.39	8.78	9.42	11.16	11.96
Northern Light (collpased)	40.58	55.80	59.81	70.88	75.94
Common coverage of engines used (collapsed)	0.16	0.22	0.23	0.28	0.30

Our lowest bound of the size of the indexable Web is 520 million pages, while Lawrence and Giles [8] obtained an estimate of 320 million pages as of December 1997. Remember that Bharat and Broder [2] argued that the Web had only 200 million pages in November 1997. In order to contrast our inferences with the results found by Lawrence and Giles, we scaled up the posterior distribution of $\mathcal{N}$ from fitting the Rasch model for table $\mathcal{S}_0$, using an estimate of the size of HotBot of 110 million pages. This technique allows us to find a distribution of the number of pages available on the indexable Web. The median of this distribution is 788 million pages (see Table 4), while the 95% HPD interval is [742, 856] million pages. If we use the same "external"

information as Lawrence and Giles, we would say that the Web was at least twice as big in 1997 as what was believed until today [2, 8]. In addition, HotBot seems to have the largest index, between 80 and 150 million pages, followed by AltaVista, between 57 and 107 million pages.

We have to emphasize that the method we used for assessing the size of the Web has several shortcomings, and consequently we need to be very careful when interpreting the results obtained by employing it. We pointed out before that the reported sizes of search engines indices are far from being reliable, hence the quantity with which we scale up might not reflect the truth. Furthermore, the "scaling up" itself might not be an adequate solution for our problem. Suppose HotBot has a very good performance in region A, but does very poorly in region B. Moreover, assume that A and B are included in the population of pages relevant to at least one of the 575 queries. According to the method we employed, we would use the same scaling factor for both regions. If these hypotheses were true, we would obviously reach an erroneous conclusion. Nonetheless, we believe that the situation we described is very unlikely to have actually occurred for the six search engines employed in our study.

5 Open Research Questions

1. How to sample from the Web directly, without exploiting the search engines?
2. How to obtain a more reliable estimate of the size of the Web without using a reported size of some search engine index?
3. Are there better ways of expressing/summarizing the amount of information on the Web besides the ones mentioned in this chapter?
4. A new generation of search engines built with different tools, such as Google, has revolutionized Web searches, and the assumptions of the Rasch model are unlikely to hold if we were to look at the engines today. How does that change the way the statistical analyses are performed?

Acknowledgements

Preparation of this chapter was supported in part by the Center for Automated Learning and Discovery at Carnegie Mellon University under Grant no. REC-9720374 from the National Science Foundation. The authors would like to thank Jennifer Pittman for assistance. Lee Giles and Brian Junker contributed with valuable discussions.

References

1. R. Albert, H. Jeong, and A. L. Barabasi. Diameter of the World-Wide Web. *Nature*, 401:130, 1999.
2. K. Bharat and A. Broder. A technique for measuring the relative size and overlap of public web search engines. In *Proceedings of the 7th International World Wide Web Conference*, pages 379–388, Brisbane, Australia, April 1998.

3. E. T. Bradlow and D. C. Schmittlein. The little engines that could: Modelling the performance of world wide web search engines. *Marketing Science*, 19:43–62, 2000.
4. A. Dobra and S. E. Fienberg. How large is the World Wide Web? *Computing Science and Statistics - Proceedings of Interface 2001*, 33, 2001.
5. S. E. Fienberg. The multiple recapture census for closed populations and incomplete 2^k contingency tables. *Biometrika*, 59:591–603, 1972.
6. S.E. Fienberg, M. S. Johnson, and B. W. Junker. Classical multi-level and bayesian approaches to population size estimation using multiple lists. *Journal of Royal Statistical Society*, 162:383–406, 1999.
7. M. S. Johnson, W. Cohen, and B. W. Junker. Measuring appropriability in research and development with item response models. Technical Report, Carnegie Mellon University, 1999.
8. S. Lawrence and C. L. Giles. Searching the world wide web. *Science*, 280:98–100, 1998.
9. S. Lawrence and C. L. Giles. Accessibility of information on the web. *Nature*, 400:107–109, 1999.
10. G. Rasch. *Probabilistic Models for Some Intelligence and Attainment Tests.* Niesen and Lydiche, Copenhagen., 1960. expanded 1980 English edition. University of Chicago Press.
11. E. Selberg. *Towards Comprehensive Web Search.* PhD thesis, University of Washington, June 1999. URL: www.cs.washington.edu/homes/speed/.