# A Sketch Algorithm for Estimating Two-Way and Multi-Way Associations

Ping Li*
Stanford University

Kenneth W. Church**
Microsoft Corporation



*We should not have to look at the entire corpus (e.g., the Web) to know if two (or more) words are strongly associated or not. One can often obtain estimates of associations from a small sample. We develop a sketch-based algorithm that constructs a contingency table for a sample. One can estimate the contingency table for the entire population using straightforward scaling. However, one can do better by taking advantage of the margins (also known as document frequencies). The proposed method cuts the errors roughly in half over Broder's sketches.*

## 1. Introduction

We develop an algorithm for efficiently computing associations, for example, word associations.[1] Word associations (co-occurrences, or joint frequencies) have a wide range of applications including: speech recognition, optical character recognition, and information retrieval (IR) (Salton 1989; Church and Hanks 1991; Dunning 1993; Baeza-Yates and Ribeiro-Neto 1999; Manning and Schutze 1999). The *Know-It-All* project computes such associations at Web scale (Etzioni et al. 2004). It is easy to compute a few association scores for a small corpus, but more challenging to compute lots of scores for lots of data (e.g., the Web), with billions of Web pages ($D$) and millions of word types.

Web search engines produce estimates of page hits, as illustrated in Tables 1–3.[2] Table 1 shows hits for two high frequency words, *a* and *the*, suggesting that the total number of English documents is roughly $D \approx 10^{10}$. In addition to the two high-frequency words, there are three low-frequency words selected from *The New Oxford Dictionary of English* (Pearsall 1998). The low-frequency words demonstrate that there are many hits, even for relatively rare words.

How many page hits do "ordinary" words have? To address this question, we randomly picked 15 pages from a learners' dictionary (Hornby 1989), and selected the first entry on each page. According to Google, there are 10 million pages/word (median value, aggregated over the 15 words). To compute all two-way associations for the 57,100 entries in this dictionary would probably be infeasible, let alone all multi-way associations.

* Department of Statistical Science, Cornell University, Ithaca, NY 14853. E-mail: pl332@cornell.edu.
** Microsoft Research, Microsoft Corp., Redmond, WA 98052. E-mail: church@microsoft.com.

1 This paper considers boolean (0/1) data. See Li, Church, and Hastie (2006, 2007) for generalizations to real-valued data (and $l_p$ distances).

2 All experiments with MSN.com and Google were conducted in August 2005.

Submission received: 6 December 2005; revised submission received: 5 September 2006; accepted for publication: 7 December 2006.

**Table 1**
Page hits for a few high-frequency words and a few low-frequency words (as of August 2005).

| Query | Hits (MSN.com) | Hits (Google) |
|---|---|---|
| A | 2,452,759,266 | 3,160,000,000 |
| The | 2,304,929,841 | 3,360,000,000 |
| Kalevala | 159,937 | 214,000 |
| Griseofulvin | 105,326 | 149,000 |
| Saccade | 38,202 | 147,000 |

**Table 2**
Estimates of page hits are not always consistent. Joint frequencies ought to decrease monotonically as we add terms to the query, but estimates produced by current state-of-the-art search engines sometimes violate this invariant.

| Query | Hits (MSN.com) | Hits (Google) |
|---|---|---|
| America | 150,731,182 | 393,000,000 |
| America, China | 15,240,116 | 66,000,000 |
| America, China, Britain | 235,111 | 6,090,000 |
| America, China, Britain, Japan | 154,444 | 23,300,000 |

**Table 3**
This table illustrates the usefulness of joint counts in query planning for databases. To minimize intermediate writes, the optimal order of joins is: ((“Schwarzenegger” ∩ “Austria”) ∩ “Terminator”) ∩ “Governor,” with 136,000 intermediate results. The standard practice starts with the least frequent terms, namely, ((“Schwarzenegger” ∩ “Terminator”) ∩ “Governor”) ∩ “Austria,” with 579,100 intermediate results.

| | Query | Hits (Google) |
|---|---|---|
| One-way | Austria | 88,200,000 |
| | Governor | 37,300,000 |
| | Schwarzenegger | 4,030,000 |
| | Terminator | 3,480,000 |
| Two-way | Governor, Schwarzenegger | 1,220,000 |
| | Governor, Austria | 708,000 |
| | Schwarzenegger, Terminator | 504,000 |
| | Terminator, Austria | 171,000 |
| | Governor, Terminator | 132,000 |
| | Schwarzenegger, Austria | 120,000 |
| Three-way | Governor, Schwarzenegger, Terminator | 75,100 |
| | Governor, Schwarzenegger, Austria | 46,100 |
| | Schwarzenegger, Terminator, Austria | 16,000 |
| | Governor, Terminator, Austria | 11,500 |
| Four-way | Governor, Schwarzenegger, Terminator, Austria | 6,930 |

Estimates are often good enough. We should not have to look at every document to determine whether two words are strongly associated or not. One could use the estimated co-occurrences from a small sample to compute the test statistics, most commonly Pearson’s chi-squared test, the likelihood ratio test, Fisher’s exact test, cosine similarity, or resemblance (Jaccard coefficient) (Dunning 1993; Manning and Schutze 1999; Agresti 2002; Moore 2004).

Sampling can make it possible to work in physical memory, avoiding disk accesses. Brin and Page (1998) reported an inverted index of 37.2 GBs for 24 million pages. By extrapolation, we should expect the size of the inverted indexes for current Web scale to be 1.5 TBs/billion pages, probably too large for physical memory. A sample is more manageable.

When estimating associations, it is desirable that the estimates be consistent. Joint frequencies ought to decrease monotonically as we add terms to the query. Table 2 shows that estimates produced by current search engines are not always consistent.

### 1.1 The Data Matrix, Postings, and Contingency Tables

We assume a *term-by-document matrix*, $\mathbf{A}$, with $n$ rows (words) and $D$ columns (documents). Because we consider boolean (0/1) data, the $(i,j)^{th}$ entry of $\mathbf{A}$ is 1 if word $i$ occurs in document $j$ and 0 otherwise. Computing all pair-wise associations of $\mathbf{A}$ is a matrix multiplication, $\mathbf{A}\mathbf{A}^T$.

Because word distributions have long tails, the term-by-document matrix is highly sparse. It is common practice to avoid materializing the zeros in $\mathbf{A}$, by storing the matrix in adjacency format, also known as *postings*, and an inverted index (Witten, Moffat, and Bell 1999, Section 3.2). For each word W, the postings list, P, contains a sorted list of document IDs, one for each document containing W.

Figure 1(a) shows a contingency table. The contingency table for words $W_1$ and $W_2$ can be expressed as intersections (and complements) of their postings $P_1$ and $P_2$ in the obvious way:

$$a = |P_1 \cap P_2|, \quad b = |P_1 \cap \neg P_2|, \quad c = |\neg P_1 \cap P_2|, \quad d = |\neg P_1 \cap \neg P_2| \tag{1}$$

where $\neg P_1$ is short-hand for $\Omega - P_1$, and $\Omega = \{1, 2, 3, \ldots, D\}$ is the set of all document IDs. As shown in Figure 1(a), we denote the margins by $f_1 = a + b = |P_1|$ and $f_2 = a + c = |P_2|$.

For larger corpora, it is natural to introduce sampling. For example, we can randomly sample $D_s$ (out of $D$) documents, as illustrated in Figure 1(b). This sampling scheme, which we call *sampling over documents*, is simple and easy to describe—but we can do better, as we will see in the next subsection.

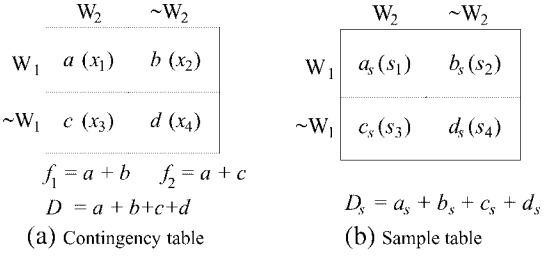



**Figure 1**
(a) A contingency table for word $W_1$ and word $W_2$. Cell $a$ is the number of documents that contain both $W_1$ and $W_2$, $b$ is the number that contain $W_1$ but not $W_2$, $c$ is the number that contain $W_2$ but not $W_1$, and $d$ is the number that contain neither. The margins, $f_1 = a + b$ and $f_2 = a + c$ are known as document frequencies in IR. $D = a + b + c + d$ is the total number of documents in the collection. For consistency with the notation we use for multi-way associations, $a$, $b$, $c$, and $d$ are also denoted, in parentheses, by $x_1$, $x_2$, $x_3$, and $x_4$, respectively. (b) A sample contingency table ($a_s$, $b_s$, $c_s$, $d_s$), where the subscript $s$ indicates the *sample space*. The cells are also numbered as ($s_1$, $s_2$, $s_3$, $s_4$).

### 1.2 Sampling Over Documents and Sampling Over Postings

Sampling over documents selects $D_s$ documents randomly from a collection of $D$ documents, as illustrated in Figure 1.

The task of computing associations is broken down into three subtasks:

1. Compute sample contingency table.
2. Estimate contingency table for population from sample.
3. Summarize contingency table to produce desired measure of association: cosine, resemblance, mutual information, correlation, and so on.

Sampling over documents is simple and well understood. The estimation task is straightforward if we ignore the margins. That is, we simply scale up the sample in the obvious way: $\hat{a}_{MF} = a_s \frac{D}{D_s}$. We refer to these estimates as the "margin-free" baseline. However, we can do better when we know the margins, $f_1 = a + b$ and $f_2 = a + c$ (called **document frequencies** in IR), using a maximum likelihood estimator (MLE) with fixed margin constraints.

Rare words can be a challenge for sampling over documents. In terms of the term-by-document matrix **A**, sampling over documents randomly picks a fraction ($\frac{D_s}{D}$) of columns from **A**. This is a serious drawback because **A** is highly sparse (as word distributions have long tails) with a few high-frequency words and many low-frequency words. The jointly non-zero entries in **A** are unlikely to be sampled unless the sampling rate $\frac{D_s}{D}$ is high. Moreover, the word sparsity differs drastically from one word to another; it is thus desirable to have a sampling mechanism that can adapt to the data sparsity with flexible sample sizes. One size does not fit all.

"Sampling over postings" is an interesting alternative to sampling over documents. Unfortunately, it doesn't work out all that well either (at least using a simple straightforward implementation), but we present it here nevertheless, because it provides a convenient segue between sampling over documents and our sketch-based recommendation.

"Naive sampling over postings" obtains a random sample of size $k_1$ from $P_1$, denoted as $Z_1$, and a random sample $Z_2$ of size $k_2$ from $P_2$. Also, we denote $a_s^N = |Z_1 \cap Z_2|$. We then use $a_s^N$ to infer $a$. For simplicity, assume $k_1 = k_2 = k$ and $f_1 = f_2 = f$. It follows that[3] $\mathrm{E}\left(\frac{a_s^N}{a}\right) = \frac{k^2}{f^2}$. In other words, under naive sampling over postings, one could estimate the associations by $\frac{f^2}{k^2} a_s^N$.

3 Suppose there are $m$ defectives among $N$ objects. We randomly pick $k$ objects (without replacement) and obtain $x$ defectives. Then $x$ follows a hypergeometric distribution, $x \sim HG(N, m, k)$. It is known that $\mathrm{E}(x) = \frac{m}{N}k$. In our setting, suppose we know that among $Z_1$ (of size $k_1$), there are $a_s^{Z_1}$ samples that belong to the original intersection $P_1 \cap P_2$. Similarly, suppose we know that there are $a_s^{Z_2}$ samples among $Z_2$ (of size $k_2$) that belong to $P_1 \cap P_2$. Then $a_s^N = |Z_1 \cap Z_2| \sim HG(a, a_s^{Z_1}, a_s^{Z_2})$. Therefore $\mathrm{E}\left(a_s^N\right) = \frac{1}{a} a_s^{Z_1} a_s^{Z_2}$. Because $a_s^{Z_1}$ and $a_s^{Z_2}$ are both random, we should use conditional expectations: $\mathrm{E}\left(a_s^N\right) = \mathrm{E}\left(\mathrm{E}\left(a_s^N | a_s^{Z_1}, a_s^{Z_2}\right)\right) = \mathrm{E}\left(\frac{1}{a} a_s^{Z_1} a_s^{Z_2}\right) = \frac{1}{a}\mathrm{E}\left(a_s^{Z_1}\right)\mathrm{E}\left(a_s^{Z_2}\right)$. (Recall that $Z_1$ and $Z_2$ are independent.) Note that $a_s^{Z_1} \sim HG(f_1, a, k_1)$ and $a_s^{Z_2} \sim HG(f_2, a, k_2)$, that is, $\mathrm{E}\left(a_s^{Z_1}\right) = \frac{a}{f_1}k_1$ and $\mathrm{E}\left(a_s^{Z_2}\right) = \frac{a}{f_2}k_2$. Therefore, $\mathrm{E}\left(a_s^N\right) = \frac{1}{a}\frac{a}{f_1}k_1\frac{a}{f_2}k_2$, namely, $\mathrm{E}\left(\frac{a_s^N}{a}\right) = \frac{k_1 k_2}{f_1 f_2}$.

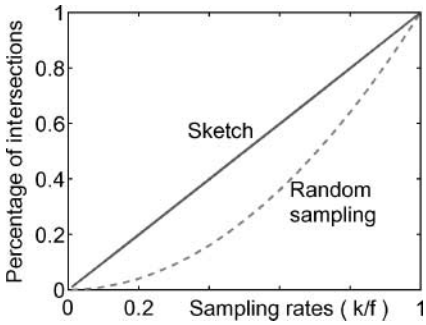



**Figure 2**
The proposed sketch method (solid curve) produces larger counts ($a_s$) with less work ($k$). With "naive sampling over postings," there is an undesirable quadratic: $\mathrm{E}\left(\frac{a_s^{\mathrm{N}}}{a}\right) = \frac{k^2}{f^2}$ (dashed curve), whereas with sketches, $\mathrm{E}\left(\frac{a_s}{a}\right) \approx \frac{k}{f}$. These results were generated by simulation, with $f_1 = f_2 = f = 0.2D$, $D = 10^5$ and $a$ = 0.22, 0.38, 0.65, 0.80, 0.85$f$. There is only one dashed curve across all values of $a$. There are different (but indistinguishable) solid curves depending on $a$.

Of course, the quadratic relation, $\mathrm{E}\left(\frac{a_s^{\mathrm{N}}}{a}\right) = \frac{k^2}{f^2}$, is undesirable; 1% effort returns only 0.01% useful information. Ideally, to maximize the signal, we'd like to see large counts in a small sample, not small counts in a large sample. The crux is $a_s$, which tends to have the smallest counts. We'd like $a_s$ to be as large as possible, but we'd also like to do as little work ($k$) as possible. The next subsection on sketches proposes an improvement, where 1% effort returns roughly 1% useful information, as illustrated in Figure 2.

### 1.3 An Improvement Based on Sketches

A sketch is simply the front of the postings (after a random permutation). We find it helpful, as an informal practical metaphor, to imagine a virtual machine architecture where sketches (Broder 1997), the front of the postings, reside in physical memory, and the rest of the postings are stored on disk. More formally, the sketch, $\mathrm{K} = \mathrm{MIN}_k(\pi(\mathrm{P}))$, contains the $k$ smallest postings, after applying a random permutation $\pi$ to document IDs, $\Omega = \{1, 2, 3, \ldots, D\}$, to eliminate whatever structure there might be.

Given two words, $\mathrm{W}_1$ and $\mathrm{W}_2$, we have two sets of postings, $\mathrm{P}_1$ and $\mathrm{P}_2$, and two sketches, $\mathrm{K}_1 = \mathrm{MIN}_{k_1}(\pi(\mathrm{P}_1))$ and $\mathrm{K}_2 = \mathrm{MIN}_{k_2}(\pi(\mathrm{P}_2))$. We construct a sample contingency table from the two sketches. Let $\Omega_s = \{1, 2, 3, \ldots, D_s\}$ be the sample space, where $D_s$ is set to $\min(\max(\mathrm{K}_1), \max(\mathrm{K}_2))$. With this choice of $D_s$, all the document IDs in the sample space, $\Omega_s$, can be assigned to the appropriate cell in the sample contingency table without looking outside the sketch. One could use a smaller $D_s$, but doing so would throw out data points unnecessarily.

The sample contingency table is constructed from $\mathrm{K}_1$ and $\mathrm{K}_2$ in $O(k_1 + k_2)$ time, using a straightforward linear pass over the two sketches:

$$\begin{aligned} a_s &= |\mathrm{K}_1 \cap \mathrm{K}_2 \cap \Omega_s| = |\mathrm{K}_1 \cap \mathrm{K}_2| \quad & b_s &= |\mathrm{K}_1 \cap \neg\mathrm{K}_2 \cap \Omega_s| \\ c_s &= |\neg\mathrm{K}_1 \cap \mathrm{K}_2 \cap \Omega_s| & d_s &= |\neg\mathrm{K}_1 \cap \neg\mathrm{K}_2 \cap \Omega_s| \end{aligned} \tag{2}$$

The final step is an estimation task. The margin-free (MF) estimator recovers the original contingency table by a simple scaling. For better accuracy, one could take advantage of the margins by using a maximum likelihood estimator (MLE).

With "sampling over documents," it is convenient to express the sampling rate in terms of $D_s$ and $D$, whereas with sketches, it is convenient to express the sampling rate in terms of $k$ and $f$. The following two approximations allow us to flip back and forth between the two views:

$$\mathrm{E}\left(\frac{D_s}{D}\right) \approx \min\left(\frac{k_1}{f_1}, \frac{k_2}{f_2}\right) \tag{3}$$

$$\mathrm{E}\left(\frac{D}{D_s}\right) \approx \max\left(\frac{f_1}{k_1}, \frac{f_2}{k_2}\right) \tag{4}$$

In other words, using sketches with size $k$, the corresponding sample size $D_s$ in "sampling over documents" would be $D_s \approx \frac{D}{f}k$, where $\frac{D}{f}$ represents the data sparsity. Because the estimation errors (variances) are inversely proportional to sample size, we know the proposed algorithm improves "sampling over documents" by a factor proportional to the data sparsity.

### 1.4 Improving Estimates Using Margins

When we know the margins, we ought to use them. The basic idea is to maximize the likelihood of the sample contingency table under margin constraints. In the pair-wise case, we will show that the resultant maximum likelihood estimator is the solution to a cubic equation, which has a remarkably accurate quadratic approximation.

The use of margins for estimating contingency tables was suggested in the 1940s (Deming and Stephan 1940; Stephan 1942) for a census application. They developed a straightforward iterative estimation method called **iterative proportional scaling**, which was an approximation to the maximum likelihood estimator.

Computing margins is usually much easier than computing interactions. For a data matrix $\mathbf{A}$ of $n$ rows and $D$ columns, computing all marginal $l_2$ norms costs only $O(nD)$, whereas computing all pair-wise associations (or $l_2$ distances) costs $O(n^2D)$. One could compute the margins in a separate prepass over the data, without increasing the time and space complexity, though we suggest computing the margins while applying the random permutation $\pi$ to all the document IDs on all the postings.

### 1.5 An Example

Let's start with conventional random sampling over documents, using a running example in Figure 3. We choose a sample of $D_s = 18$ documents randomly out of a collection of $D = 36$. After applying the random permutation, document IDs will be uniformly random. Thus, we can construct the random sample by picking any $D_s$ documents. For convenience, we pick the first $D_s$. The sample contingency table is then constructed, as illustrated in Figure 3.

The recommended procedure is illustrated in Figure 4. The two sketches, $\mathrm{K}_1$ and $\mathrm{K}_2$, are highlighted in the large box. We find it convenient, as an informal practical metaphor, to think of the large box as physical memory. Thus, the sketches reside in physical memory, and the rest are paged out to disk. We choose $D_s$ to be $\min(\max(\mathrm{K}_1), \max(\mathrm{K}_2)) = \min(18, 21) = 18$, so that we can compute the sample contin-

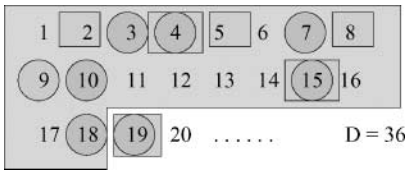



**Figure 3**
In this example, the corpus contains $D = 36$ documents. The population is: $\Omega = \{1, 2, \ldots, D\}$. The sample space is $\Omega_s = \{1, 2, \ldots, D_s\}$, where $D_s = 18$. Circles denote documents containing $W_1$, and squares denote documents containing $W_2$. The sample contingency table is: $a_s = |\{4, 15\}| = 2$, $b_s = |\{3, 7, 9, 10, 18\}| = 5$, $c_s = |\{2, 5, 8\}| = 3$, $d_s = |\{1, 6, 11, 12, 13, 14, 16, 17\}| = 8$.

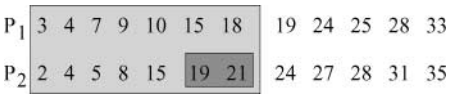



**Figure 4**
This procedure, which we recommend, produces the same sample contingency table as in Figure 3: $a_s = 2$, $b_s = 5$, $c_s = 3$, and $d_s = 8$. The two sketches, $K_1$ and $K_2$ (larger shaded box), reside in physical memory, and the rest of the postings are paged out to disk. $K_1$ contains of the first $k_1 = 7$ document IDs in $P_1$ and $K_2$ contains of the first $k_2 = 7$ IDs in $P_2$. We assume $P_1$ and $P_2$ are already permuted, otherwise we should write $\pi(P_1)$ and $\pi(P_2)$ instead. $D_s = \min(\max(K_1), \max(K_2)) = \min(18, 21) = 18$. The sample contingency table is computed from the sketches (large box) in time $k_1 + k_2$, but documents exceeding $D_s$ are excluded from $\Omega_s$ (small box), because we can't tell if they are in the intersection or not, without looking outside the sketch. As it turns out, 19 is in the intersection and 21 is not.

gency table for $\Omega_s = \{1, 2, 3, \ldots, D_s\}$ in physical memory in time $O(k_1 + k_2)$ from $K_1$ and $K_2$. In this example, documents 19 and 21 (highlighted in the smaller box) are excluded from $\Omega_s$. It turns out that 19 is part of the intersection, and 21 is not, but we would have to look outside the sketches (and suffer a page fault) to determine that. The resulting sample contingency table is the same as in Figure 3:

$$a_s = |\{4, 15\}| = 2 \qquad b_s = |K_1 \cap \Omega_s| - a_s = 7 - 2 = 5$$
$$c_s = |K_2 \cap \Omega_s| - a_s = 5 - 2 = 3 \qquad d_s = D_s - (a_s + b_s + c_s) = 8$$

### 1.6 A Five-Word Example

Figure 5 shows an example with more than two words. There are $D = 15$ documents in the collection. We generate a random permutation $\pi$ as shown in Figure 5(b). For every ID in postings $P_i$ in Figure 5(a), we apply the random permutation $\pi$, but we only store the $k_i$ smallest IDs as a sketch $K_i$, that is, $K_i = \text{MIN}_{k_i}(\pi(P_i))$. In this example, we choose $k_1 = 4$, $k_2 = 4$, $k_3 = 4$, $k_4 = 3$, $k_5 = 6$. The sketches are stored in Figure 5(c). In addition, because $\pi(P_i)$ operates on every ID in $P_i$, we know the total number of non-zeros in $P_i$, denoted by $f_i = |P_i|$.

The estimation procedure is straightforward if we ignore the margins. For example, suppose we need to estimate the number of documents containing the first two words. In other words, we need to estimate the inner product between $P_1$ and $P_2$, denoted by $a_{(1,2)}$. (We have to use the additional subscript $_{(1,2)}$ because we have more than

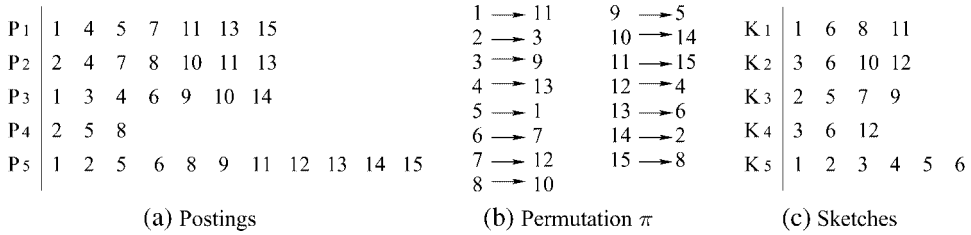



**Figure 5**
The original postings sets are given in (a). There are $D = 15$ documents in the collection. We generate a random permutation $\pi$ as shown in (b). We apply $\pi$ to the postings $P_i$ and store the sketch $K_i = MIN_{k_i}(\pi(P_i))$. For example, $\pi(P_1) = \{11, 13, 1, 12, 15, 6, 8\}$. We choose $k_1 = 4$; and hence the four smallest IDs in $\pi(P_1)$ are $K_1 = \{1, 6, 8, 11\}$. We choose $k_2 = 4$, $k_3 = 4$, $k_4 = 3$, and $k_5 = 6$.



just two words in the vocabulary.) We calculate, from sketches $K_1$ and $K_2$, the sample inner product $a_{s,(1,2)} = |\{6\}| = 1$, and the corresponding corpus sample size, denoted by $D_{s,(1,2)} = \min(\max(K_1), \max(K_2)) = \min(11, 12) = 11$. Therefore, the "margin-free" estimate of $a_{(1,2)}$ is simply $a_{s,(1,2)} \frac{D}{D_{s,(1,2)}} = 1\frac{15}{11} = 1.4$.

This estimate can be compared to the "truth," which is obtained from the complete postings list, as opposed to the sketch. In this case, $P_1$ and $P_2$ have 4 documents in common. And therefore, the estimation error is $4 - 1.4$ or 2.6 documents.

Similarly, for $P_1$ and $P_5$, $D_{s,(1,5)} = \min(11, 6) = 6$, $a_{s,(1,5)} = 2$. Hence, the "margin-free" estimate of $a_{(1,5)}$ is simply $2\frac{15}{6} = 5.0$. In this case, the estimate matches the "truth" perfectly.

The procedure can be easily extended to more than two rows. Suppose we would like to estimate the three-way inner product (three-way joins) among $P_1$, $P_4$, and $P_5$, denoted by $a_{(1,4,5)}$. We calculate the three-way sample inner product from $K_1$, $K_4$, and $K_5$, $a_{s,(1,4,5)} = |\{6\}| = 1$, and the corpus sample size $D_{s,(1,4,5)} = \min(\max(K_1), \max(K_4), \max(K_5)) = \min(11, 12, 6) = 6$. Then the "margin-free" estimate of $a_{(1,4,5)}$ is $1\frac{15}{6} = 2.5$.

Of course, we can improve these estimates by taking advantage of the margins.

## 2. Applications

There is a large literature on sketching techniques (e.g., Alon, Matias, and Szegedy 1996; Broder 1997; Vempala 2004). Such techniques have applications in information retrieval, databases, and data mining (Broder et al. 1997; Haveliwala, Gionis, and Indyk 2000; Haveliwala et al. 2002).

Broder's sketches (Broder 1997) were originally introduced to detect duplicate documents in Web crawls. Many URLs point to the same (or nearly the same) HTML blobs. Approximate answers are often good enough. We don't need to find all such pairs, but it is handy to find many of them, without spending more than it is worth on computational resources.

In IR applications, physical memory is often a bottleneck, because the Web collection is too large for memory, but we want to minimize seeking data in the disk as the query response time is critical (Brin and Page 1998). As a space saving device, dimension reduction techniques use a compact representation to produce approximate answers in physical memory.

Section 1 mentioned page hit estimation. If we have a two-word query, we'd like to know how many pages mention both words. We assume that pre-computing and storing page hits is infeasible, at least not for infrequent pairs of words (and multi-word sequences).

It is customary in information retrieval to start with a large boolean term-by-document matrix. The boolean values indicate the presence or absence of a term in a document. We assume that these matrices are too large to store in physical memory. Depending on the specific applications, we can construct an inverted index and store sketches either for terms (to estimate word association) or for documents (to estimate document similarity).

### 2.1 Association Rule Mining

"Market-basket" analysis and association rules (Agrawal, Imielinski, and Swami 1993; Agrawal and Srikant 1994; Agrawal et al. 1996; Hastie, Tibshirani, and Friedman 2001, Chapter 14.2) are useful tools for mining commercial databases. Commercial databases tend to be large and sparse (Aggarwal and Wolf 1999; Strehl and Ghosh 2000). Various sampling algorithms have been proposed (Toivonen 1996; Chen, Haas, and Scheuermann 2002). The proposed algorithm scales better than traditional random sampling (i.e., a fixed sample of columns of the data matrix) for reasons mentioned earlier. In addition, the proposed algorithm makes it possible to estimate association rules on-line, which may have some advantage in certain applications (Hidber 1999).

### 2.2 All Pair-Wise Associations (Distances)

In many applications, including distance-based classification or clustering and bi-gram language modeling (Church and Hanks 1991), we need to compute all pair-wise associations (or distances). Given a data matrix $\mathbf{A}$ of $n$ rows and $D$ columns, brute force computation of $\mathbf{A}\mathbf{A}^T$ would cost $O(n^2D)$, or more efficiently, $O(n^2\bar{f})$, where $\bar{f}$ is the average number of non-zeros among all rows of $\mathbf{A}$. Brute force could be very time-consuming. In addition, when the data matrix is too large to fit in the physical memory, the computation may become especially inefficient.

Using our proposed algorithm, the cost of computing $\mathbf{A}\mathbf{A}^T$ can be reduced to $O(n\bar{f}) + O(n^2\bar{k})$, where $\bar{k}$ is the average sketch size. It costs $O(n\bar{f})$ for constructing sketches and $O(n^2\bar{k})$ for computing all pair-wise associations. The savings would be significant when $\bar{k} \ll \bar{f}$. Note that $\mathbf{A}\mathbf{A}^T$ is called "Gram Matrix" in machine learning; and various algorithms have been proposed for speeding up the computation (e.g., Drineas and Mahoney 2005).

Ravichandran, Pantel, and Hovy (2005) computed pair-wise word associations (boolean data) among $n \approx 0.6$ million nouns in $D \approx 70$ million Web pages, using random projections. We have discovered that in boolean data, our method exhibits (much) smaller errors (variances); but we will present the detail in other papers (Li, Church, and Hastie 2006, 2007).

For applications which are mostly interested in finding the strongly associated pairs, the $n^2$ might appear to be a show stopper. But actually, in a practical application, we implemented an inverted index on top of the sketches, which made it possible to find many of the most interesting associations quickly.

### 2.3 Database Query Optimization

In databases, an important task is to determine the order of joins, which has a large impact on the system performance (Garcia-Molina, Ullman, and Widom 2002, Chapter 16). Based on the estimates of two-way, three-way, and even higher-order join sizes, query optimizers construct a plan to minimize a cost function (e.g., intermediate writes). Efficiency is critical as we certainly do not want to spend more time optimizing the plan than executing it.

We use an example (called *Governator*) to illustrate that estimates of two-way and multi-way association can help the query optimizer.

Table 3 shows estimates of hits for four words and their two-way, three-way, and four-way combinations. Suppose the optimizer wants to construct a plan for the query: "Governor, Schwarzenegger, Terminator, Austria." The standard solution starts with the least frequent terms: (("Schwarzenegger" ∩ "Terminator") ∩ "Governor") ∩ "Austria." That plan generates 579,100 intermediate writes after the first and second joins. An improvement would be (("Schwarzenegger" ∩ "Austria") ∩ "Terminator") ∩ "Governor," reducing the 579,100 down to 136,000.

## 3. Outline of Two-Way Association Results

To approximate the associations between words $W_1$ and $W_2$, we work with sketches $K_1$ and $K_2$. We first determine $D_s = \min(\max(K_1), \max(K_2))$ and then construct the sample contingency table on $\Omega_s = \{1, 2, \ldots, D_s\}$. The contingency table for the entire document collection, $\Omega = \{1, 2, \ldots, D\}$, is estimated using a maximum likelihood estimator (MLE):

$$\hat{a}_{MLE} = \underset{a}{\operatorname{argmax}} \, \mathbf{Pr}\left(a_s, b_s, c_s, d_s | D_s; a\right) \tag{5}$$

Section 5 will show that $\hat{a}_{MLE}$ is the solution to a cubic equation:

$$\frac{f_1 - a + 1 - b_s}{f_1 - a + 1} \frac{f_2 - a + 1 - c_s}{f_2 - a + 1} \frac{D - f_1 - f_2 + a}{D - f_1 - f_2 + a - d_s} \frac{a}{a - a_s} = 1 \tag{6}$$

Instead of solving a cubic equation, we recommend a convenient and accurate quadratic approximation:

$$\hat{a}_{MLE,a} = \frac{f_1\left(2a_s + c_s\right) + f_2\left(2a_s + b_s\right) - \sqrt{\left(f_1\left(2a_s + c_s\right) - f_2\left(2a_s + b_s\right)\right)^2 + 4 f_1 f_2 b_s c_s}}{2\left(2a_s + b_s + c_s\right)} \tag{7}$$

We will compare the proposed MLE to two baselines: the independence baseline, $\hat{a}_{IND}$, and the margin-free baseline, $\hat{a}_{MF}$:

$$\hat{a}_{IND} = \frac{f_1 f_2}{D} \qquad \hat{a}_{MF} = a_s \frac{D}{D_s} \tag{8}$$

The margin-free baseline has smaller errors than the independence baseline, but we can do even better if we know the margins, as is common in practice.

As expected, computational work and statistical accuracy (variance or errors) depend on sampling rate. The larger the sample, the better the estimate, but the more work we have to do.

These results are demonstrated both empirically and theoretically. In our field, it is customary to end with a large empirical evaluation. But there are always lingering questions. Do the results generalize to other collections with more documents or different documents? This paper attempts to put such questions to rest by deriving closed-form expressions for the variances.

$$\mathrm{Var}\left(\hat{a}_{MLE}\right) \approx \frac{\mathrm{E}\left(\frac{D}{D_s}\right) - 1}{\frac{1}{a} + \frac{1}{f_1 - a} + \frac{1}{f_2 - a} + \frac{1}{D - f_1 - f_2 + a}}, \tag{9}$$

$$\approx \frac{\max\left(\frac{f_1}{k_1}, \frac{f_2}{k_2}\right) - 1}{\frac{1}{a} + \frac{1}{f_1 - a} + \frac{1}{f_2 - a} + \frac{1}{D - f_1 - f_2 + a}}. \tag{10}$$

$$\mathrm{Var}\left(\hat{a}_{MF}\right) = \frac{\mathrm{E}\left(\frac{D}{D_s}\right) - 1}{\frac{1}{a} + \frac{1}{D - a}} \approx \frac{\max\left(\frac{f_1}{k_1}, \frac{f_2}{k_2}\right) - 1}{\frac{1}{a} + \frac{1}{D - a}}. \tag{11}$$

These formulas establish the superiority of the proposed method over the alternatives, not just for a particular data set, but more generally. These formulas will also be used to determine stopping rules. How many samples do we need? We will use such an argument to suggest that a sampling rate of $10^{-3}$ may be sufficient for certain Web applications.

The proposed method generalizes naturally to multi-way associations, as presented in Section 6. Section 7 describes Broder's sketches, which were designed for estimating resemblance, a particular association statistic. It will be shown, both theoretically and empirically, that our proposed method reduces the mean square error (MSE) by about 50%. In other words, the proposed method achieves the same accuracy with about half the sample size (work).

## 4. Evaluation of Two-Way Associations

We evaluated our two-way association sampling/estimation algorithm with a chunk of Web crawls ($D = 2^{16}$) produced by the crawler for MSN.com. We collected two sets of English words which we will refer to as the small data set and the large data set. The small data set contains just four high frequency words: *THIS*, *HAVE*, *HELP* and *PROGRAM* (see Table 4), whereas the large data set contains 968 words (i.e., 468,028 pairs). The large data set was constructed by taking a random sample of English words that appeared in at least 20 documents in the collection. The histograms of the margins and co-occurrences have long tails, as expected (see Figure 6).

For the small data set, we applied $10^5$ independent random permutations to the $D = 2^{16}$ document IDs, $\Omega = \{1, 2, \ldots, D\}$. High-frequency words were selected so we could study a large range of sampling rates ($\frac{k}{f}$), from 0.002 to 0.95. A pair of sketches was constructed for each of the 6 pairs of words in Table 4, each of the $10^5$ permutations and each sampling rate. The sketches were then used to compute a sample contingency table, leading to an estimate of co-occurrence, $\hat{a}$. An error was computed by comparing this estimate, $\hat{a}$, to the appropriate gold standard value for $a$ in Table 4. Mean square errors (MSE = $\mathrm{E}(\hat{a} - a)^2$) and other statistics were computed by aggregating over the $10^5$

**Table 4**
Small dataset: co-occurrences and margins for the population. The task is to estimate these values, which will be referred to as the gold standard, from a sample.

| Case # | Words | Co-occurrence ($a$) | Margin ($f_1$) | Margin ($f_2$) |
|---|---|---|---|---|
| Case 2-1 | THIS, HAVE | 13,517 | 27,633 | 17,369 |
| Case 2-2 | THIS, HELP | 7,221 | 27,633 | 10,791 |
| Case 2-3 | THIS, PROGRAM | 3,682 | 27,633 | 5,327 |
| Case 2-4 | HAVE, HELP | 5,781 | 17,369 | 10,791 |
| Case 2-5 | HAVE, PROGRAM | 3,029 | 17,369 | 5,327 |
| Case 2-6 | HELP, PROGRAM | 1,949 | 17,369 | 5,327 |



Monte Carlo trials. In this way, the small data set experiment made it possible to verify our theoretical results, including the approximations in the variance formulas.

The larger experiment contains many words with a large range of frequencies; and hence the experiment was repeated just six times (i.e., six different permutations). With such a large range of frequencies and sampling rates, there is a danger that some samples would be too small, especially for very rare words and very low sampling rates. A floor was imposed to make sure that every sample contains at least 20 documents.

### 4.1 Results from Large Monte Carlo Experiment

Figure 7 shows that the proposed methods (solid lines) are better than the baselines (dashed lines), in terms of MSE, estimated by the large Monte Carlo experiment over the small data set, as described herein. Note that errors generally decrease with sampling rate, as one would expect, at least for the methods that take advantage of the sample. The independence baseline ($\hat{a}_{IND}$), which does not take advantage of the sample, has very large errors. The sample is a very useful source of information; even a small sample is much better than no sample.

The recommended quadratic approximation, $\hat{a}_{MLE,a}$, is remarkably close to the exact MLE solution. Both of the proposed methods, $\hat{a}_{MLE,a}$ and $\hat{a}_{MLE}$ (solid lines), have

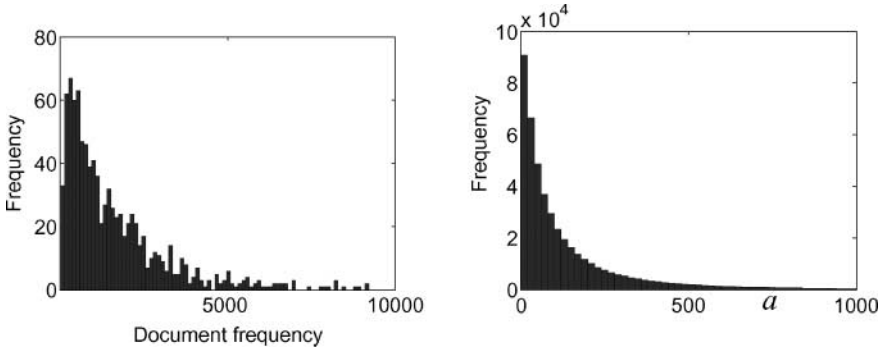



**Figure 6**
Large data set: histograms of document frequencies, $df$ (left), and co-occurrences, $a$ (right). Left: max document frequency $df$ = 42,564, median = 1135, mean = 2135, standard deviation = 3628. Right: max co-occurrence $a$ = 33,045, mean = 188, median = 74, standard deviation = 459.

much smaller MSE than the margin-free baseline $\hat{a}_{MF}$ (dashed lines), especially at low sampling rates. When we know the margins, we ought to use them.

Note that MSE can be decomposed into variance and bias: $\text{MSE}(\hat{a}) = \text{E}\,(\hat{a} - a)^2 = \text{Var}\,(\hat{a}) + \text{Bias}^2\,(\hat{a})$. If $\hat{a}$ is unbiased, $\text{MSE}(\hat{a}) = \text{Var}\,(\hat{a}) = \text{SE}^2\,(\hat{a})$, where SE is called "standard error."

*4.1.1 Margin Constraints Improve Smoothing.* Though not a major emphasis of this paper, Figure 8 shows that smoothing is effective at low sampling rates, but only for those methods that take advantage of the margin constraints (solid lines as opposed to dashed lines). Figure 8 compares smoothed estimates ($\hat{a}_{MLE}$, $\hat{a}_{MLE,a}$, and $\hat{a}_{MF}$) with their unsmoothed counterparts. The $y$-axis reports percentage improvement of the MSE due to smoothing. Smoothing helps the proposed methods (solid lines) for all six word pairs, and hurts the baseline methods (dashed lines), for most of the six word pairs. We believe margin constraints keep the smoother from wandering too far astray; without margin constraints, smoothing can easily do more harm than good, especially when the smoother isn't very good. In this experiment, we used the simple "add-one" smoother that replaces $a_s$, $b_s$, $c_s$, and $d_s$ with $a_s + 1$, $b_s + 1$, $c_s + 1$, and $d_s + 1$, respectively. We could have used a more sophisticated smoother (e.g., Good–Turing), but if we had done so, it would have been harder to see how the margin constraints keep the smoother from wandering too far astray.

*4.1.2 Monte Carlo Verification of Variance Formula.* How accurate is the approximation of the variance in Equations (9) and (11)? Figure 9 shows that the Monte Carlo simulation is remarkably close to the theoretical formula (9). Formula (11) is the same as (9), except that $\text{E}\left(\frac{D}{D_s}\right)$ is replaced with the approximation

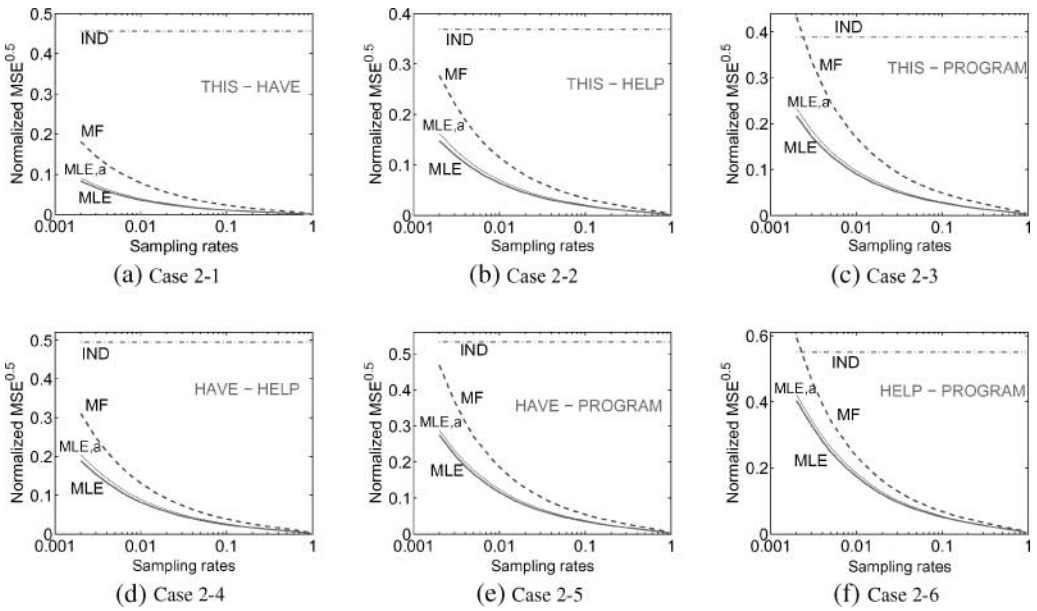



**Figure 7**

The proposed estimator, $\hat{a}_{MLE}$, outperforms the margin-free baseline, $\hat{a}_{MF}$, in terms of $\frac{\sqrt{\text{MSE}}}{a}$. The quadratic approximation, $\hat{a}_{MLE,a}$, is close to $\hat{a}_{MLE}$. All methods are better than assuming independence (IND).

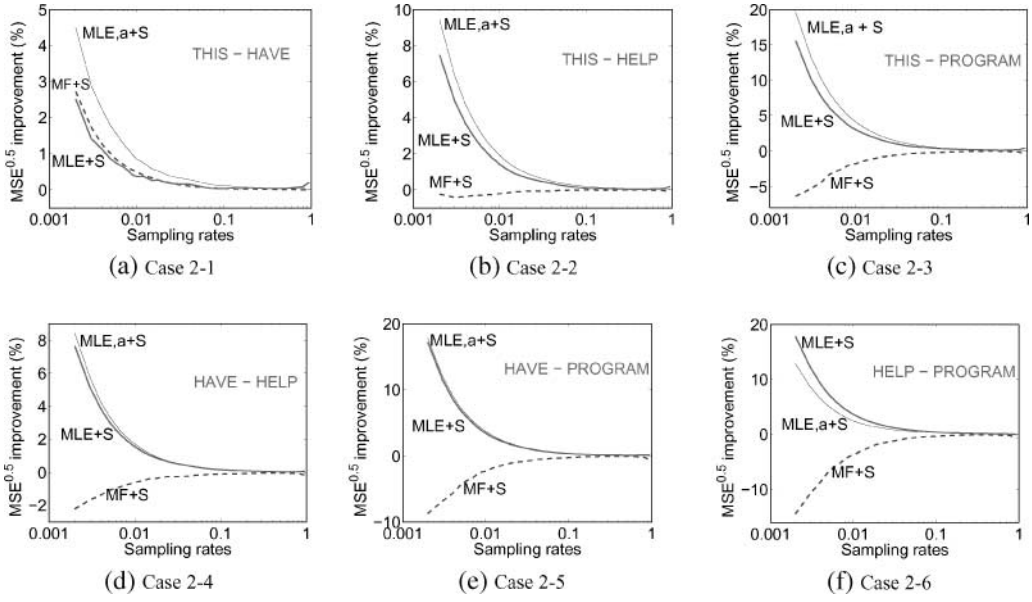



**Figure 8**
Smoothing improves the proposed MLE estimators but hurts the margin-free estimator in most cases. The vertical axis is the percentage of relative improvement in $\sqrt{\text{MSE}}$ of each smoothed estimator with respect to its un-smoothed version.

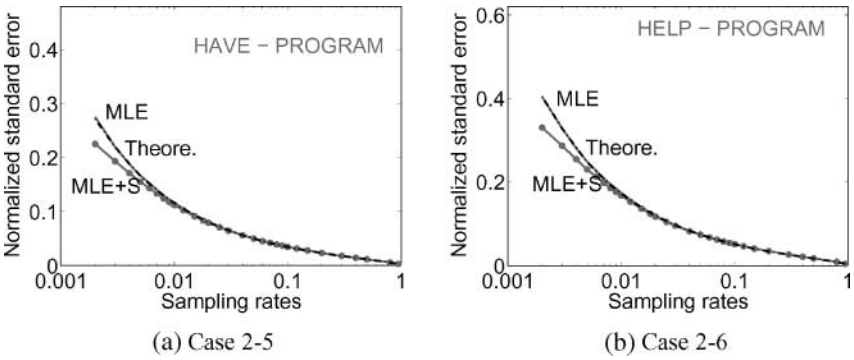



**Figure 9**
Normalized standard error, $\frac{\text{SE}(\hat{a})}{a}$, for the MLE. The theoretical variance formula (9) fits the simulation results so well that the curves are indistinguishable. Also, smoothing is effective in reducing variance, especially at low sampling rates.

$\max\left(\frac{f_1}{k_1}, \frac{f_2}{k_2}\right)$. Theoretically, we expect $\max\left(\frac{f_1}{k_1}, \frac{f_2}{k_2}\right) \leq \text{E}\left(\frac{D}{D_s}\right)$. Figure 10 verifies the inequality, and shows that the inequality is not too far from an equality. We will use (11) instead of (9), because the differences are not too large, and (11) is more convenient.

*4.1.3 Monte Carlo Estimate of Bias.* Finally, we also compare the biases in Figure 11 for Case 2-5 and Case 2-6. The figure shows that the MLE estimator is essentially unbiased.

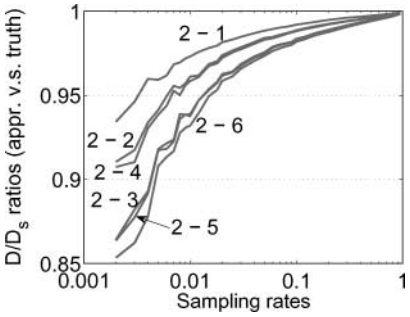



**Figure 10**

For all 6 cases, the ratios $\max\left(\frac{f_1}{k_1}, \frac{f_2}{k_2}\right) / \mathrm{E}\left(\frac{D}{D_s}\right)$ are close to 1, and the differences roughly monotonically decrease with increasing sampling rates. When the sampling rates $\geq 0.005$ (roughly the sketch sizes $\geq 20$), $\max\left(\frac{f_1}{k_1}, \frac{f_2}{k_2}\right)$ is an accurate approximation of $\mathrm{E}\left(\frac{D}{D_s}\right)$.

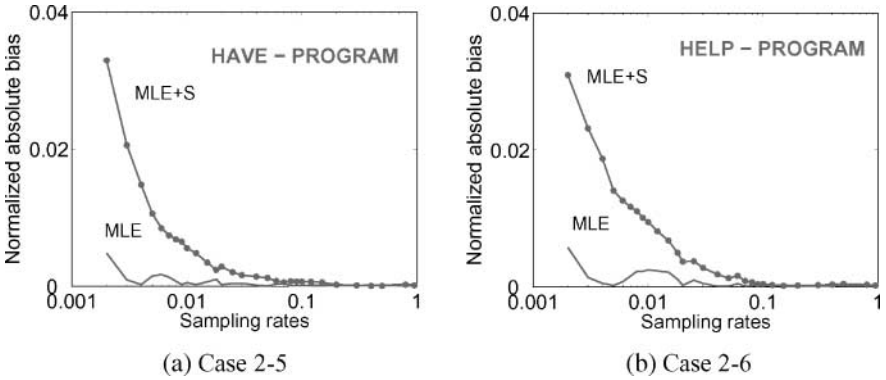



**Figure 11**

Biases in terms of $\frac{|\mathrm{E}(\hat{a})-a|}{a}$. $\hat{a}_{MLE}$ is practically unbiased. Smoothing increases bias slightly.

### 4.2 Results from Large Data Set Experiment

In Figure 12, the large data set experiment confirms the findings of the large Monte Carlo experiment: The proposed MLE method is better than the margin-free and independence baselines. The recommended quadratic approximation, $\hat{a}_{MLE,a}$, is close to the exact solution, $\hat{a}_{MLE}$.

### 4.3 Rank Retrieval by Cosine

We are often interested in finding top ranking pairs according to some measure of similarity such as cosine. Performance improves with sampling rate for this task (as well as almost any other task; there is no data like more data), but nevertheless, Figure 13 shows that we can find many of the top ranking pairs, even at low sampling rates.

Note that the estimate of cosine, $\frac{a}{\sqrt{f_1 f_2}}$, depends solely on the estimate of $a$, because we know the margins, $f_1$ and $f_2$. If we sort word pairs by their cosines, using estimates of $a$ based on a small sample, the rankings will hopefully be close to what we would

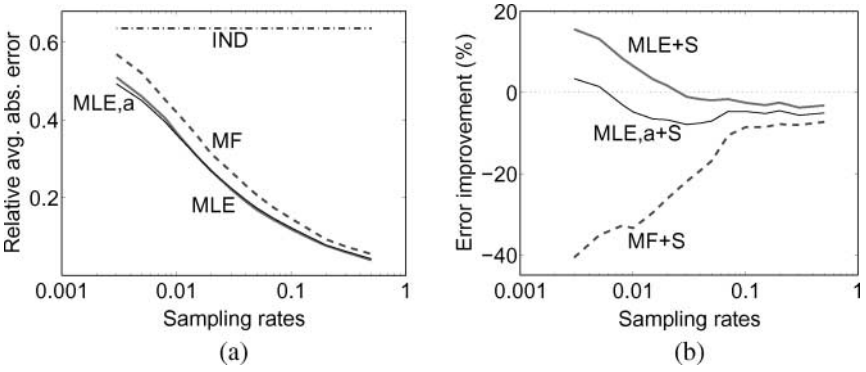



**Figure 12**
(a) The proposed MLE methods (solid lines) have smaller errors than the baselines (dashed lines). We report the mean absolute errors (normalized by the mean co-occurrences, 188). All curves are averaged over six permutations. The two solid lines, the proposed MLE and the recommended quadratic approximation, are close to one another. Both are well below the margin-free (MF) baseline and the independence (IND) baseline. (b) Percentage of improvement due to smoothing. Smoothing helps MLE, but hurts MF.

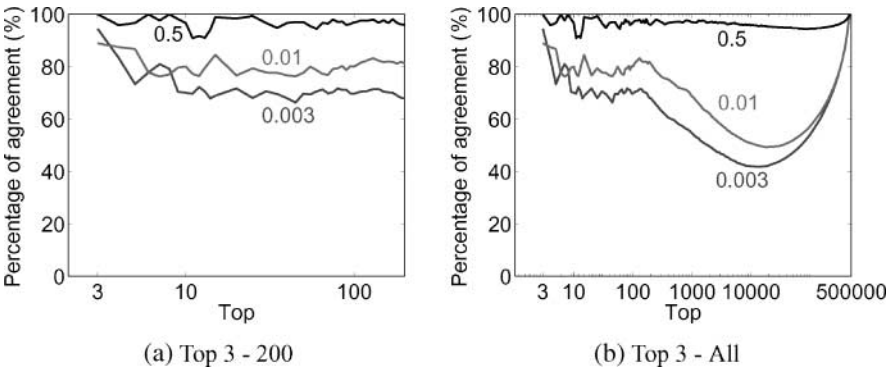



**Figure 13**
We can find many of the most obvious associations with very little work. Two sets of cosine scores were computed for the 468,028 pairs in the large dataset experiment. The gold standard scores were computed over the entire dataset, whereas sample scores were computed over a sample of the data set. The plots show the percentage of agreement between these two lists, as a function of $S$. As expected, agreement rates are high ($\approx 100\%$) at high sampling rates (0.5). But it is reassuring that agreement rates remain pretty high ($\approx 70\%$) even when we crank the sampling rate way down (0.003).

obtain if we used the entire data set. This section will compare the rankings based on a small sample to a gold standard, the rankings based on the entire data set.

How should we evaluate rankings? We follow the suggestion in Ravichandran, Pantel, and Hovy (2005) of reporting the percentage of agreements in the top-$S$. That is, we compare the top-$S$ pairs based on a sample with the top-$S$ pairs based on the entire data set. We report the intersection of the two lists, normalized by $S$. Figure 13(a) emphasizes high precision region ($3 \leq S \leq 200$), whereas Figure 13(b) emphasizes higher recall, extending $S$ to cover all 468,028 pairs in the large dataset experiment. Of course, agreement rates are high at high sampling rates. For example, we have nearly $\approx 100\%$ agreement at a sampling rate of 0.5. It is reassuring that agreement rates remain fairly high ($\approx 70\%$), even when we push the sampling rate way down

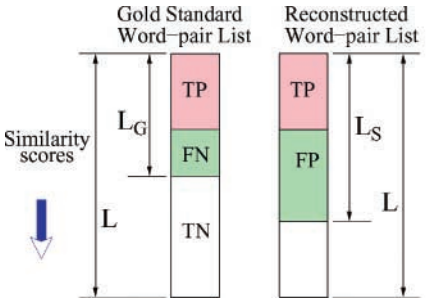
(0.003). In other words, we can find many of the most obvious associations with very little work.

The same comparisons can be evaluated in terms of precision and recall, by fixing the top-$L_G$ gold standard list but varying the length of the sample list $L_S$. More precisely, recall = relevant/$L_G$, and precision = relevant/$L_S$, where "relevant" means the retrieved pairs in the gold standard list. Figure 14 gives a graphical representation of this evaluation scheme, using notation in Manning and Schutze (1999), Chapter 8.1.

Figure 15 presents the precision–recall curves for $L_G = 1\%L$ and $10\%L$, where $L = 468,028$. For each $L_G$, there is one precision–recall curve corresponding to each sampling rate. All curves indicate the precision–recall trade-off and that the only way to improve both precision and recall simultaneously is to increase the sampling rate.

### 4.4 Summary

To summarize the main results of the large and small data set experiments, we found that the proposed MLE (and the recommended quadratic approximation) have smaller




| | Relevant | Irrelevant |
|---|---|---|
| Retrieved | TP ( a ) | FP ( b ) |
| Not Retrieved | FN ( c ) | TN ( d ) |

$$\text{Precision} = \frac{TP}{TP + FP} = \frac{a}{a + b}$$

$$\text{Recall} = \frac{TP}{TP + FN} = \frac{a}{a + c}$$

**Figure 14**
Definitions of recall and precision. $L$ = total number of pairs. $L_G$ = number of pairs from the top of the gold standard similarity list. $L_S$ = number of pairs from the top of the reconstructed similarity list.

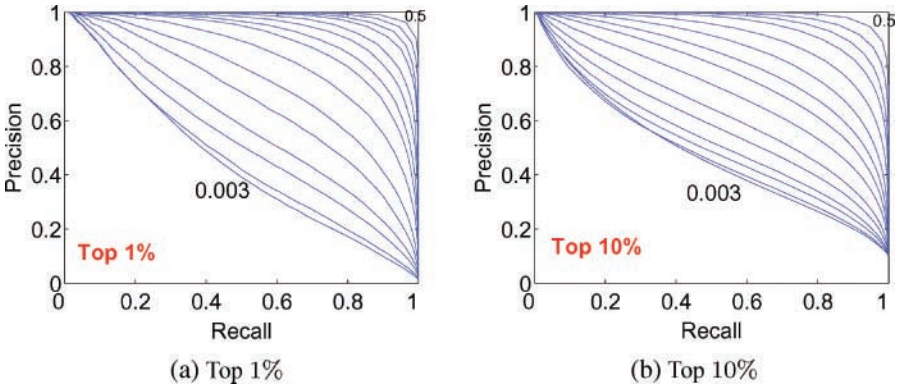



**Figure 15**
Precision–recall curves in retrieving the top 1% and top 10% gold standard pairs, at different sampling rates from 0.003 to 0.5. Note that the precision is always larger than $\frac{L_G}{L}$.

errors than the two baselines (the MF baseline and the independence (IND) baseline). Margin constraints improve smoothing, because the margin constraints keep the smoother from wandering too far astray. Monte Carlo simulations verified the variance formulas (9) and (11), and showed that the proposed MLE method is essentially unbiased. The ranking experiment showed that we can find many of the most obvious associations with very little work.

## 5. The Maximum Likelihood Estimator (MLE)

Section 4 evaluated the proposed method empirically; this section will explore the statistical theory behind the method. The task is to estimate the contingency table $(a, b, c, d)$ from the sample contingency table $(a_s, b_s, c_s, d_s)$, the margins, and $D$.

We can factor the (full) likelihood (probability mass function, PMF) $\mathbf{Pr}(a_s, b_s, c_s, d_s; a)$ into

$$\mathbf{Pr}(a_s, b_s, c_s, d_s; a) = \mathbf{Pr}(a_s, b_s, c_s, d_s | D_s; a) \times \mathbf{Pr}(D_s; a) \tag{12}$$

We seek the $a$ that maximizes the *partial likelihood* $\mathbf{Pr}(a_s, b_s, c_s, d_s | D_s; a)$, that is,

$$\hat{a}_{MLE} = \underset{a}{\operatorname{argmax}} \mathbf{Pr}\,(a_s, b_s, c_s, d_s | D_s; a) = \underset{a}{\operatorname{argmax}} \log \mathbf{Pr}\,(a_s, b_s, c_s, d_s | D_s; a) \tag{13}$$

$\mathbf{Pr}(a_s, b_s, c_s, d_s | D_s; a)$ is just the PMF of a two-way sample contingency table. That is relatively straightforward, but $\mathbf{Pr}(D_s; a)$ is difficult. As illustrated in Figure 16, there is no strong dependency of $D_s$ on $a$, and therefore, we can focus on the easy part.

Before we delve into maximizing $\mathbf{Pr}(a_s, b_s, c_s, d_s | D_s; a)$ under margin constraints, we will first consider two simplifications, which lead to two baseline estimators. The independence baseline does not use any samples, whereas the margin-free baseline does not take advantage of the margins.

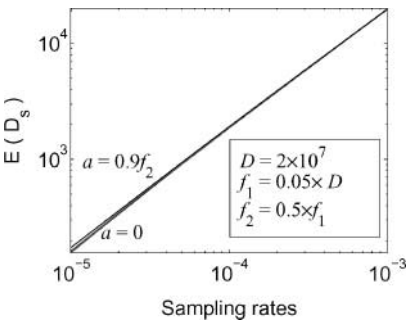



**Figure 16**
This experiment shows that $\mathrm{E}(D_s)$ is not sensitive to $a$. $D = 2 \times 10^7, f_1 = D/20, f_2 = f_1/2$. The different curves correspond to $a$ = 0, 0.05, 0.2, 0.5, and $0.9\,f_2$. These curves are almost indistinguishable except at very low sampling rates. Note that, at sampling rate = $10^{-5}$, the sample size $k_2 = 5$ only.

### 5.1 The Independence Baseline

Independence assumptions are often made in databases (Garcia-Molina, Ullman, and Widom 2002, Chapter 16.4) and NLP (Manning and Schutze 1999, Chapter 13.3). When two words $W_1$ and $W_2$ are independent, the size of intersections, $a$, follows a hypergeometric distribution,

$$\mathbf{Pr}(a) = \binom{f_1}{a}\binom{D-f_1}{f_2-a}\Big/\binom{D}{f_2}, \tag{14}$$

where $\binom{n}{m} = \frac{n!}{m!(n-m)!}$. This distribution suggests an estimator

$$\hat{a}_{IND} = \mathrm{E}(a) = \frac{f_1 f_2}{D}. \tag{15}$$

Note that (14) is also a common null-hypothesis distribution in testing the independence of a two-way contingency table, that is, the so-called Fisher's exact test (Agresti 2002, Section 3.5.1).

### 5.2 The Margin-Free Baseline

Conditional on $D_s$, the sample contingency table $(a_s, b_s, c_s, d_s)$ follows the **multivariate hypergeometric** distribution with moments[4]

$$\begin{aligned}
&\mathrm{E}(a_s|D_s) = \frac{D_s}{D}a, \quad \mathrm{E}(b_s|D_s) = \frac{D_s}{D}b, \quad \mathrm{E}(c_s|D_s) = \frac{D_s}{D}c, \quad \mathrm{E}(d_s|D_s) = \frac{D_s}{D}d, \\
&\mathrm{Var}(a_s|D_s) = D_s\frac{a}{D}\left(1-\frac{a}{D}\right)\frac{D-D_s}{D-1}
\end{aligned} \tag{16}$$

where the term $\frac{D-D_s}{D-1} \approx 1 - \frac{D_s}{D}$, is known as the "finite population correction factor."

An unbiased estimator and its variance would be

$$\hat{a}_{MF} = \frac{D}{D_s}a_s, \quad \mathrm{Var}(\hat{a}_{MF}|D_s) = \frac{D^2}{D_s^2}\mathrm{Var}(a_s|D_s) = \frac{D}{D_s}\frac{1}{\frac{1}{a}+\frac{1}{D-a}}\frac{D-D_s}{D-1}. \tag{17}$$

We refer to this estimator as "margin-free" because it does not take advantage of the margins.

The multivariate hypergeometric distribution can be simplified to a *multinomial* assuming "sample-with-replacement," which is often a good approximation when $\frac{D_s}{D}$ is small. According to the multinomial model, an estimator and its variance would be:

$$\hat{a}_{MF,r} = \frac{D}{D_s}a_s, \qquad \mathrm{Var}(\hat{a}_{MF,r}|D_s) = \frac{D}{D_s}\frac{1}{\frac{1}{a}+\frac{1}{D-a}} \tag{18}$$

That is, for the margin-free model, the "sample-with-replacement" simplification still results in the same estimator but slightly overestimates the variance.

4 http://www.ds.unifi.it/VL/VL_EN/urn/urn4.html.

Note that these expectations in (16) hold both when the margins are known, as well as when they are not known, because the samples $(a_s, b_s, c_s, d_s)$ are obtained randomly without consulting the margins. Of course, when we know the margins, we can do better than when we don't.

### 5.3 The Exact MLE with Margin Constraints

Considering the margin constraints, the partial likelihood $\mathbf{Pr}(a_s, b_s, c_s, d_s | D_s; a)$ can be expressed as a function of a single unknown parameter, $a$:

$$\begin{aligned}
\mathbf{Pr}(a_s, b_s, c_s, d_s | D_s; a) &= \frac{\binom{a}{a_s}\binom{b}{b_s}\binom{c}{c_s}\binom{d}{d_s}}{\binom{a+b+c+d}{a_s+b_s+c_s+d_s}} = \frac{\binom{a}{a_s}\binom{f_1-a}{b_s}\binom{f_2-a}{c_s}\binom{D-f_1-f_2+a}{d_s}}{\binom{D}{D_s}} \\
&\propto \frac{a!}{(a-a_s)!} \times \frac{(f_1-a)!}{(f_1-a-b_s)!} \times \frac{(f_2-a)!}{(f_2-a-c_s)!} \times \frac{(D-f_1-f_2+a)!}{(D-f_1-f_2+a-d_s)!} \\
&= \prod_{i=0}^{a_s-1}(a-i) \times \prod_{i=0}^{b_s-1}(f_1-a-i) \times \prod_{i=0}^{c_s-1}(f_2-a-i) \times \prod_{i=0}^{d_s-1}(D-f_1-f_2+a-i)
\end{aligned} \tag{19}$$

where the multiplicative terms not mentioning $a$ are discarded, because they do not contribute to the MLE.

Let $\hat{a}_{MLE}$ be the value of $a$ that maximizes the partial likelihood (19), or equivalently, maximizes the log likelihood, $\log \mathbf{Pr}(a_s, b_s, c_s, d_s | D_s; a)$:

$$\sum_{i=0}^{a_s-1}\log(a-i) + \sum_{i=0}^{b_s-1}\log\left(f_1-a-i\right) + \sum_{i=0}^{c_s-1}\log\left(f_2-a-i\right) + \sum_{i=0}^{d_s-1}\log\left(D-f_1-f_2+a-i\right)$$

whose first derivative, $\frac{\partial \log \mathbf{Pr}(a_s,b_s,c_s,d_s|D_s;a)}{\partial a}$, is

$$\sum_{i=0}^{a_s-1}\frac{1}{a-i} - \sum_{i=0}^{b_s-1}\frac{1}{f_1-a-i} - \sum_{i=0}^{c_s-1}\frac{1}{f_2-a-i} + \sum_{i=0}^{d_s-1}\frac{1}{D-f_1-f_2+a-i} \tag{20}$$

Because the second derivative, $\frac{\partial^2 \log \mathbf{Pr}(a_s,b_s,c_s,d_s|D_s;a)}{\partial a^2}$,

$$-\sum_{i=0}^{a_s-1}\frac{1}{(a-i)^2} - \sum_{i=0}^{b_s-1}\frac{1}{(f_1-a-i)^2} - \sum_{i=0}^{c_s-1}\frac{1}{(f_2-a-i)^2} - \sum_{i=0}^{d_s-1}\frac{1}{(D-f_1-f_2+a-i)^2}$$

is negative, the log likelihood function is concave, and therefore, there is a unique maximum. One could solve (20) for $\frac{\partial \log \mathbf{Pr}(a_s,b_s,c_s,d_s|D_s;a)}{\partial a} = 0$ numerically, but it turns out there is a more direct solution using the updating formula from (19):

$$\mathbf{Pr}(a_s, b_s, c_s, d_s | D_s; a) = \mathbf{Pr}(a_s, b_s, c_s, d_s | D_s; a-1) \times g(a)$$

Because we know that the MLE exists and is unique, it suffices to find the $a$ such that $g(a) = 1$,

$$g(a) = \frac{a}{a - a_s}\frac{f_1 - a + 1 - b_s}{f_1 - a + 1}\frac{f_2 - a + 1 - c_s}{f_2 - a + 1}\frac{D - f_1 - f_2 + a}{D - f_1 - f_2 + a - d_s} = 1 \tag{21}$$

which is cubic in $a$ (because the fourth term vanishes).

We recommend a straightforward numerical procedure for solving $g(a) = 1$. Note that $g(a) = 1$ is equivalent to $q(a) = \log g(a) = 0$. The first derivative of $q(a)$ is

$$\begin{aligned} q'(a) = &\left(\frac{1}{f_1 - a + 1} - \frac{1}{f_1 - a + 1 - b_s}\right) + \left(\frac{1}{f_2 - a + 1} - \frac{1}{f_2 - a + 1 - c_s}\right) \\ &+ \left(\frac{1}{D - f_1 - f_2 + a} - \frac{1}{D - f_1 - f_2 + a - d_s}\right) + \left(\frac{1}{a} - \frac{1}{a - a_s}\right) \end{aligned} \tag{22}$$

We can solve for $q(a) = 0$ iteratively using Newton's method: $a^{(\text{new})} = a^{(\text{old})} - \frac{q(a^{(\text{old})})}{q'(a^{(\text{old})})}$. See Appendix 1 for a C code implementation.

### 5.4 The "Sample-with-Replacement" Simplification

Under the "sample-with-replacement" assumption, the likelihood function is slightly simpler:

$$\begin{aligned} \mathbf{Pr}(a_s, b_s, c_s, d_s | D_s; a, r) &= \binom{D_s}{a_s, b_s, c_s, d_s}\left(\frac{a}{D}\right)^{a_s}\left(\frac{b}{D}\right)^{b_s}\left(\frac{c}{D}\right)^{c_s}\left(\frac{d}{D}\right)^{d_s} \\ &\propto a^{a_s}(f_1 - a)^{b_s}(f_2 - a)^{c_s}(D - f_1 - f_2 + a)^{d_s} \end{aligned} \tag{23}$$

Setting the first derivative of the log likelihood to be zero yields a cubic equation:

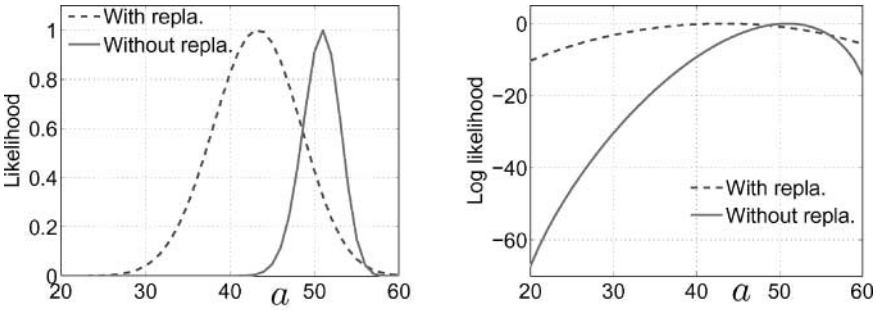
$$\frac{a_s}{a} - \frac{b_s}{f_1 - a} - \frac{c_s}{f_2 - a} + \frac{d_s}{D - f_1 - f_2 + a} = 0 \tag{24}$$

As shown in Section 5.2, using the margin-free model, the "sample-with-replacement" assumption amplifies the variance but does not change the estimation. With our proposed MLE, the "sample-with-replacement" assumption will change the estimation, although in general we do not expect the differences to be large. Figure 17 gives an (exaggerated) example, to show the concavity of the log likelihood and the difference caused by assuming "sample-with-replacement."

### 5.5 A Convenient Practical Quadratic Approximation

Solving a cubic equation for the exact MLE may be so inconvenient that one may prefer the less accurate margin-free baseline because of its simplicity. This section derives a convenient closed-form quadratic approximation to the exact MLE.

The idea is to assume "sample-with-replacement" and that one can identify $a_s$ from $\mathrm{K}_1$ without knowledge of $\mathrm{K}_2$. In other words, we assume $a_s^{(1)} \sim \text{Binomial}\left(a_s + b_s, \frac{a}{f_1}\right)$,




(a) Likelihood (b) Log likelihood

**Figure 17**
An example: $a_s = 20$, $b_s = 40$, $c_s = 40$, $d_s = 800$, $f_1 = f_2 = 100$, $D = 1000$. The estimated $\hat{a} = 43$ for "sample-with-replacement," and $\hat{a} = 51$ for "sample-without-replacement." (a) The likelihood profile, normalized to have a maximum = 1. (b) The log likelihood profile, normalized to have a maximum = 0.

$a_s^{(2)} \sim$ Binomial $\left(a_s + c_s, \frac{a}{f_2}\right)$, and $a_s^{(1)}$ and $a_s^{(2)}$ are independent with $a_s^{(1)} = a_s^{(2)} = a_s$. The PMF of $\left(a_s^{(1)}, a_s^{(2)}\right)$ is a product of two binomials:

$$\left[\binom{f_1}{a_s + b_s}\left(\frac{a}{f_1}\right)^{a_s}\left(\frac{f_1 - a}{f_1}\right)^{b_s}\right] \times \left[\binom{f_2}{a_s + c_s}\left(\frac{a}{f_2}\right)^{a_s}\left(\frac{f_2 - a}{f_2}\right)^{c_s}\right]$$
$$\propto a^{2a_s}\left(f_1 - a\right)^{b_s}\left(f_2 - a\right)^{c_s} \tag{25}$$

Setting the first derivative of the logarithm of (25) to be zero, we obtain

$$\frac{2a_s}{a} - \frac{b_s}{f_1 - a} - \frac{c_s}{f_2 - a} = 0 \tag{26}$$

which is quadratic in $a$ and has a convenient closed-form solution:

$$\hat{a}_{MLE,a} = \frac{f_1\left(2a_s + c_s\right) + f_2\left(2a_s + b_s\right) - \sqrt{\left(f_1\left(2a_s + c_s\right) - f_2\left(2a_s + b_s\right)\right)^2 + 4f_1 f_2 b_s c_s}}{2\left(2a_s + b_s + c_s\right)} \tag{27}$$

The second root can be ignored because it is always out of range:

$$\frac{f_1\left(2a_s + c_s\right) + f_2\left(2a_s + b_s\right) + \sqrt{\left(f_1\left(2a_s + c_s\right) - f_2\left(2a_s + b_s\right)\right)^2 + 4f_1 f_2 b_s c_s}}{2\left(2a_s + b_s + c_s\right)}$$
$$\geq \frac{f_1\left(2a_s + c_s\right) + f_2\left(2a_s + b_s\right) + \left|f_1\left(2a_s + c_s\right) - f_2\left(2a_s + b_s\right)\right|}{2\left(2a_s + b_s + c_s\right)}$$
$$\geq \begin{cases} f_1 & \text{if } f_1\left(2a_s + c_s\right) \geq f_2\left(2a_s + b_s\right) \\ f_2 & \text{if } f_1\left(2a_s + c_s\right) < f_2\left(2a_s + b_s\right) \end{cases} \geq \min(f_1, f_2)$$

The evaluation in Section 4 showed that $\hat{a}_{MLE,a}$ is close to $\hat{a}_{MLE}$.

### 5.6 The Conditional Variance and Bias

Usually, a maximum likelihood estimator is nearly unbiased. Furthermore, assuming "sample-with-replacement," we can apply the large sample theory[5] (Lehmann and Casella 1998, Theorem 6.3.10), which says that $\hat{a}_{MLE}$ is asymptotically unbiased and converges in distribution to a Normal with mean $a$ and variance $\frac{1}{\mathrm{I}(a)}$, where $\mathrm{I}(a)$, the expected Fisher Information, is

$$\begin{aligned}
\mathrm{I}(a) &= -\mathrm{E}\left(\frac{\partial^2}{\partial a^2}\log \mathbf{Pr}\left(a_s, b_s, c_s, d_s | D_s; a, r\right)\right) \\
&= \mathrm{E}\left(\frac{a_s}{a^2} + \frac{b_s}{(f_1-a)^2} + \frac{c_s}{(f_2-a)^2} + \frac{d_s}{(D-f_1-f_2+a)^2}\middle| D_s\right) \\
&= \frac{\mathrm{E}(a_s|D_s)}{a^2} + \frac{\mathrm{E}(b_s|D_s)}{\left(f_1-a\right)^2} + \frac{\mathrm{E}(c_s|D_s)}{\left(f_2-a\right)^2} + \frac{\mathrm{E}(d_s|D_s)}{\left(D-f_1-f_2+a\right)^2} \\
&= \frac{D_s}{D}\left(\frac{1}{a} + \frac{1}{f_1-a} + \frac{1}{f_2-a} + \frac{1}{D-f_1-f_2+a}\right)
\end{aligned} \tag{28}$$

where we evaluate $\mathrm{E}(a_s|D_s)$, $\mathrm{E}(b_s|D_s)$, $\mathrm{E}(c_s|D_s)$, $\mathrm{E}(d_s|D_s)$ by (16).

For "sampling-without-replacement," we correct the asymptotic variance $\frac{1}{\mathrm{I}(a)}$ by multiplying by the finite population correction factor $1 - \frac{D_s}{D}$:

$$\mathrm{Var}\left(\hat{a}_{MLE}|D_s\right) \approx \frac{1}{\mathrm{I}(a)}\left(1 - \frac{D_s}{D}\right) = \frac{\frac{D}{D_s} - 1}{\frac{1}{a} + \frac{1}{f_1-a} + \frac{1}{f_2-a} + \frac{1}{D-f_1-f_2+a}} \tag{29}$$

Comparing (17) with (29), we know that $\mathrm{Var}\left(\hat{a}_{MLE}|D_s\right) < \mathrm{Var}\left(\hat{a}_{MF}|D_s\right)$, and the difference could be substantial. In other words, when we know the margins, we ought to use them.

### 5.7 The Unconditional Variance and Bias

Errors are a combination of variance and bias. Fortunately, we don't need to be concerned about bias, at least asymptotically:

$$\mathrm{E}\left(\hat{a}_{MLE} - a\right) = \mathrm{E}\left(\mathrm{E}\left(\hat{a}_{MLE} - a|D_s\right)\right) \rightarrow \mathrm{E}(0) = 0 \tag{30}$$

The unconditional variance can be computed using the conditional variance formula:

$$\begin{aligned}
\mathrm{Var}\left(\hat{a}_{MLE}\right) &= \mathrm{E}\left(\mathrm{Var}\left(\hat{a}_{MLE}\,|D_s\right)\right) + \mathrm{Var}\left(\mathrm{E}\left(\hat{a}_{MLE}\,|D_s\right)\right) \\
&\rightarrow \frac{\mathrm{E}\left(\frac{D}{D_s}\right) - 1}{\frac{1}{a} + \frac{1}{f_1-a} + \frac{1}{f_2-a} + \frac{1}{D-f_1-f_2+a}}
\end{aligned} \tag{31}$$

5 See Rosen (1972a, 1972b) for the rigorous regularity conditions that ensure convergence in the case of "sample-without-replacement."

because $\mathrm{E}(\hat{a}_{MLE}|D_s) \rightarrow a$, which is a constant. Hence $\mathrm{Var}(\mathrm{E}(\hat{a}_{MLE}|D_s)) \rightarrow 0$.

To evaluate $\mathrm{E}\left(\frac{D}{D_s}\right)$ exactly, we need PMF $\mathbf{Pr}(D_s; a)$, which is unavailable. Even if it were available, $\mathrm{E}\left(\frac{D}{D_s}\right)$ probably wouldn't have a convenient closed-form.

Here we recommend the approximations, (3) and (4), mentioned previously. To derive these approximations, recall that $D_s = \min(\max(\mathrm{K}_1), \max(\mathrm{K}_2))$. Using the discrete order statistics distribution (David 1981, Exercise 2.1.4),[6] we obtain:

$$\mathrm{E}(\max(\mathrm{K}_1)) = \frac{k_1(D+1)}{f_1+1} \approx \frac{k_1}{f_1}D, \qquad \mathrm{E}(\max(\mathrm{K}_2)) \approx \frac{k_2}{f_2}D \tag{32}$$

The min function can be considered to be concave. By Jensen's inequality (see Cover and Thomas 1991, Theorem 2.6.2), we know that

$$\begin{aligned}\mathrm{E}\left(\frac{D_s}{D}\right) &= \mathrm{E}\left(\min\left(\frac{\max(\mathrm{K}_1 k_1)}{D}, \frac{\max(\mathrm{K}_2)}{D}\right)\right) \\ &\leq \min\left(\frac{\mathrm{E}(\max(\mathrm{K}_1)}{D}, \frac{\mathrm{E}(\max(\mathrm{K}_2)}{D}\right) = \min\left(\frac{k_1}{f_1}, \frac{k_2}{f_2}\right)\end{aligned} \tag{33}$$

The reciprocal function is convex. Again by Jensen's inequality, we have

$$\mathrm{E}\left(\frac{D}{D_s}\right) = \mathrm{E}\left(\frac{1}{D_s/D}\right) \geq \frac{1}{\mathrm{E}\left(\frac{D_s}{D}\right)} \geq \max\left(\frac{f_1}{k_1}, \frac{f_2}{k_2}\right) \tag{34}$$

By replacing the inequalities with equalities, we obtain (35) and (36):

$$\mathrm{E}\left(\frac{D_s}{D}\right) \approx \min\left(\frac{k_1}{f_1}, \frac{k_2}{f_2}\right) \tag{35}$$

$$\mathrm{E}\left(\frac{D}{D_s}\right) \approx \max\left(\frac{f_1}{k_1}, \frac{f_2}{k_2}\right) \tag{36}$$

In our experiments, when the sample size is reasonably large ($D_s \geq 20$), the errors in (35) and (36) are usually within 5%.

Approximations (35) and (36) provide an intuitive relationship between two views of the sampling rate: (a) $\frac{D_s}{D}$, which depends on corpus size and (b) $\frac{k}{f}$, which depends on the size of the postings. The difference between these two views is important when the term-by-document matrix is sparse, which is often the case in practice.

Using (36), we obtain the following approximation for the unconditional variance:

$$\mathrm{Var}(\hat{a}_{MLE}) \approx \frac{\max\left(\frac{f_1}{k_1}, \frac{f_2}{k_2}\right) - 1}{\frac{1}{a} + \frac{1}{f_1 - a} + \frac{1}{f_2 - a} + \frac{1}{D - f_1 - f_2 + a}} \tag{37}$$

6 Also, see http://www.ds.unifi.it/VL/VL_EN/urn/urn5.html.

### 5.8 The Variance of $h(\hat{a}_{MLE})$

We can estimate any function $h(a)$ by $h(\hat{a}_{MLE})$. In practical applications, $h$ could be any measure of association including cosine, resemblance, mutual information, etc. When $h(a)$ is a nonlinear function of $a$, $h(\hat{a}_{MLE})$ will be biased. One can remove the bias to some extent using Taylor expansions. See some examples in Li and Church (2005). Bias correction is important for small samples and highly nonlinear $h$'s (e.g., the log likelihood ratio, LLR).

The bias of $h(\hat{a}_{MLE})$ decreases with sample size. Precisely, the **delta method** (Agresti 2002, Chapter 3.1.5) says that $h(\hat{a}_{MLE})$ is asymptotically unbiased and the variance of $h(\hat{a}_{MLE})$ is

$$\text{Var}(h(\hat{a}_{MLE})) \rightarrow \text{Var}(\hat{a}_{MLE})(h'(a))^2 \tag{38}$$

provided $h'(a)$ exists and is non-zero. Non-asymptotically, it is easy to show that

$$\text{Var}(h(\hat{a}_{MLE})) \geq \text{Var}(\hat{a}_{MLE})(h'(a))^2 \qquad \text{if } h(a) \text{ is convex} \tag{39}$$

$$\text{Var}(h(\hat{a}_{MLE})) \leq \text{Var}(\hat{a}_{MLE})(h'(a))^2 \qquad \text{if } h(a) \text{ is concave} \tag{40}$$

### 5.9 How Many Samples Are Sufficient?

The answer depends on the trade-off between computational costs (time and space) and estimation errors. For very infrequent words, we might afford to sample 100%. In general, a reasonable criterion is the coefficient of variation, cv = $\frac{\text{SE}(\hat{a})}{a}$, SE = $\sqrt{\text{Var}(\hat{a})}$. We consider the estimate is accurate if the cv is below some threshold $\rho_0$ (e.g., $\rho_0 = 0.1$). The cv can be expressed as

$$\text{cv} = \frac{\text{SE}(\hat{a})}{a} \approx \frac{1}{a}\sqrt{\frac{\max\left(\frac{f_1}{k_1}, \frac{f_2}{k_2}\right) - 1}{\frac{1}{a} + \frac{1}{f_1 - a} + \frac{1}{f_2 - a} + \frac{1}{D - f_1 - f_2 + a}}} \tag{41}$$

Figure 18(a) plots the required sampling rate $\min\left(\frac{k_1}{f_1}, \frac{k_2}{f_2}\right)$ computed from (41). The figure shows that at Web scale (i.e., $D \approx 10$ billion), a sampling rate as low as $10^{-3}$ may suffice for "ordinary" words (i.e., $f_1 \approx 10^7 = 0.001D$). Figure 18(b) plots the required sample size $k_1$, for the same experiment in Figure 18(a), where for simplicity, we assume $\frac{k_1}{f_1} = \frac{k_2}{f_2}$. The figure shows that, after $D$ is large enough, the required sample size does not increase as much.

To apply (41) to the real data, Table 5 presents the critical sampling rates and sample sizes for all pair-wise combinations of the four-word query *Governor, Schwarzenegger, Terminator, Austria.* Here we assume the estimates in Table 3 are exact. The table verifies that only a very small sample may suffice to achieve a reasonable cv.

### 5.10 Tail Bound and Multiple Comparisons Effect

To choose the sample size, it is often necessary to consider the effect of multiple comparisons. For example, when we estimate all pair-wise associations among $n$ data points,

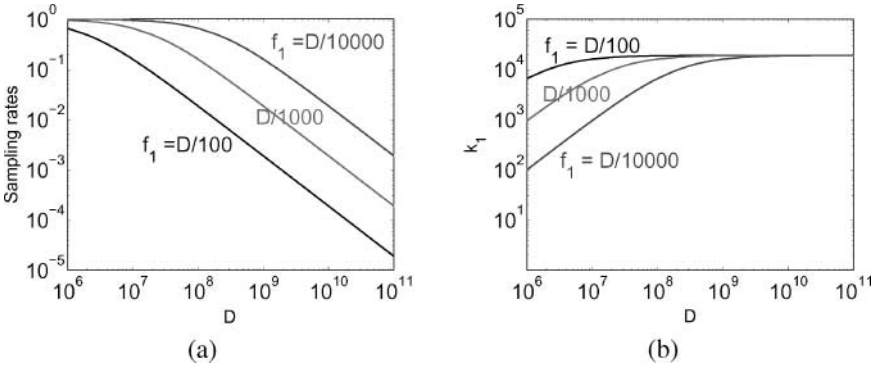



**Figure 18**
(a) An analysis based on cv = $\frac{\text{SE}}{a}$ = 0.1 suggests that we can get away with very low sampling rates. The three curves plot the critical value for the sampling rate, min $\left(\frac{k_1}{f_1}, \frac{k_2}{f_2}\right)$, as a function of corpus size, $D$. At Web scale, $D \approx 10^{10}$, sampling rates above $10^{-2}$ to $10^{-4}$ satisfy cv $\leq 0.1$, at least for these settings of $f_1, f_2$, and $a$. The settings were chosen to simulate "ordinary" words. The three curves correspond to three choices of $f_1$: $D/100$, $D/1000$, and $D/10,000$. $f_2 = f_1/10$, $a = f_2/20$. (b) The critical sample size $k_1$ (assuming $\frac{k_1}{f_1} = \frac{k_2}{f_2}$), corresponding to the sampling rates in (a).

**Table 5**
The critical sampling rates and sample sizes (for cv = 0.1) are computed for all two-way combinations among the four words *Governor, Schwarzenegger, Terminator, Austria,* assuming the estimated document frequencies and two-way associations in Table 3 are exact. The required sampling rates are all very small, verifying our claim that for "ordinary" words, a sampling rate as low as $10^{-3}$ may suffice. In these computations, we used $D = 5 \times 10^9$ for the number of English documents in the collection.

| Query | Critical Sampling Rate |
|---|---|
| Governor, Schwarzenegger | $5.6 \times 10^{-5}$ |
| Governor, Terminator | $7.2 \times 10^{-4}$ |
| Governor, Austria | $1.4 \times 10^{-4}$ |
| Schwarzenegger, Terminator | $1.5 \times 10^{-4}$ |
| Schwarzenegger, Austria | $8.1 \times 10^{-4}$ |
| Terminator, Austria | $5.5 \times 10^{-4}$ |

we are estimating $\frac{n(n-1)}{2}$ pairs simultaneously. A convenient approach is to bound the tail probability

$$\mathbf{Pr}\left(|\hat{a}_{MLE} - a| > \epsilon a\right) \leq \delta/p \tag{42}$$

where $\delta$ (e.g., 0.05) is the level of significance, $\epsilon$ is the specified accuracy (e.g., $\epsilon < 0.5$), and $p$ is the correction factor for multiple comparisons. The most conservative choice is $p = \frac{n^2}{2}$, known as the Bonferroni Correction. But often it is reasonable to let $p$ be much smaller (e.g., $p$ = 100).

We can gain some insight from (42). In particular, our previous argument based on coefficient of variations (cv) is closely related to (42).

Assuming $\hat{a}_{MLE} \sim N(a, \mathrm{Var}(\hat{a}_{MLE}))$, then, based on the known normal tail bound,

$$\mathbf{Pr}\left(|\hat{a}_{MLE} - a| > \epsilon a\right) \leq 2\exp\left(-\frac{\epsilon^2 a^2}{2\mathrm{Var}\left(\hat{a}_{MLE}\right)}\right) = 2\exp\left(-\frac{\epsilon^2}{2\mathrm{cv}^2}\right) \tag{43}$$

combined with (42), leads to the following criterion on cv

$$\mathrm{cv} \geq \epsilon\sqrt{-\frac{1}{2\log\left(\delta/2p\right)}} \tag{44}$$

For example, if we let $\delta = 0.05$, $p = 100$, and $\epsilon = 0.4$, then (44) will output $\mathrm{cv} \approx 0.1$.

### 5.11 Sample Size Selection Based on Storage Constraints

Suppose we can compute the maximum allowed total samples, $T$, for example, based on the available memory. That is, $\sum_{i=1}^{n} k_i = T$, where $n$ is the total number of words. We could allocate $T$ according to document frequencies $f_j$, that is,

$$k_j = \frac{f_j}{\sum_{i=1}^{n} f_i} T \tag{45}$$

Usually, we will need to define a lower bound $k_l$ and an upper bound $k_u$, which have to be selected from engineering experience, depending on the specific applications. We will truncate the computed $k_j$ if it is outside $[k_l,\ k_u]$. Equation (45) implies a uniform corpus sampling rate, which may not be always desirable, but the confinement by $[k_l,\ k_u]$ can effectively vary the sampling rates.

More carefully, we can minimize the total number of "unused" samples. For a pair, $\mathrm{W}_i$ and $\mathrm{W}_j$, if $\frac{k_i}{f_i} \geq \frac{k_j}{f_j}$, then on average, there are $\left(\frac{k_i}{f_i} - \frac{k_j}{f_j}\right) f_i$ samples unused in $\mathrm{K}_i$. This is the basic idea behind the following linear program for choosing the "optimal" sample sizes:

$$\begin{aligned} \text{Minimize} \quad & \sum_{i=1}^{n}\sum_{j=i+1}^{n}\left[f_i\left(\frac{k_i}{f_i} - \frac{k_j}{f_j}\right)_+ + f_j\left(\frac{k_j}{f_j} - \frac{k_i}{f_i}\right)_+\right] \\ \text{subject to} \quad & \sum_{i=1}^{n} k_i = T, \qquad k_i \leq f_i, \qquad k_l \leq k_i \leq k_u \end{aligned} \tag{46}$$

where $(z)_+ = \max(0, z)$, is the positive part of $z$. This program can be modified (possibly no longer a linear program) to consider other factors in different applications. For example, some applications may care more about the very rare words, so we would weight the rare words more.

### 5.12 When Will Sketches Not Perform Well?

We consider three scenarios. (A) $f_1$ and $f_2$ are both large; (B) $f_1$ and $f_2$ are both small; (C) $f_1$ is very large but $f_2$ is very small. Conventional sampling over documents can handle situation (A), but will perform poorly on (B) because there is a good chance that the sample will miss the rare words. The sketch algorithm can handle both (A) and (B) well.

In fact, it will do very well when both words are rare because the equivalent sampling rate $\frac{D_s}{D} \approx \min\left(\frac{k_1}{f_1}, \frac{k_2}{f_2}\right)$ can be high, even 100%.

When $f_2 \ll f_1$, no sampling method can work well unless we are willing to sample $P_1$ with a sufficiently large sample. Otherwise even if we let $\frac{k_2}{f_2} = 100\%$, the corpus sampling rate, $\frac{D_s}{D} \approx \frac{k_1}{f_1}$, will be low. For example, Google estimates 14,000,000 hits for *Holmes*, 37,500 hits for *Diaconis*, and 892 joint hits. Assuming $D = 5 \times 10^9$ and cv = 0.1, the critical sample size for *Holmes* would have to be $1.4 \times 10^6$, probably too large as a sample.[7]

## 6. Extension to Multi-Way Associations

Many applications involve multi-way associations, for example, association rules, databases, and Web search. The "Governator" example in Table 3, for example, made use of both two-way and three-way associations. Fortunately, our sketch construction and estimation algorithm can be naturally extended to multi-way associations. We have already presented an example of estimating multi-way associations in Section 1.6. When we do not consider the margins, the estimation task is as simple as in the pair-wise case. When we do take advantage of margins, estimating multi-way associations amounts to a convex program. We will also analyze the theoretical variances.

### 6.1 Multi-Way Sketches

Suppose we are interested in the associations among $m$ words, denoted by $W_1$, $W_2, \ldots, W_m$. The document frequencies are $f_1, f_2, \ldots,$ and $f_m$, which are also the lengths of the postings $P_1, P_2, \ldots, P_m$. There are $N = 2^m$ combinations of associations, denoted by $x_1, x_2, \ldots, x_N$. For example,

$$
\begin{aligned}
a = x_1 &= |P_1 \cap P_2 \cap \ldots \cap P_{m-1} \cap P_m| \\
x_2 &= |P_1 \cap P_2 \cap \ldots \cap P_{m-1} \cap \neg P_m| \\
x_3 &= |P_1 \cap P_2 \cap \ldots \cap \neg P_{m-1} \cap P_m| \\
&\ldots \\
x_{N-1} &= |\neg P_1 \cap \neg P_2 \cap \ldots \cap \neg P_{m-1} \cap P_m| \\
x_N &= |\neg P_1 \cap \neg P_2 \cap \ldots \cap \neg P_{m-1} \cap \neg P_m|
\end{aligned} \tag{47}
$$

which can be directly corresponded to the binary representation of integers.

Using the vector and matrix notation, $\mathbf{X} = [x_1, x_2, \ldots, x_N]^T$, $\mathbf{F} = [f_1, f_2, \ldots, f_m, D]^T$, where the superscript "T" stands for "transpose", that is, we always work with column vectors. We can write down the margin constraints in terms of a linear matrix equation as

$$\mathbf{AX} = \mathbf{F} \tag{48}$$

7 Readers familiar with random projections can verify that in this case we need $k = 6.6 \times 10^7$ projections in order to achieve cv = 0.1. See Li, Hastie, and Church (2006a, 2006b) for the variance formula of random projections.

where $\mathbf{A}$ is the constraint matrix. If necessary, we can use $\mathbf{A}^{(m)}$ to identify $\mathbf{A}$ for different $m$ values. For example, when $m = 2$ or $m = 3$,

$$\mathbf{A}^{(2)} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \qquad \mathbf{A}^{(3)} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \tag{49}$$

For each word $W_i$, we sample the $k_i$ smallest elements from its permuted postings, $\pi(P_i)$, to form a sketch, $K_i$. Recall $\pi$ is a random permutation on $\Omega = \{1, 2, \ldots, D\}$. We compute

$$D_s = \min\{\max(K_1), \max(K_2), \ldots, \max(K_m)\}. \tag{50}$$

After removing the elements in all $m$ $K_i$'s that are larger than $D_s$, we intersect these $m$ trimmed sketches to generate the sample table counts. The samples are denoted as $\mathbf{S} = [s_1, s_2, \ldots, s_N]^T$.

Conditional on $D_s$, the samples $\mathbf{S}$ are statistically equivalent to $D_s$ random samples over documents from the corpus. The corresponding conditional PMF and log PMF would be

$$\mathbf{Pr}(\mathbf{S}|D_s; \mathbf{X}) = \frac{\binom{x_1}{s_1}\binom{x_2}{s_2}\cdots\binom{x_N}{s_N}}{\binom{D}{D_s}} \propto \prod_{i=1}^{N}\prod_{j=0}^{s_i-1}(x_i - j) \tag{51}$$

$$\log \mathbf{Pr}(\mathbf{S}|D_s; \mathbf{X}) \propto Q = \sum_{i=1}^{N}\sum_{j=0}^{s_i-1}\log(x_i - j) \tag{52}$$

The log PMF is concave, as in two-way associations. A partial likelihood MLE solution, namely, the $\hat{\mathbf{X}}$ that maximizes $\log \mathbf{Pr}(\mathbf{S}|D_s; \hat{\mathbf{X}})$, will again be adopted, which leads to a convex optimization problem. But first, we shall discuss two baseline estimators.

### 6.2 Baseline Independence Estimator

Assuming independence, an estimator of $x_1$ would be

$$\hat{x}_{1,IND} = D\prod_{i=1}^{m}\frac{f_i}{D} \tag{53}$$

which can be easily proved using a conditional expectation argument.

By the property of the hypergeometric distribution, $E(|P_i \cap P_j|) = \frac{f_i f_j}{D}$. Therefore,

$$\begin{aligned} E(x_1) &= E(|P_1 \cap P_2 \cap \ldots \cap P_m|) = E(|\cap_{i=1}^{m} P_i|) \\ &= E(E(|P_1 \cap (\cap_{i=2}^{m} P_i)||(\cap_{i=2}^{m} P_i))) = \frac{f_1}{D}E(|\cap_{i=2}^{m} P_i|) \\ &= \frac{f_1 f_2 \ldots f_{m-2}}{D^{m-2}}E(|P_{m-1} \cap P_m|) = D\prod_{i=1}^{m}\frac{f_i}{D} \end{aligned} \tag{54}$$

### 6.3 Baseline Margin-Free Estimator

The conditional PMF $\mathbf{Pr}(\mathbf{S}|D_s;\mathbf{X})$ is a multivariate hypergeometric distribution, based on which we can derive the margin-free estimator:

$$\mathrm{E}(s_i|D_s) = \frac{D_s}{D}x_i, \quad \hat{x}_{i,MF} = \frac{D}{D_s}s_i, \quad \mathrm{Var}(\hat{x}_{i,MF}|D_s) = \frac{D}{D_s}\frac{1}{\frac{1}{x_i}+\frac{1}{D-x_i}}\frac{D-D_s}{D-1} \tag{55}$$

We can see that the margin-free estimator remains its simplicity in the multi-way case.

### 6.4 The MLE

The exact MLE can be formulated as a standard convex optimization problem,

$$\begin{aligned} &\text{minimize} \quad -Q = -\sum_{i=1}^{N}\sum_{j=0}^{s_i-1}\log(x_i - j) \\ &\text{subject to} \quad \mathbf{AX} = \mathbf{F}, \ \text{and} \ \mathbf{X} \succeq \mathbf{S} \end{aligned} \tag{56}$$

where $\mathbf{X} \succeq \mathbf{S}$ is a compact representation for $x_i \geq s_i, 1 \leq i \leq N$.

This optimization problem can be solved by a variety of standard methods such as Newton's method (Boyd and Vandenberghe 2004, Chapter 10.2). Note that we can ignore the implicit inequality constraints, $\mathbf{X} \succeq \mathbf{S}$, if we start with a feasible initial guess.

It turns out that the formulation in (56) will encounter numerical difficulty due to the inner summation in the objective function $Q$. Smoothing will bring in more numerical issues. Recall that in estimating two-way associations we do not have this problem, because we have eliminated the summation in the objective function, using an (integer) updating formula. In multi-way associations, it seems not easy to reformulate the objective function $Q$ in a similar form.

To avoid the numerical problems, a simple solution is to assume "sample-with-replacement," under which the conditional likelihood and log likelihood become

$$\mathbf{Pr}(\mathbf{S}|D_s;\mathbf{X},r) \propto \prod_{i=1}^{N}\left(\frac{x_i}{D}\right)^{s_i} \propto \prod_{i=1}^{N}x_i^{s_i} \tag{57}$$

$$\log \mathbf{Pr}(\mathbf{S}|D_s;\mathbf{X},r) \propto Q_r = \sum_{i=1}^{N}s_i \log x_i \tag{58}$$

Our MLE problem can then be reformulated as

$$\begin{aligned} &\text{minimize} \quad -Q = -\sum_{i=1}^{N}s_i \log x_i \\ &\text{subject to} \quad \mathbf{AX} = \mathbf{F}, \ \text{and} \ \mathbf{X} \succeq \mathbf{S} \end{aligned} \tag{59}$$

which is again a convex program. To simplify the notation, we neglect the subscript "r."

We can compute the gradient ($\bigtriangledown Q$) and Hessian ($\bigtriangledown^2 Q$). The gradient is a vector of the first derivatives of $Q$ with respect to $x_i$, for $1 \leq i \leq N$,

$$\bigtriangledown Q = \left[\frac{\partial Q}{\partial x_i}, 1 \leq i \leq N\right] = \left[\frac{s_1}{x_1}, \frac{s_2}{x_2}, \ldots, \frac{s_N}{x_N}\right]^{\mathrm{T}} \tag{60}$$

The Hessian is a matrix whose $(i,j)^{th}$ entry is the partial derivative $\frac{\partial^2 Q}{\partial x_i x_j}$, that is,

$$\bigtriangledown^2 Q = -\mathrm{diag}\left[\frac{s_1}{x_1^2}, \frac{s_2}{x_2^2}, \ldots, \frac{s_N}{x_N^2}\right] \tag{61}$$

The Hessian has a very simple diagonal form, implying that Newton's method will be a good algorithm for solving this optimization problem. We implement, in Appendix 2, the equality constrained Newton's method with feasible start and backtracking line search (Boyd and Vandenberghe 2004, Algorithm 10.1). A key step is to solve for Newton's step, $\triangle \mathbf{X}_{nt}$:

$$\begin{bmatrix} -\bigtriangledown^2 Q & \mathbf{A}^{\mathrm{T}} \\ \mathbf{A} & 0 \end{bmatrix} \begin{bmatrix} \triangle \mathbf{X}_{nt} \\ \text{dummy} \end{bmatrix} = \begin{bmatrix} \bigtriangledown Q \\ 0 \end{bmatrix}. \tag{62}$$

Because the Hessian $\bigtriangledown^2 Q$ is a diagonal matrix, solving for Newton's step in (62) can be sped up substantially (e.g., using the block matrix inverse formula).

### 6.5 The Covariance Matrix

We apply the large sample theory to estimate the covariance matrix of the MLE. Recall that we have $N = 2^m$ variables and $m+1$ constraints. The effective number of variables would be $2^m - (m+1)$, which is also the dimension of the covariance matrix.

We seek a partition of $\mathbf{A} = [\mathbf{A_1}, \mathbf{A_2}]$, such that $\mathbf{A_2}$ is invertible. We may have to switch some columns of $\mathbf{A}$ in order to find an invertible $\mathbf{A_2}$. In our construction, the $j$th column of $\mathbf{A}_2$ is the column of $\mathbf{A}$ such that last entry of the $j$th row of $\mathbf{A}$ is 1. An example for $m = 3$ would be

$$\mathbf{A_1^{(3)}} = \begin{bmatrix} 1&1&1&0 \\ 1&1&0&1 \\ 1&0&1&1 \\ 1&1&1&1 \end{bmatrix} \quad \mathbf{A_2^{(3)}} = \begin{bmatrix} 1&0&0&0 \\ 0&1&0&0 \\ 0&0&1&0 \\ 1&1&1&1 \end{bmatrix} \tag{63}$$

where $\mathbf{A_1^{(3)}}$ is the [1 2 3 5] columns of $\mathbf{A^{(3)}}$ and $\mathbf{A_2^{(3)}}$ is the [4 6 7 8] columns of $\mathbf{A^{(3)}}$. We can see that $\mathbf{A}_2$ constructed this way is always invertible because its determinant is always one.

Corresponding to the partition of $\mathbf{A}$, we partition $\mathbf{X} = [\mathbf{X}_1, \mathbf{X}_2]^{\mathrm{T}}$. For example, when $m = 3$, $\mathbf{X}_1 = [x_1, x_2, x_3, x_5]^{\mathrm{T}}$, $\mathbf{X}_2 = [x_4, x_6, x_7, x_8]^{\mathrm{T}}$. We can then express $\mathbf{X}_2$ to be

$$\mathbf{X}_2 = \mathbf{A}_2^{-1}(\mathbf{F} - \mathbf{A}_1\mathbf{X_1}) = \mathbf{A}_2^{-1}\mathbf{F} - \mathbf{A}_2^{-1}\mathbf{A}_1\mathbf{X_1} \tag{64}$$

The log likelihood function $Q$, which is separable, can then be expressed as

$$Q(\mathbf{X}) = Q_1(\mathbf{X}_1) + Q_2(\mathbf{X}_2) \tag{65}$$

By the matrix derivative chain rule, the Hessian of $Q$ with respect to $\mathbf{X}_1$ would be

$$\bigtriangledown_1^2 Q = \bigtriangledown_1^2 Q_1 + \bigtriangledown_1^2 Q_2 = \bigtriangledown_1^2 Q_1 + \left(\mathbf{A}_2^{-1}\mathbf{A}_1\right)^{\mathrm{T}} \bigtriangledown_2^2 Q_2 \left(\mathbf{A}_2^{-1}\mathbf{A}_1\right) \tag{66}$$

where we use $\bigtriangledown_1^2$ and $\bigtriangledown_2^2$ to indicate the Hessians are with respect to $\mathbf{X}_1$ and $\mathbf{X}_2$, respectively.

Conditional on $D_s$, the Expected Fisher Information of $\mathbf{X}_1$ is

$$\mathrm{I}(\mathbf{X}_1) = \mathrm{E}\left(-\bigtriangledown_1^2 Q | D_s\right) = -\mathrm{E}(\bigtriangledown_1^2 Q_1 | D_s) - \left(\mathbf{A}_2^{-1}\mathbf{A}_1\right)^{\mathrm{T}} \mathrm{E}(\bigtriangledown_2^2 Q_2 | D_s) \left(\mathbf{A}_2^{-1}\mathbf{A}_1\right) \tag{67}$$

where

$$\mathrm{E}(-\bigtriangledown_1^2 Q_1 | D_s) = \mathrm{diag}\left[\mathrm{E}\left(\frac{s_i}{x_i^2}\right), x_i \in \mathbf{X}_1\right] = \frac{D_s}{D}\mathrm{diag}\left[\frac{1}{x_i}, x_i \in \mathbf{X}_1\right] \tag{68}$$

$$\mathrm{E}(-\bigtriangledown_2^2 Q_2 | D_s) = \frac{D_s}{D}\mathrm{diag}\left[\frac{1}{x_i}, x_i \in \mathbf{X}_2\right] \tag{69}$$

By the large sample theory, and also considering the finite population correction factor, we can approximate the (conditional) covariance matrix of $\mathbf{X}_1$ to be

$$\begin{aligned}\mathrm{Cov}(\mathbf{X}_1 | D_s) &\approx \mathrm{I}(\mathbf{X}_1)^{-1}\left(1 - \frac{D_s}{D}\right) \\ &= \left(\frac{D}{D_s} - 1\right)\left(\mathrm{diag}\left[\frac{1}{x_i}, x_i \in \mathbf{X}_1\right] + \left(\mathbf{A}_2^{-1}\mathbf{A}_1\right)^{\mathrm{T}} \mathrm{diag}\left[\frac{1}{x_i}, x_i \in \mathbf{X}_2\right]\left(\mathbf{A}_2^{-1}\mathbf{A}_1\right)\right)^{-1}\end{aligned} \tag{70}$$

For a sanity check, we verify that this approach recovers the same variance formula in the two-way association case. Recall that, when $m = 2$, we have

$$\bigtriangledown^2 Q = -\begin{bmatrix} \frac{s_1}{x_1^2} & 0 & 0 & 0 \\ 0 & \frac{s_2}{x_2^2} & 0 & 0 \\ 0 & 0 & \frac{s_3}{x_3^2} & 0 \\ 0 & 0 & 0 & \frac{s_4}{x_4^2} \end{bmatrix}, \ \bigtriangledown_1^2 Q_1 = -\frac{s_1}{x_1^2}, \ \bigtriangledown_2^2 Q_2 = -\begin{bmatrix} \frac{s_2}{x_2^2} & 0 & 0 \\ 0 & \frac{s_3}{x_3^2} & 0 \\ 0 & 0 & \frac{s_4}{x_4^2} \end{bmatrix} \tag{71}$$

$$\mathbf{A}^{(2)} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}, \ \mathbf{A}_1^{(2)} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \ \mathbf{A}_2^{(2)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} \tag{72}$$

$$\left(\mathbf{A}_2^{-1}\mathbf{A}_1\right)^{\mathrm{T}} \bigtriangledown_2^2 Q_2 \mathbf{A}_2^{-1}\mathbf{A}_1 = -\begin{bmatrix}1 & 1 & -1\end{bmatrix}\begin{bmatrix}\frac{s_2}{x_2^2} & 0 & 0\\ 0 & \frac{s_3}{x_3^2} & 0\\ 0 & 0 & \frac{s_4}{x_4^2}\end{bmatrix}\begin{bmatrix}1\\1\\-1\end{bmatrix} = -\frac{s_2}{x_2^2} - \frac{s_3}{x_3^2} - \frac{s_4}{x_4^2} \quad (73)$$

Hence,

$$-\bigtriangledown_1^2 Q = \frac{s_1}{x_1^2} + \frac{s_2}{x_2^2} + \frac{s_3}{x_3^2} + \frac{s_4}{x_4^2} = \frac{a_s}{a^2} + \frac{b_s}{(f_1-a)^2} + \frac{c_s}{(f_2-a)^2} + \frac{d_s}{(D-f_1-f_2+a)^2} \quad (74)$$

which leads to the same Fisher Information for the two-way association as we have derived.

### 6.6 The Unconditional Covariance Matrix

Similar to two-way associations, the unconditional variance of the proposed MLE can be estimated by replacing $\frac{D}{D_s}$ in (70) with $\mathrm{E}\left(\frac{D}{D_s}\right)$, namely,

$$\mathrm{Cov}(\mathbf{X}_1) \approx \left(\mathrm{E}\left(\frac{D}{D_s}\right) - 1\right) \times \left(\mathrm{diag}\left[\frac{1}{x_i}, x_i \in \mathbf{X}_1\right] + \left(\mathbf{A}_2^{-1}\mathbf{A}_1\right)^{\mathrm{T}} \mathrm{diag}\left[\frac{1}{x_i}, x_i \in \mathbf{X}_2\right]\left(\mathbf{A}_2^{-1}\mathbf{A}_1\right)\right)^{-1} \quad (75)$$

Similar to two-way associations, we recommend the following approximations:

$$\mathrm{E}\left(\frac{D_s}{D}\right) \approx \min\left(\frac{k_1}{f_1}, \frac{k_2}{f_2}, \ldots, \frac{k_m}{f_m}\right) \quad (76)$$

$$\mathrm{E}\left(\frac{D}{D_s}\right) \approx \max\left(\frac{f_1}{k_1}, \frac{f_2}{k_2}, \ldots, \frac{f_m}{k_m}\right) \quad (77)$$

Again, the approximation (76) will overestimate $\mathrm{E}\left(\frac{D_s}{D}\right)$ and (77) will underestimate $\mathrm{E}\left(\frac{D}{D_s}\right)$ hence also underestimating the unconditional variance.

### 6.7 Empirical Evaluation

We use the same four words as in Table 4 to evaluate the multi-way association algorithm, as merely a sanity check. There are four different combinations of three-way associations and one four-way association, as listed in Table 6.

We present results for $x_1$ (i.e., $a$ in two-way associations) for all cases. The evaluations for four three-way cases are presented in Figures 19, 20 and 21. From these figures, we see that the proposed MLE has lower MSE than the MF. As in the two-way case, smoothing helps MLE but still hurts MF in most cases. Also, the experiments verify that our approximate variance formulas are fairly accurate.

Figure 22 presents the evaluation results for the four-way association case, including MSE, smoothing, and variance. The results are similar to the three-way case.

**Table 6**
The same four words as in Table 4 are used for evaluating multi-way associations. There are in total four three-way combinations and one four-way combination.

| | Case No. | Words | Co-occurrences |
|---|---|---|---|
| Three-way | Case 3-1 | THIS, HAVE, HELP | 4940 |
| | Case 3-2 | THIS, HAVE, PROGRAM | 2575 |
| | Case 3-3 | THIS, HELP, PROGRAM | 1626 |
| | Case 3-4 | HAVE, HELP, PROGRAM | 1460 |
| Four-way | Case 4 | THIS, HAVE, HELP, PROGRAM | 1316 |

We have used the empirical $\mathrm{E}\left(\frac{D}{D_s}\right)$ to compute the unconditional variance. Figure 23 plots $\max\left(\frac{f_1}{k_1}, \frac{f_2}{k_2}, \ldots, \frac{f_m}{k_m}\right) / \frac{D}{D_s}$ for all cases. The figure indicates that using $\max\left(\frac{f_1}{k_1}, \frac{f_2}{k_2}, \ldots, \frac{f_m}{k_m}\right)$ to estimate $\mathrm{E}\left(\frac{D}{D_s}\right)$ is still fairly accurate when the sample size is reasonable.

Combining the results of two-way associations for the same four words, we can study the trend how the proposed MLE improve the MF baseline. Figure 24(a) sug-

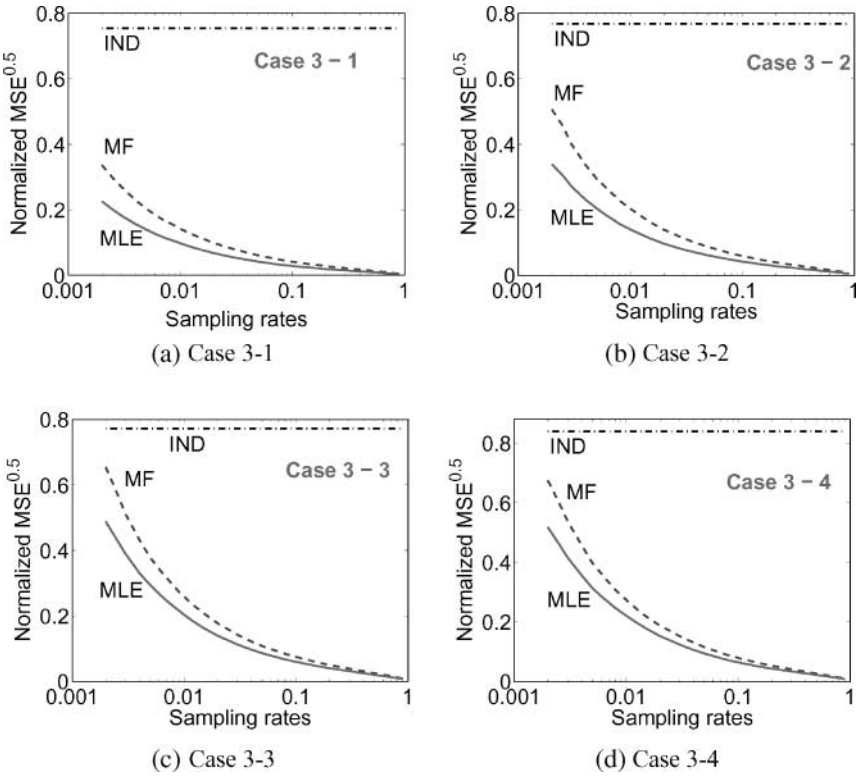



**Figure 19**
In terms of $\frac{\sqrt{\mathrm{MSE}(x_1)}}{x_1}$, the proposed MLE is consistently better than the MF, which is better than the IND, for four three-way association cases.

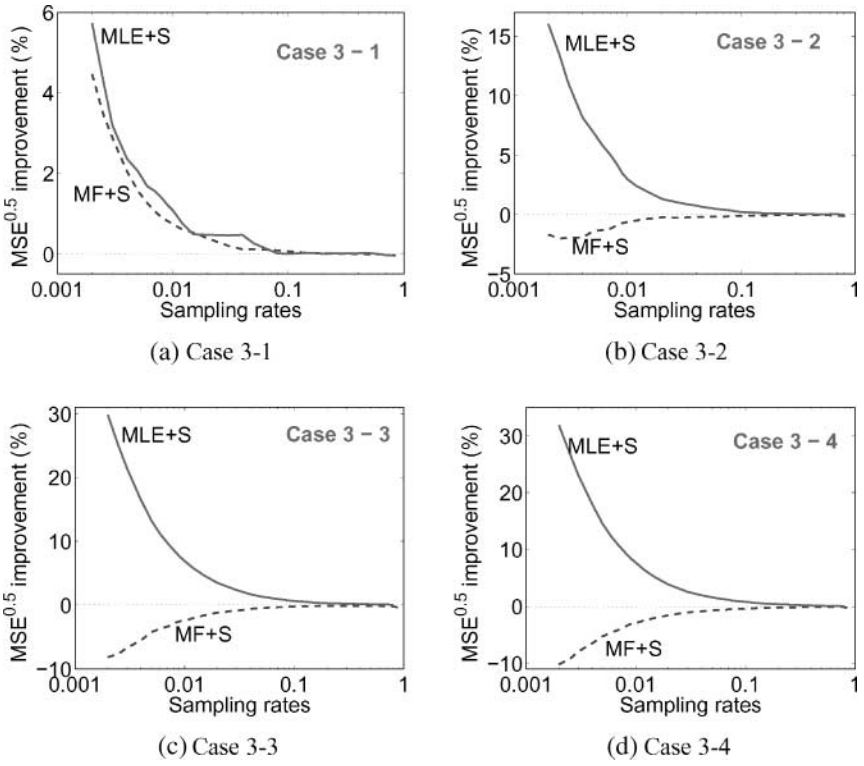



**Figure 20**
The simple "add-one" smoothing improves the estimation accuracies for the proposed MLE. Smoothing, however, in all cases except Case 3-1 hurts the margin-free estimator.

gests that the proposed MLE is a big improvement over the MF baseline for two-way associations, but the improvement becomes less and less noticeable with higher order associations. This observation is not surprising, because the number of degrees of freedom, $2^m - (m+1)$, increases exponentially with $m$. In order words, the margin constraints are most effective for small $m$, but the effectiveness decreases rapidly with $m$.

On the other hand, smoothing becomes more and more important as $m$ increases, as shown in Figure 24(b), partly because of the data sparsity in high order associations.

## 7. Related Work: Comparison with Broder's Sketches

Broder's sketches (Broder 1997), originally introduced for removing duplicates in the AltaVista index, have been applied to a variety of applications (Broder et al. 1997; Haveliwala, Gionis, and Indyk 2000; Haveliwala et al. 2002). Broder et al. (1998, 2000) presented some theoretical aspects of the sketch algorithm. There has been considerable exciting work following up on this line of research including Indyk (2001), Charikar (2002), and Itoh, Takei, and Tarui (2003).

Broder and his colleagues introduced two algorithms, which we will refer to as the "original sketch" and the "minwise sketch" for estimating resemblance, $R = \frac{|P_1 \cap P_2|}{|P_1 \cup P_2|}$. The original sketch uses a single random permutation on $\Omega = \{1, 2, 3, \ldots, D\}$, and the minwise sketch uses $k$ random permutations. Both algorithms have similar estimation accuracies, as will see.

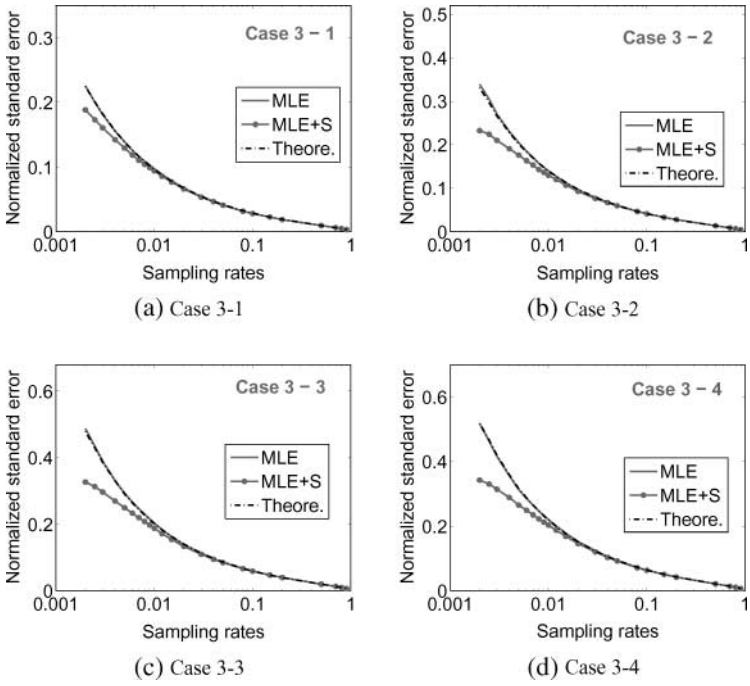



**Figure 21**
In terms of $\frac{\mathrm{SE}(x_1)}{x_1}$, the theoretical variance of MLE fits the empirical values very well. At low sampling rates, smoothing effectively reduces the variance. Note that we plug in the empirical $\mathrm{E}\left(\frac{D}{D_s}\right)$ into (75) to estimate the unconditional variance. The errors due to this approximation are presented in Figure 23.

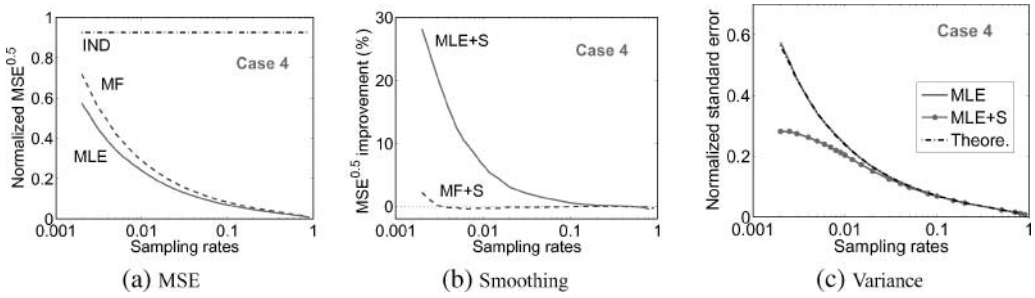



**Figure 22**
Four-way associations (Case 4). (a) The proposed MLE has smaller MSE than the margin-free (MF) baseline, which has smaller MSE than the independence baseline. (b) Smoothing considerably improves the accuracy for MLE and also slightly improves MF. (c) For the proposed MLE, the theoretical prediction fits the empirical variance very well. Smoothing considerably reduces variance.

Our proposed sketch algorithm is closer to Broder's original sketch, with a few important differences. A key difference is that Broder's original sketch throws out half of the sample, whereas we throw out less. In addition, the sketch sizes are fixed over all words for Broder, whereas we allow different sizes for different words. Broder's method was designed for a single statistic (resemblance), whereas we generalize the method to

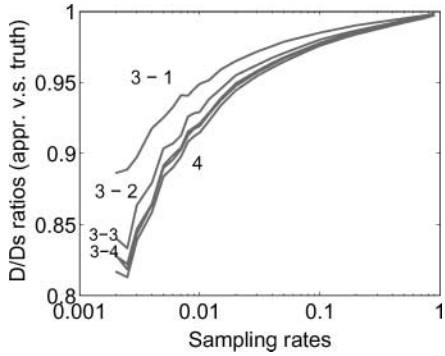



**Figure 23**
The ratios $\max\left(\frac{f_1}{k_1}, \frac{f_2}{k_2}, \ldots, \frac{f_m}{k_m}\right) / \frac{D}{D_s}$ are plotted for all cases. At sampling rates $> 0.01$, the ratios are $> 0.9 - 0.95$, indicating good accuracy.

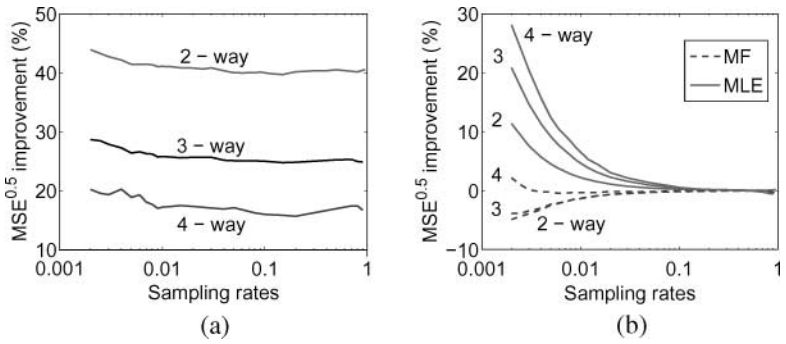



**Figure 24**
(a) Combining the three-way, four-way, and two-way association results for the four words in the evaluations, the average relative improvements of $\sqrt{\text{MSE}}$ suggests that the proposed MLE is consistently better than the MF baseline but the improvement decreases monotonically as the order of associations increases. (b) Average $\sqrt{\text{MSE}}$ improvements due to smoothing imply that smoothing becomes more and more important as the order of association increases.

compute contingency tables (and summaries thereof). Broder's method was designed for pairwise associations, whereas our method generalizes to multi-way associations. Finally, Broder's method was designed for boolean data, whereas our method generalizes to reals.

### 7.1 Broder's Minwise Sketch

Suppose a random permutation $\pi_1$ is performed on the document IDs. We denote the smallest IDs in the postings $P_1$ and $P_2$, by $\min(\pi_1(P_1))$ and $\min(\pi_1(P_2))$, respectively. Obviously,

$$\mathbf{Pr}\left(\min(\pi_1(P_1)) = \min(\pi_1(P_2))\right) = \frac{|P_1 \cap P_2|}{|P_1 \cup P_2|} = R \tag{78}$$

After $k$ minwise independent permutations, denoted as $\pi_1, \pi_2, \ldots, \pi_k$, we can estimate $R$ without bias, as a binomial probability, namely,

$$\hat{R}_{B,r} = \frac{1}{k}\sum_{i=1}^{k}\{\min(\pi_i(\mathrm{P}_1)) = \min(\pi_i(\mathrm{P}_2))\} \qquad \text{and} \qquad \mathrm{Var}\left(\hat{R}_{B,r}\right) = \frac{1}{k}R(1-R) \tag{79}$$

### 7.2 Broder's Original Sketch

A single random permutation $\pi$ is applied to the document IDs. Two sketches are constructed: $\mathrm{K}_1 = \mathrm{MIN}_{k_1}(\pi(\mathrm{P}_1))$, $\mathrm{K}_2 = \mathrm{MIN}_{k_2}(\pi(\mathrm{P}_2))$.[8] Broder (1997) proposed an unbiased estimator for the resemblance:

$$\hat{R}_B = \frac{|\mathrm{MIN}_k(\mathrm{K}_1 \cup \mathrm{K}_2) \cap \mathrm{K}_1 \cap \mathrm{K}_2|}{|\mathrm{MIN}_k(\mathrm{K}_1 \cup \mathrm{K}_2)|} \tag{80}$$

Note that intersecting by $\mathrm{MIN}_k(\mathrm{K}_1 \cup \mathrm{K}_2)$ throws out half the samples, which can be undesirable (and unnecessary).

The following explanation for (80) is slightly different from Broder (1997). We can divide the set $\mathrm{P}_1 \cup \mathrm{P}_2$ (of size $a + b + c = f_1 + f_2 - a$) into two disjoint sets: $\mathrm{P}_1 \cap \mathrm{P}_2$ and $\mathrm{P}_1 \cup \mathrm{P}_2 - \mathrm{P}_1 \cap \mathrm{P}_2$. Within the set $\mathrm{MIN}_k(\mathrm{K}_1 \cup \mathrm{K}_2)$ (of size $k$), the document IDs that belong to $\mathrm{P}_1 \cap \mathrm{P}_2$ would be $\mathrm{MIN}_k(\mathrm{K}_1 \cup \mathrm{K}_2) \cap \mathrm{K}_1 \cap \mathrm{K}_2$, whose size is denoted by $a_s^{\mathrm{B}}$. This way, we have a hypergeometric sample, that is, we sample $k$ document IDs from $\mathrm{P}_1 \cup \mathrm{P}_2$ randomly without replacement and obtain $a_s^{\mathrm{B}}$ IDs that belong to $\mathrm{P}_1 \cap \mathrm{P}_2$. By the property of the hypergeometric distribution, the expectation of $a_s^{\mathrm{B}}$ would be

$$\mathrm{E}\left(a_s^{\mathrm{B}}\right) = \frac{ak}{f_1 + f_2 - a} \Longrightarrow \mathrm{E}\left(\frac{a_s^{\mathrm{B}}}{k}\right) = \frac{a}{f_1 + f_2 - a} = \frac{|\mathrm{P}_1 \cap \mathrm{P}_2|}{|\mathrm{P}_1 \cup \mathrm{P}_2|} \Longrightarrow \mathrm{E}(\hat{R}_B) = R \tag{81}$$

The variance of $\hat{R}_B$, according to the hypergeometric distribution, is:

$$\mathrm{Var}\left(\hat{R}_B\right) = \frac{1}{k}R(1-R)\frac{f_1 + f_2 - a - k}{f_1 + f_2 - a - 1} \tag{82}$$

where the term $\frac{f_1+f_2-a-k}{f_1+f_2-a-1}$ is the "finite population correction factor."

The minwise sketch can be considered as a "sample-with-replacement" variate of the original sketch. The analysis of minwise sketch is slightly simpler mathematically whereas the original sketch is more efficient. The original sketch requires only one random permutation and has slightly smaller variance than the minwise sketch, that is, $\mathrm{Var}\left(\hat{R}_{B,r}\right) \geq \mathrm{Var}\left(\hat{R}_B\right)$. When $k$ is reasonably small, as is common in practice, two sketch algorithms have similar errors.

### 7.3 Why Our Algorithm Improves Broders's Sketch

Our proposed sketch algorithm starts with Broder's original (one permutation) sketch; but our estimation method differs in two important aspects.

8 Actually, the method required fixing sketch sizes: $k_1 = k_2 = k$, a restriction that we find convenient to relax.

Firstly, Broder's estimator (80) uses $k$ out of $2 \times k$ samples. In particular, it uses only $a_s^B = |\text{MIN}_k(K_1 \cup K_2) \cap K_1 \cap K_2|$ intersections, which is always smaller than $a_s = |K_1 \cap K_2|$ available in the samples. In contrast, our algorithm takes advantage of all useful samples up to $D_s = \min(\max(K_1), \max(K_2))$, particularly all $a_s$ intersections. If $\frac{k_1}{f_1} = \frac{k_2}{f_2}$, that is, if we sample proportionally to the margins:

$$k_1 = 2k\frac{f_1}{f_1+f_2} \qquad k_2 = 2k\frac{f_2}{f_1+f_2} \tag{83}$$

it is expected that almost all samples will be utilized.

Secondly, Broder's estimator (80) considers a two-cell hypergeometric model $(a, b+c)$ whereas the two-way association is a four-cell model $(a, b, c, d)$, which is used in our proposed estimator. Simpler data models often result in simpler estimation methods but with larger errors.

Therefore, it is obvious that our proposed method has smaller estimator errors. Next, we compare our estimator with Broder's sketches in terms of the theoretical variances.

### 7.4 Comparison of Variances

Broder's method was designed to estimate resemblance. Thus, this section will compare the proposed method with Broder's sketches in terms of resemblance, $R$.

We can compute $R$ from our estimated association $\hat{a}_{MLE}$:

$$\hat{R}_{MLE} = \frac{\hat{a}_{MLE}}{f_1+f_2-\hat{a}_{MLE}} \tag{84}$$

$\hat{R}_{MLE}$ is slightly biased. However, because the second derivative $R''(a)$

$$R''(a) = \frac{2(f_1+f_2)}{(f_1+f_2-a)^3} \leq \frac{2(f_1+f_2)}{\max(f_1,f_2)^3} \leq \frac{4}{\max(f_1,f_2)^2} \tag{85}$$

is small (i.e., the nonlinearity is weak), it is unlikely that the bias will be noticeable in practice.

By the delta method as described in Section 5.8, the variance of $\hat{R}_{MLE}$ is approximately

$$\text{Var}\left(\hat{R}_{MLE}\right) \approx \text{Var}(\hat{a}_{MLE})(R'(a))^2 = \frac{\max\left(\frac{f_1}{k_1}, \frac{f_2}{k_2}\right)}{\frac{1}{a}+\frac{1}{f_1-a}+\frac{1}{f_2-a}+\frac{1}{D-f_1-f_2+a}} \frac{(f_1+f_2)^2}{(f_1+f_2-a)^4} \tag{86}$$

conservatively ignoring the "finite population correction factor," for convenience.

Define the ratio of the variances to be $V_B = \frac{\text{Var}\left(\hat{R}_{MLE}\right)}{\text{Var}\left(\hat{R}_B\right)}$, then

$$V_B = \frac{\text{Var}\left(\hat{R}_{MLE}\right)}{\text{Var}\left(\hat{R}_B\right)} = \frac{\max\left(\frac{f_1}{k_1}, \frac{f_2}{k_2}\right)}{\frac{1}{a}+\frac{1}{f_1-a}+\frac{1}{f_2-a}+\frac{1}{D-f_1-f_2+a}} \frac{(f_1+f_2)^2}{(f_1+f_2-a)^2} \frac{k}{a(f_1+f_2-2a)} \tag{87}$$

To help our intuitions, let us consider some reasonable simplifications to $V_B$. Assuming $a << \min(f_1, f_2) < \max(f_1, f_2) << D$, then approximately

$$V_B \approx \frac{k \max(\frac{f_1}{k_1}, \frac{f_2}{k_2})}{f_1 + f_2} = \begin{cases} \frac{\max(f_1, f_2)}{f_1 + f_2} & \text{if } k_1 = k_2 = k \\ \frac{1}{2} & \text{if } k_1 = 2k\frac{f_1}{f_1+f_2}, \quad k_2 = 2k\frac{f_2}{f_1+f_2} \end{cases} \tag{88}$$

which indicates that the proposed method is a considerable improvement over Broder's sketches. In order to achieve the same accuracy, our method requires only half as many samples.

Figure 25 plots the $V_B$ in (87) for the whole range of $f_1$, $f_2$, and $a$, assuming equal samples: $k_1 = k_2 = k$. We can see that $V_B \leq 1$ always holds and $V_B = 1$ only when $f_1 = f_2 = a$. There is also the possibility that $V_B$ is close to zero.

Proportional samples further reduce $V_B$, as shown in Figure 26.

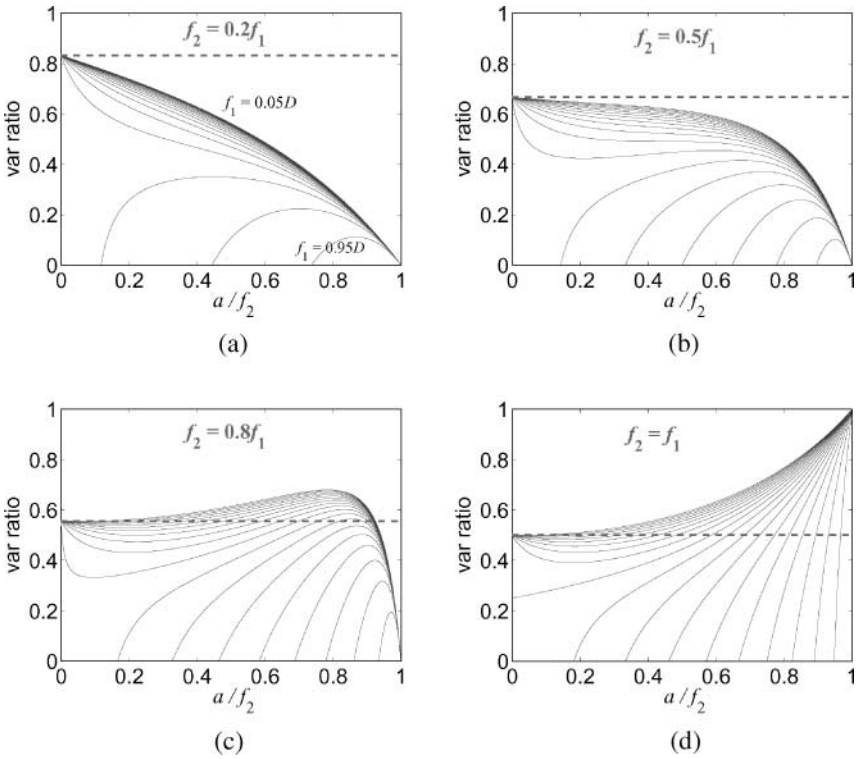



**Figure 25**
We plot $V_B$ in (87) for the whole range of $f_1$, $f_2$, and $a$, assuming equal samples: $k_1 = k_2 = k$. (a), (b), (c), and (d) correspond to $f_2 = 0.2f_1$, $f_2 = 0.5f_1$, $f_2 = 0.8f_1$, and $f_2 = f_1$, respectively. Different curves are for different $f_1$'s, ranging from $0.05D$ to $0.95D$ spaced at $0.05D$. The horizontal lines are $\frac{\max(f_1,f_2)}{f_1+f_2}$. We can see that for all cases, $V_B \leq 1$ holds. $V_B = 1$ when $f_1 = f_2 = a$, a trivial case. When $a/f_2$ is small, $V_B \approx \frac{\max(f_1,f_2)}{f_1+f_2}$ holds well. It is also possible that $V_B$ is very close to zero.

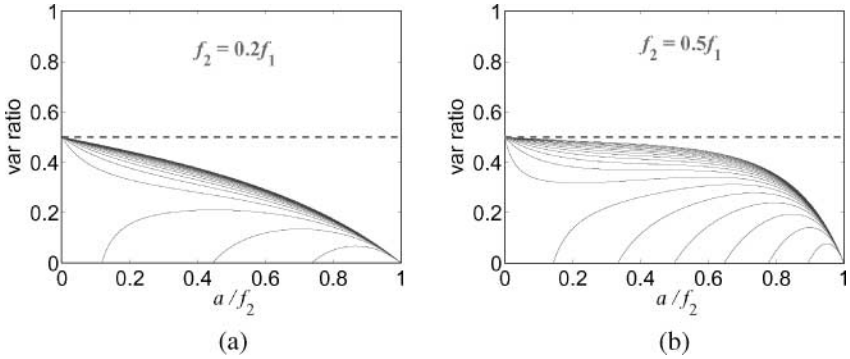



**Figure 26**
Compared with equal samples in Figure 25, proportional samples further reduce $V_B$.

We can show algebraically that $V_B$ in (87) is always less than unity unless $f_1 = f_2 = a$. For convenience, we use the notion $a$, $b$, $c$, $d$ in (87). Assuming $k_1 = k_2 = k$ and $f_1 > f_2$, we obtain

$$V_B = \frac{a+b}{\frac{1}{a}+\frac{1}{b}+\frac{1}{c}+\frac{1}{d}} \frac{(2a+b+c)^2}{(a+b+c)^2} \frac{1}{a(b+c)} \tag{89}$$

To show $V_B \leq 1$, it suffices to show

$$(a+b)(2a+b+c)^2 bcd \leq (bcd+acd+abd+abc)(a+b+c)^2(b+c) \tag{90}$$

which is equivalent to following true statement:

$$(a^3(b-c)^2 + bc^2(b+c)^2 + a^2(2b+c)(b^2-bc+2c^2) + a(b+c)(b^3+4bc^2+c^2))d + abc(b+c)(a+b+c)^2 \geq 0 \tag{91}$$

### 7.5 Empirical Evaluations

We have theoretically shown that our proposed method is a considerable improvement over Broder's sketch. Next, we would like to evaluate these theoretical results using the same experiment data as in evaluating two-way associations (i.e., Table 4).

Figure 27 compares the MSE. Here we assume equal samples and later we will show that proportional samples could further improve the results. The figure shows that our MLE estimator is consistently better than Broder's sketch. In addition, the approximate MLE $\hat{a}_{MLE,a}$ still gives very close answers to the exact MLE, and the simple "add-one" smoothing improves the estimations at low sampling rates, quite substantially.

Figure 28 illustrates the bias. As expected, estimating resemblance from $\hat{a}_{MLE}$ introduces a small bias. This bias will be ignored since it is small compared to the MSE.

Figure 29 verifies that the variance of our estimator is always smaller than Broder's sketch. Our theoretical variance in (86) underestimates the true variances because the approximation $\mathrm{E}\left(\frac{D}{D_s}\right) = \max\left(\frac{f_1}{k_1}, \frac{f_2}{k_2}\right)$ underestimates the variance. In addition, because

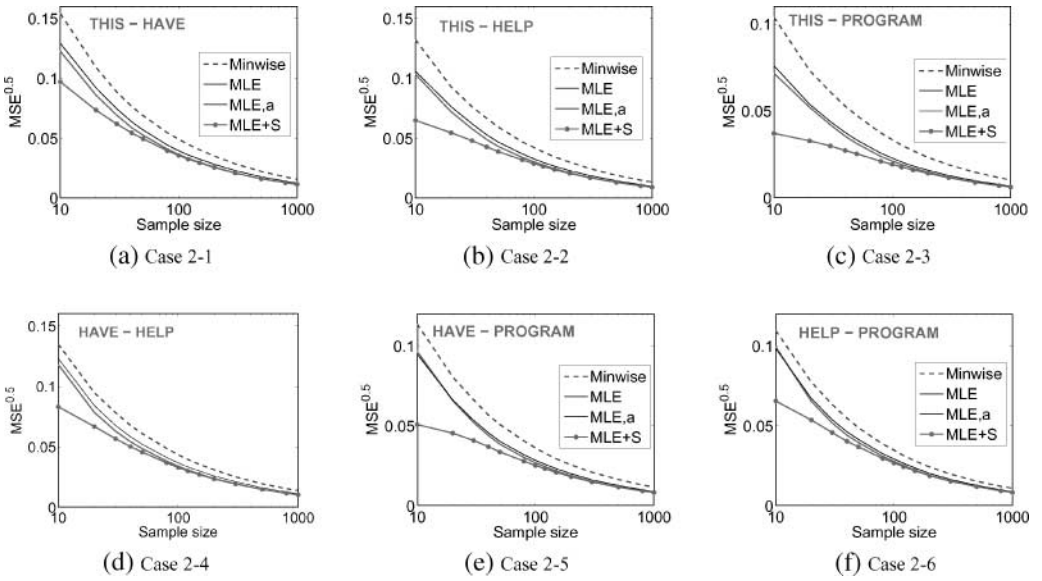



**Figure 27**
When estimating the resemblance, our algorithm gives consistently more accurate answers than Broder's sketch. In our experiments, Broder's "minwise" construction gives almost the same answers as the "original" sketch, thus only the "minwise" results are presented here. The approximate MLE again gives very close answers to the exact MLE. Also, smoothing improves at low sampling rates.

the resemblance $R(a)$ is a convex function of $a$, the delta method also underestimates the variance. However, Figure 29 shows that the errors are not very large, and become negligible with reasonably large sample sizes (e.g., 50). This evidence suggests that the variance formula (86) is reliable.

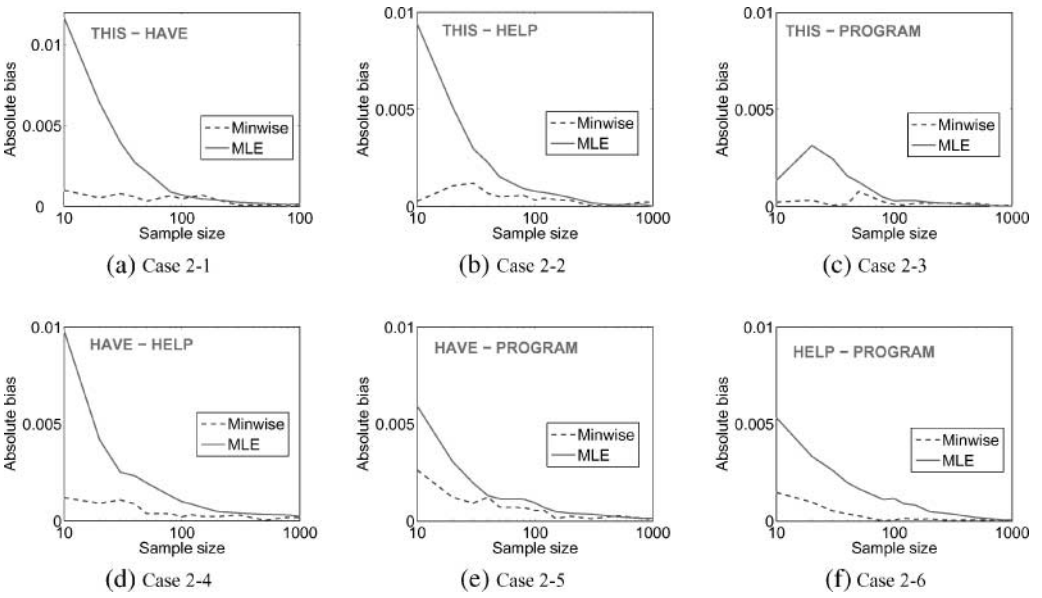



**Figure 28**
Our proposed MLE has higher bias than the "minwise" estimator because of the non-linearity of resemblance. However, the bias is very small compared with the MSE.

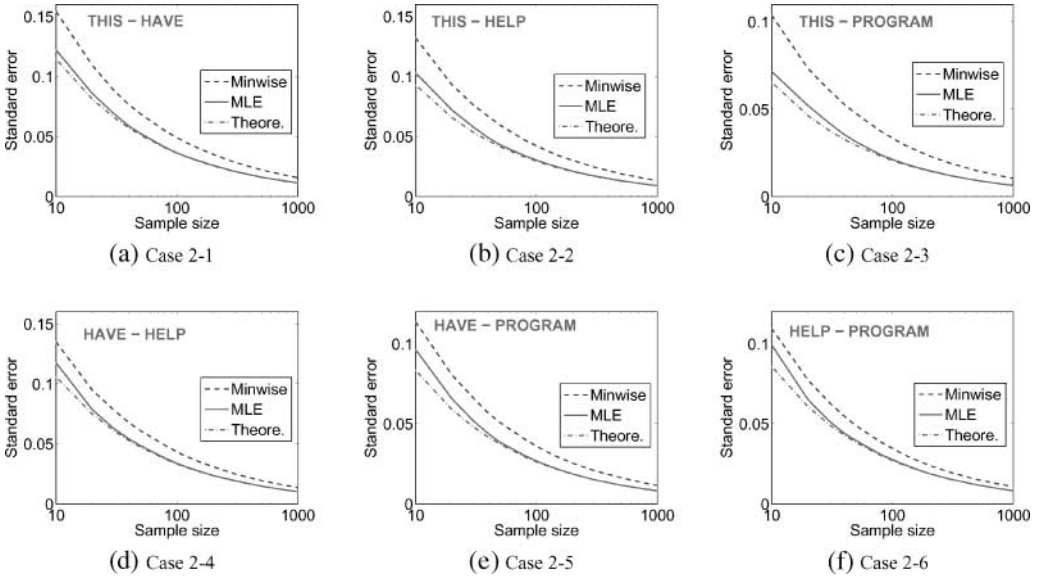



**Figure 29**
Our proposed estimator has consistently smaller variances than Broder's sketch. The theoretical variance, computed by (86), slightly underestimates the true variance with small samples. Here we did not plot the theoretical variance for Broder's sketch because it is very close to the empirical curve.

Finally, in Figure 30, we show that with proportional samples, our algorithm further improves the estimates in terms of MSE. With equal samples, our estimators improve Broder's sketch by 30–50%. With proportional samples, improvements become 40–80%. Note that the maximum possible improvement is 100%.

## 8. Conclusion

In databases, data mining, and information retrieval, there has been considerable interest in sampling and sketching techniques (Chaudhuri, Motwani, and Narasayya 1998; Indyk and Motwani 1998; Manku, Rajagopalan, and Lindsay 1999; Charikar 2002; Achlioptas 2003; Gilbert et al. 2003; Li, Hastie, and Church 2007; Li 2006), which are useful for numerous applications such as association rules (Brin et al. 1997; Brin, Motwani, and Silverstein 1997), clustering (Guha, Rastogi, and Shim 1998; Broder 1998; Aggarwal et al. 1999; Haveliwala, Gionis, and Indyk 2000; Haveliwala et al. 2002), query optimization (Matias, Vitter, and Wang 1998; Chaudhuri, Motwani, and Narasayya 1999), duplicate detection (Broder 1997; Brin, Davis, and Garcia-Molina 1995), and more. Sampling methods become more and more important with larger and larger collections.

The proposed method generates random sample contingency tables directly from the sketch, the front of the inverted index. Because the term-by-document matrix is extremely sparse, it is possible for a relatively small sketch, $k$, to characterize a large sample of $D_s$ documents. The front of the inverted index not only tells us about the presence of the word in the first $k$ documents, but it also tells us about the absence of the word in the remaining $D_s - k$ documents. This observation becomes increasingly important with larger Web collections (with ever increasing sparsity). Typically, $D_s \gg k$.

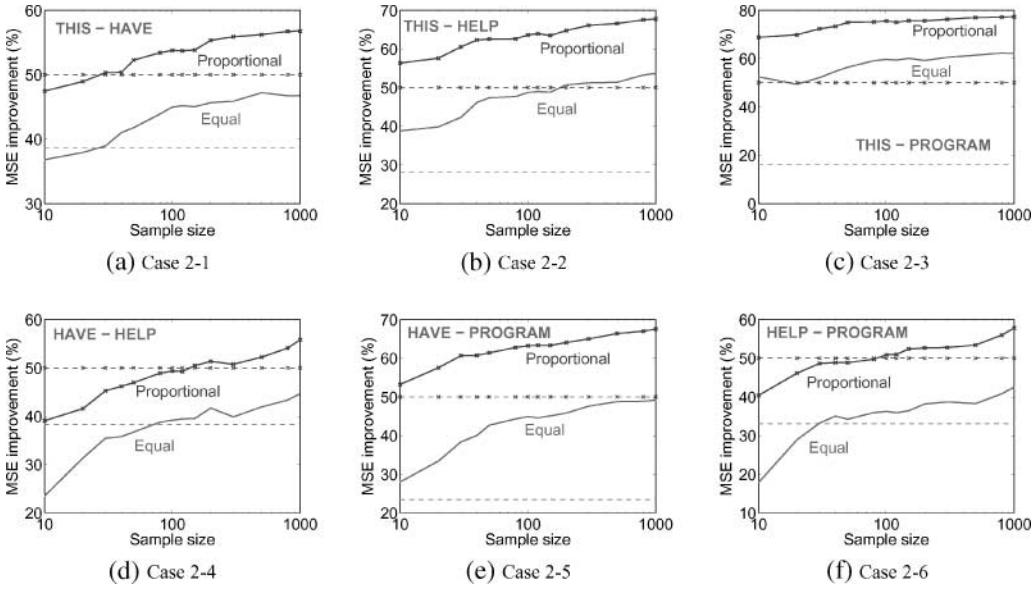



**Figure 30**
Compared with Broder's sketch, the relative MSE improvement should be, approximately, $\frac{\min(f_1, f_2)}{f_1+f_2}$ with equal samples, and $\frac{1}{2}$ with proportional samples. The two horizontal lines in each figure correspond to these two approximates. The actual improvements could be lower or higher. The figure verifies that proportional samples can considerably improve the accuracies.

To estimate the contingency table for the entire population, one can use the "margin-free" baseline, which simply multiplies the sample contingency table by the appropriate scaling factor. However, we recommend taking advantage of the margins (also known as document frequencies). The maximum likelihood solution under margin constraints is a cubic equation, which has a remarkably accurate quadratic approximation. The proposed MLE methods were compared empirically and theoretically to the MF baseline, finding large improvements. When we know the margins, we ought to use them.

Our proposed method differs from Broder's sketches in important aspects. (1) Our sketch construction allows more flexibility in that the sketch size can be different from one word to the next. (2) Our estimation is more accurate. The estimator in Broder's sketches uses one half of the samples whereas our method always uses more. More samples lead to smaller errors. (3) Broder's method considers a two-cell model whereas our method works with a more refined (hence more accurate) four-cell contingency table model. (4) Our method extends naturally to estimating multi-way associations. (5) Although this paper only considers boolean (0/1) data, our method extends naturally to general real-valued data; see Li, Church, and Hastie (2006, 2007).

Although we have used "word associations" for explaining the algorithm, the method is a general sampling technique, with potential applications in Web search, databases, association rules, recommendation systems, nearest neighbors, and machine learning such as clustering.

**Acknowledgments**
The authors thank Trevor Hastie, Chris Meek, David Heckerman, Mark Manasse, David Siegmund, Art Owen, Robert Tibshirani, Bradley Efron, Andrew Ng, and Tze Leung Lai. Much of the work was conducted at Microsoft while the first author was an intern during the summers of 2004 and 2005.

## Appendix 1: Sample C Code for Estimating Two-Way Associations

```
#include <stdio.h>
#include <math.h>
#define MAX(x,y) ( (x) > (y) ? (x) : (y) )
#define MIN(x,y) ( (x) < (y) ? (x) : (y) )
#define EPS 1e-10
#define MAX_ITER 50
int est_a_appr(int as,int bs,int cs, int f1, int f2);
int est_a_mle(int as,int bs, int cs, int ds, int f1, int f2,int D);

int main(void)
{
   int f1 = 10000, f2 = 5000, D = 65536;         // test data
   int as = 25, bs = 45, cs = 150, ds = 540;
   int a_appr = est_a_appr(as,bs,cs,f1,f2);
   int a_mle = est_a_mle(as,bs,cs,ds,f1,f2,D);
   printf("Estimate a_appr = %d\n",a_appr);      // output 1138
   printf("Estimate a_mle  = %d\n",a_mle);       // output 821
   return 0;
}

// The approximate MLE is the solution to a quadratic equation
int est_a_appr(int as,int bs,int cs, int f1, int f2)
{
   int sx = 2*as + bs,  sy = 2*as + cs, sz = 2*as+bs+cs;
   double  tmp = (double)f1*sy + (double)f2*sx;
   return  (int)((tmp-sqrt(tmp*tmp-8.0*f1*f2*as*sz))/sz/2.0);
}

// Newton's method to solve for the exact MLE
int est_a_mle(int as,int bs, int cs, int ds, int f1, int f2,int D)
{
   int a_min = MAX(as,ds+f1+f2-D), a_max = MIN(f1-bs,f2-cs);
   int a1  = est_a_appr(as,bs,cs,f1,f2);  // A good start
   a1 = MAX( a_min, MIN(a1, a_max) );     // Sanity check
   int k = 0, a = a1;
   do {
      a = a1;
      double q = log(a+EPS) - log(a-as+EPS)
               +log(f1-a-bs+1+EPS) - log(f1-a+1+EPS)
               +log(f2-a-cs+1+EPS) - log(f2-a+1+EPS)
               +log(D-f1-f2+a+EPS) - log(D-f1-f2-ds+a+EPS);
      double dq = 1.0/(a+EPS)-1.0/(a-as+EPS)
                 -1.0/(f1-a-bs+1+EPS) + 1.0/(f1-a+1+EPS)
                 -1.0/(f2-a-cs+1+EPS) + 1.0/(f2-a+1+EPS)
                 -1.0/(D-f1-f2-ds+a+EPS) + 1.0/(D-f1-f2+a+EPS);
      a1 = (int)(a - q/dq);   a1 = MAX(a_min, MIN(a1,a_max));
      if( ++k > MAX_ITER )  break;
   }while( a1 != a );
   return a;
}
```

## Appendix 2: Sample Matlab Code for Estimating Multi-Way Associations

```
function test_program
% A short program for testing the multi-way association algorithm.
% First generate a random gold standard dataset. Then construct
% sketches by sampling a certain portion of the postings. Associations
% are estimated by the exact MLE as well as the margin-free (MF) method.
%
clear all;
m = max(2,ceil(rand*6));    % Number of words (random)
D = 1000*m;                 % Total number of documents
f = ceil(rand(m,1)*D/2);    % document frequencies (random)
```

```
P{1} = sort(randsample(D,f(1)));  % Posting of the first word (random)
Pc = setdiff(1:D, P{1})';         % Compliment of the posting

% The postings of words 2 to m are randomly generated. 30% are
% sampled from the postings of word 1.
for i = 2:m
    k = ceil(0.3*min(f(i),f(1)));
    P{i} = sort([randsample(P{1},k);randsample(Pc,f(i)-k)]);  % Postings
end
X = compute_intersection(P,D);     % Gold standard associations
pc = 1;  % Pseudo-count(pc), pc=0 for no smoothing, pc=1 for "add-one".
sampling_rate = 0.1;
for i = 1:m
    k = ceil(sampling_rate*f(i));
    K{i} = P{i}(1:k);              % Sketches
end
% Estimate the associations and covariance matrices
[X_MLE, X_MF, Var_c, Var_o] = multi_way_est(K,f,D,pc);
% Display the estimations of associations
[X X_MLE X_MF]            %  [Gold standard, MLE, MF]
-------------------------------------------------

function [X_MLE, X_MF, Var_c, Var_o] = multi_way_est(K,f,D,pc);
% Matlab code for estimating multi-way associations
% K:      Sketches   (Cell array data type)
% f:      Document frequencies, a column vector
% D:      Total number of documents
% pc:     Pseudo-count for smoothing.
% X_MLE:  Maximum likelihood estimator (MLE),  a column vector
% X_MF :  Margin-free (MF) estimator, a column vector
% Var_c:  Conditional (on Ds) covariance matrix, using the estimated X,
% Var_o:  Covariance computed using the observed Fisher information
%
pc = max(pc,1e-4); % Always use a small pc for numerical stability.
m = length(K);     % The order of associations, i.e., number of words.
[A,A1,A2,A3,ind1,ind2] = gen_A(m);   % Margin constraint matrix

for i = 1:m;
    last_elem(i) = K{i}(end);
end
Ds = min(last_elem);
for i = 1:m
    K{i} = K{i}(find(K{i}<=Ds));  % Trim sketches according to D_s
end

S = compute_intersection(K,Ds);   % Intersect the sketches to get samples
[X_MLE, X_MF] = newton_est(pc,S,Ds,D,A,f); % Estimate X

% Conditional variance
Z_c = 1./(X_MLE+eps); Z1_c = diag(Z_c(ind1)); Z2_c = diag(Z_c(ind2));
Var_c = inv(Z1_c + A3'*Z2_c*A3)*(D/Ds-1);
% Observed variance
Z_o = S./(X_MLE+eps).^2; Z1_o = diag(Z_o(ind1)); Z2_o = diag(Z_o(ind2));
Var_o = inv(Z1_o + A3'*Z2_o*A3)*(D-Ds)/D;
-------------------------------------------------------

function [X_MLE,X_MF] = newton_est(pc,S,Ds,D,A,f)
% Estimate multi-way associations by solving a convex
% optimization problem using the Newton's method.
%
NEWTON_ERR = 0.001;        % Threshold for termination.
MAX_ITER = 50;             % Maximum allowed iteration.
N = length(S); m = length(f);  F = [f;D];
pc = min(pc,(D-Ds)/N);  % Adjust pc, if Ds is close to D.

% Solve a quadratic programming problem to find an initial
```

```
% guess of the MLE that minimizes the 2-norm with respect to
% the MF estimation and satisfies the constraints.
while(1)
    X_MF = (S+pc)./(Ds+N*pc)*D;      % Margin-free estimations.
    [X0,dummy,flag] = quadprog(2*eye(2^m),-2*X_MF,[],[],A,F,S+pc);
    if(flag == 1)    break;    end
    pc = pc/2;  % Occasionally need reduce pc for a feasible solution.
end

S = S + pc;  X_MLE = X0;  iter = 0;
while(1);
    D1 = -S./(X_MLE+eps);         % Gradient  (first derivatives)
    D2 = diag(S./(X_MLE.^2+eps)); % Hessian   (second derivatives)

    % Solve a linear system of equations for the Newton's step.
    M = [D2 A'; A zeros(size(A,1),size(A,1))];
    dx = M\[-D1; zeros(size(A,1),1)]; dx = dx(1:size(D2,1));
    lambda = (dx'*D2*dx)^0.5;       % Measure of errors
    iter = iter + 1;
    if(iter>MAX_ITER | lambda^2/2<NEWTON_ERR)     break;   end

    % Backtracking line search for a good Newton step size.
    z = 1; Alpha = 0.1; Beta = 0.5;     iter2 = 0;
    while(min(X_MLE+z*dx-S)<0|S'*log(X_MLE./(X_MLE+z*dx))>=Alpha*z*D1'*dx);
        if(iter2 >= MAX_ITER)    break;   end
        z = Beta*z;  iter2 = iter2 + 1;
    end
    X_MLE = X_MLE + z*dx;
end
------------------------------------------------------------

function  S = compute_intersection(K,Ds);
% Compute the intersections to generate a table with N = 2^m
% cells. The cells are ordered in terms of the binary representation
% of integers from 0 to 2^m-1, where m is the number of words.
%
m = length(K);  bin_rep = char(dec2bin(0:2^m-1)); S = zeros(2^m,1);
for i = 0:2^m-1;
    if(bin_rep(i+1,1) == '0')
        c{i+1} = K{1};
    else
        c{i+1} = setdiff([1:Ds]',K{1});
    end
    for j = 2:m
        if(bin_rep(i+1,j) == '0')
            c{i+1} = intersect(c{i+1},K{j});
        else
            c{i+1} = setdiff(c{i+1},K{j});
        end
    end
    S(i+1) = length(c{i+1});
end
------------------------------------------------------------

function [A,A1,A2,A3,ind1,ind2] = gen_A(m)
% Generate the margin constraint matrix and compute its decompositions
% for analyzing the covariance matrix
%
t1 = num2str(dec2bin(0:2^m-1));  t2  = zeros(2^m,m*2-1);
t2(:,1:2:end) = t1;  t2(:,2:2:end) = ',';
A = xor(str2num(char(t2))',1); A = [A;ones(1,2^m)];
for i = 1:size(A,1);
    [last_one(i)] = max(find(A(i,:)==1));
end
ind1 = setdiff((1:size(A,2)),last_one); ind2 = last_one;
A1 = A(:,ind1); A2 = A(:,ind2);  A3 = inv(A2)*A1;
```

## References


Achlioptas, Dimitris. 2003. Database-friendly random projections: Johnson-Lindenstrauss with binary coins. *Journal of Computer and System Sciences*, 66(4):671–687.

Aggarwal, Charu C., Cecilia Magdalena Procopiuc, Joel L. Wolf, Philip S. Yu, and Jong Soo Park. 1999. Fast algorithms for projected clustering. In *SIGMOD*, pages 61–72, Philadelphia, PA.

Aggarwal, Charu C. and Joel L. Wolf. 1999. A new method for similarity indexing of market basket data. In *SIGMOD*, pages 407–418, Philadelphia, PA.

Agrawal, Rakesh, Tomasz Imielinski, and Arun Swami. 1993. Mining association rules between sets of items in large databases. In *SIGMOD*, pages 207–216, Washington, DC.

Agrawal, Rakesh, Heikki Mannila, Ramakrishnan Srikant, Hannu Toivonen, and A. Inkeri Verkamo. 1996. Fast discovery of association rules. In U. M. Fayyad, G. Pratetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors. *Advances in Knowledge Discovery and Data Mining*. AAAI/MIT Press, pages 307–328, Cambridge, MA.

Agrawal, Rakesh and Ramakrishnan Srikant. 1994. Fast algorithms for mining association rules in large databases. In *VLDB*, pages 487–499, Santiago de Chile, Chile.

Agresti, Alan. 2002. *Categorical Data Analysis*. John Wiley & Sons, Inc., Hoboken, NJ, second edition.

Alon, Noga, Yossi Matias, and Mario Szegedy. 1996. The space complexity of approximating the frequency moments. In *STOC*, pages 20–29, Philadelphia, PA.

Baeza-Yates, Ricardo and Berthier Ribeiro-Neto. 1999. *Modern Information Retrieval*. ACM Press, New York, NY.

Boyd, Stephen and Lieven Vandenberghe. 2004. *Convex Optimization*. Cambridge University Press, Cambridge, UK.

Brin, Sergey, James Davis, and Hector Garcia-Molina. 1995. Copy detection mechanisms for digital documents. In *SIGMOD*, pages 398–409, San Jose, CA.

Brin, Sergey and Lawrence Page. 1998. The anatomy of a large-scale hypertextual web search engine. In *WWW*, pages 107–117, Brisbane, Australia.

Brin, Sergy, Rajeev Motwani, and Craig Silverstein. 1997. Beyond market baskets: Generalizing association rules to correlations. In *SIGMOD*, pages 265–276, Tucson, AZ.

Brin, Sergy, Rajeev Motwani, Jeffrey D. Ullman, and Shalom Tsur. 1997. Dynamic itemset counting and implication rules for market basket data. In *SIGMOD*, pages 265–276, Tucson, AZ.

Broder, Andrei Z. 1997. On the resemblance and containment of documents. In *The Compression and Complexity of Sequences*, pages 21–29, Positano, Italy.

Broder, Andrei Z. 1998. Filtering near-duplicate documents. In *FUN*, Isola d'Elba, Italy.

Broder, Andrei Z., Moses Charikar, Alan M. Frieze, and Michael Mitzenmacher. 1998. Min-wise independent permutations (extended abstract). In *STOC*, pages 327–336, Dallas, TX.

Broder, Andrei Z., Moses Charikar, Alan M. Frieze, and Michael Mitzenmacher. 2000. Min-wise independent permutations. *Journal of Computer Systems and Sciences*, 60(3):630–659.

Broder, Andrei Z., Steven C. Glassman, Mark S. Manasse, and Geoffrey Zweig. 1997. Syntactic clustering of the web. In *WWW*, pages 1157–1166, Santa Clara, CA.

Charikar, Moses S. 2002. Similarity estimation techniques from rounding algorithms. In *STOC*, pages 380–388, Montreal, Canada.

Chaudhuri Surajit, Rajeev Motwani, and Vivek R. Narasayya. 1998. Random sampling for histogram construction: How much is enough? In *SIGMOD*, pages 436–447, Seattle, WA.

Chaudhuri, Surajit, Rajeev Motwani, and Vivek R. Narasayya. 1999. On random sampling over joins. In *SIGMOD*, pages 263–274, Philadelphia, PA.

Chen, Bin, Peter Haas, and Peter Scheuermann. 2002. New two-phase sampling based algorithm for discovering association rules. In *KDD*, pages 462–468, Edmonton, Canada.

Church, Kenneth and Patrick Hanks. 1991. Word association norms, mutual information and lexicography. *Computational Linguistics*, 16(1):22–29.

Cover, Thomas M. and Joy A. Thomas. 1991. *Elements of Information Theory*. John Wiley & Sons, Inc., New York, NY.

David, Herbert A. 1981. *Order Statistics*. John Wiley & Sons, Inc., New York, NY, second edition.

Deming, W. Edwards and Frederick F. Stephan. 1940. On a least squares adjustment of a sampled frequency table when the expected marginal totals are

known. *The Annals of Mathematical Statistics*, 11(4):427–444.

Drineas, Petros and Michael W. Mahoney. 2005. Approximating a gram matrix for improved kernel-based learning. In *COLT*, pages 323–337, Bertinoro, Italy.

Dunning, Ted. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74.

Etzioni, Oren, Michael Cafarella, Doug Downey, Stanley Kok, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates. 2004. Web-scale information extraction in knowitall (preliminary results). In *WWW*, pages 100–110, New York, NY.

Garcia-Molina, Hector, Jeffrey D. Ullman, and Jennifer Widom. 2002. *Database Systems: The Complete Book*. Prentice Hall, New York, NY.

Gilbert, Anna C., Yannis Kotidis, S. Muthukrishnan, and Martin J. Strauss. 2003. One-pass wavelet decompositions of data streams. *IEEE Transactions on Knowledge and Data Engineering*, 15(3):541–554.

Guha Sudipto, Rajeev Rastogi, and Kyuseok Shim. 1998. Cure: An efficient clustering algorithm for large databases. In *SIGMOD*, pages 73–84, Seattle, WA.

Hastie, T., R. Tibshirani, and J. Friedman. 2001. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, New York, NY.

Haveliwala, Taher H., Aristides Gionis, and Piotr Indyk. 2000. Scalable techniques for clustering the Web. In *WebDB*, pages 129–134, Dallas, TX.

Haveliwala, Taher H., Aristides Gionis, Dan Klein, and Piotr Indyk. 2002. Evaluating strategies for similarity search on the web. In *WWW*, pages 432–442, Honolulu, HI.

Hidber, Christian. 1999. Online association rule mining. In *SIGMOD*, pages 145–156, Philadelphia, PA.

Hornby, Albert Sydney, editor. 1989. *Oxford Advanced Learner's Dictionary of Current English*. Oxford University Press, Oxford, UK, fourth edition.

Indyk, Piotr. 2001. A small approximately min-wise independent family of hash functions. *Journal of Algorithm*, 38(1):84–90.

Indyk, Piotr and Rajeev Motwani. 1998. Approximate nearest neighbors: Towards removing the curse of dimensionality. In *STOC*, pages 604–613, Dallas, TX.

Itoh, Toshiya, Yoshinori Takei, and Jun Tarui. 2003. On the sample size of k-restricted min-wise independent permutations and other k-wise distributions. In *STOC*, pages 710–718, San Diego, CA.

Lehmann, Erich L. and George Casella. 1998. *Theory of Point Estimation*. Springer, New York, NY, second edition.

Li, Ping. 2006. Very sparse stable random projections, estimators and tail bounds for stable random projections. Technical report, available from `http://arxiv.org/PS_cache/cs/pdf/0611/0611114v2.pdf`.

Li, Ping and Kenneth W. Church. 2005. Using sketches to estimate two-way and multi-way associations. Technical Report TR-2005-115, Microsoft Research, Redmond, WA, September.

Li, Ping, Kenneth W. Church, and Trevor J. Hastie. 2006. Conditional random sampling: A sketched-based sampling technique for sparse data. Technical Report 2006-08, Department of Statistics, Stanford University.

Li, Ping, Kenneth W. Church, and Trevor J. Hastie. 2007. Conditional random sampling: A sketch-based sampling technique for sparse data. In *NIPS*, pages 873–880. Vancouver, BC, Canada.

Li, Ping, Trevor J. Hastie, and Kenneth W. Church. 2006a. Improving random projections using marginal information. In *COLT*, pages 635–649, Pittsburgh, PA.

Li, Ping, Trevor J. Hastie, and Kenneth W. Church. 2006b. Very sparse random projections. In *KDD*, pages 287–296, Philadelphia, PA.

Li, Ping, Trevor J. Hastie, and Kenneth W. Church. 2007. Nonlinear estimators and tail bounds for dimensional reduction in $l_1$ using Cauchy random projections. In *COLT*, pages 514–529, San Diego, CA.

Manku, Gurmeet Singh, Sridhar Rajagopalan, and Bruce G. Lindsay. 1999. Random sampling techniques for space efficient online computation of order statistics of large datasets. In *SIGCOMM*, pages 251–262, Philadelphia, PA.

Manning, Chris D. and Hinrich Schutze. 1999. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, MA.

Matias, Yossi, Jeffrey Scott Vitter, and Min Wang. 1998. Wavelet-based histograms for selectivity estimation. In *SIGMOD*, pages 448–459, Seattle, WA.

Moore, Robert C. 2004. On log-likelihood-ratios and the significance of rare events.

In *EMNLP*, pages 333–340, Barcelona, Spain.
Pearsall, Judy, editor. 1998. *The New Oxford Dictionary of English*. Oxford University Press, Oxford, UK.
Ravichandran, Deepak, Patrick Pantel, and Eduard Hovy. 2005. Randomized algorithms and NLP: Using locality sensitive hash function for high speed noun clustering. In *ACL*, pages 622–629, Ann Arbor, MI.
Rosen, Bengt. 1972a. Asymptotic theory for successive sampling with varying probabilities without replacement, I. *The Annals of Mathematical Statistics*, 43(2):373–397.
Rosen, Bengt. 1972b. Asymptotic theory for successive sampling with varying probabilities without replacement, II. *The Annals of Mathematical Statistics*, 43(3):748–776.
Salton, Gerard. 1989. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*. Addison-Wesley, New York, NY.
Stephan, Frederick F. 1942. An iterative method of adjusting sample frequency tables when expected marginal totals are known. *The Annals of Mathematical Statistics*, 13(2):166–178.
Strehl, Alexander and Joydeep Ghosh. 2000. A scalable approach to balanced, high-dimensional clustering of market-baskets. In *HiPC*, pages 525–536, Bangalore, India.
Toivonen, Hannu. 1996. Sampling large databases for association rules. In *VLDB*, pages 134–145, Bombay, India.
Vempala, Santosh. 2004. *The Random Projection Method*. American Mathematical Society, Providence, RI.
Witten, Ian H., Alstair Moffat, and Timothy C. Bell. 1999. *Managing Gigabytes: Compressing and Indexing Documents and Images*. Morgan Kaufmann Publishing, San Francisco, CA, second edition.