Where and How Duplicates Occur in the Web

Álvaro Pereira Jr
Dept. of Computer Science
Federal Univ. of Minas Gerais
Belo Horizonte, Brazil
alvaro@dcc.ufmg.br

Ricardo Baeza-Yates
Yahoo! Research
Barcelona, Spain &
Santiago, Chile
ricardo@baeza.cl

Nivio Ziviani
Dept. of Computer Science
Federal Univ. of Minas Gerais
Belo Horizonte, Brazil
nivio@dcc.ufmg.br

Abstract

In this paper we study duplicates on the Web, using collections containing documents of all sites under the .cl domain that represent accurate and representative subsets of the Web. We identify duplicate and near-duplicate documents in our collections, studying the distribution of documents in clusters of duplicates. We also study the occurrence of duplicates in both parts of our Web graphs – connected and disconnected component – aiming to identify where duplicates occur more frequently. We originally show that the number of duplicates in the Web is expressively greater than the number of duplicates in the connected component of the Web graph. Works that previously estimated the number of duplicates in the Web used collections of connected components of the Web. In those cases the sample of the Web was biased.

1 Introduction

A portion of the Web is duplicated. Automatic duplicate detection is an important task specially in three aspects [5]. First, Web crawlers may choose not to collect duplicated pages in collections that it has visited elsewhere [1]. Second, ranking algorithms may privilege pages with many duplicates, as an indicative of page quality. Third, to save space in archiving or caching the Web.

In this paper we present an algorithm to find duplicate documents in a Web collection. The algorithm works by clustering duplicate documents [5]. Once our collections are not large, the algorithm uses the whole text of the documents for comparison, and does not associate a fingerprint to the documents. We perform experiments using three collections of Chilean Web pages, containing documents of all sites under the .cl domain, that represent accurate and representative subsets of the Web. We divide each collection into two subsets: connected and disconnected subcollections.

This paper has three main objectives. Firstly, we identify duplicate and near-duplicate documents in our collections, studying the distribution of documents in clusters of duplicates. Secondly, we study the occurrence of duplicates in both parts of our Web graphs – connected and disconnected components. With this study we aim to identify where duplicates occur more frequently and what is the impact on coverage when crawling only documents from the connected component of the Web graph (verifying if a representative subset of the Web is crawled). Thirdly, we study how the number of clusters and duplicates grows according to each iteration of the algorithm to detect duplicates.

The main goal of this paper is to show that the Web has much more duplicates than previously reported in the literature. Other works used collections crawled by following links in the Web [3]. In this case the sample of the Web is biased, because most of the times only documents from the connected component of the Web graph are crawled.

This paper is organized as follows. Section 2 presents definitions and information about the Web collections used in the experiments. Section 3 presents the algorithm to detect duplicates and near-duplicates. Section 4 presents our experimental results, with data about our Web collections and subcollections. Section 5 presents some related work about duplicates. Finally, Section 6 presents the conclusion of our work.

2 Definitions and Web Collections

In this section we present some definitions and the Web collections used in the experiments. The definitions are the following:

Definition 1 (Shingle Paragraph): It is a measure of content similarity among text documents, using the concept of shingles [2]. A shingle paragraph is a sequence of three sentences of the document, where a sentence is a sequence of words ended by a period. If a period is not found until the 150th character, then the sentence is finished at that

point and a new sentence is initialized at the 151th character. This limitation is due to the fact that some documents have no period (for example some program codes). In this work we use shingle paragraphs *without overlap* of sentences. As an example, suppose we have a document containing six sentences $s_1.\ s_2.\ s_3.\ s_4.\ s_5.\ s_6$, where s_i, $1 \leq i \leq 6$, is a sentence of the text. The shingle paragraphs without overlap of sentences are: "$s_1.\ s_2.\ s_3.$" and "$s_4.\ s_5.\ s_6.$".

Definition 2 (Cluster): It is a set of documents with exactly the same shingle paragraphs, without overlap of sentences. Thus, two documents belong to the same cluster if they have the same number of shingle paragraphs and every shingle paragraph of one document is found in the other document. Each document in a collection is either (i) *clustered*, if it belongs to a cluster, or (ii) *unique*, otherwise.

Definition 3 (Original Duplicated Document): It is the document that initiated a cluster. To find this document is not important. We just need to consider that one document in the cluster is the original duplicated document.

Definition 4 (Duplicate Document): It is any clustered document with exception of the original duplicated document.

Definition 5 (Near-Duplicate): It is a document with a given minimal percentage of identical shingle paragraphs of another document in the collection. This percentage is related to the number of shingle paragraphs of both documents.

For the experiments we used three collections of pages of the Chilean Web that were crawled in three distinct periods of time. Table 1 presents the main characteristics of the three collections. Each collection was crawled by the Web search engine TodoCL[1]. In each crawl, the complete list of the Chilean Web primary domains were used to start the crawling, guaranteeing that a set of pages under every Chilean domain (.cl) was crawled, once the crawls were pruned by depth. Once we used accurate and representative subsets of the Web, we also had accurate and representative samples of the Web for experimentation.

Table 1. Characteristics of the collections.

Col. name	Crawl date	# docs	Size (Gbytes)
2003	Aug 2003	2,862,126	9.4
2004	Jan 2004	2,796,749	11.8
2005	Feb 2005	2,883,455	11.3

[1] www.todocl.cl

3 Duplicate Detection Algorithm

In this section we present the algorithm to detect duplicate and near-duplicate documents in a collection C containing n documents.

The comparison step of the algorithm uses shingle paragraphs (see Definition 1). Collection C is divided into m subcollections S_i, $0 \leq i < m$. The algorithm runs in m steps. For each subcollection S_i, $0 \leq i < m$, the shingles of the documents in S_i are first inserted into a hash table. The collection must be divided because the hash table is loaded in the main memory.

Next, the shingles of documents in C are searched in the hash table. A duplicate is detected if all shingles of a document in C have a match in a document of S_i and both documents have the same number of shingles. At the end of each iteration i, the subcollection S_i is excluded from C ($C \leftarrow C - S_i$).

For each new duplicate (or near-duplicate) pair found, a new cluster (see Definition 2) is created and the duplicate pair is inserted into the new cluster. For that, a cluster identifier is associated to each document. If one of the documents of the pair was previously inserted into a given cluster, then the other document of the pair is inserted into the same cluster. At the end, the algorithm returns a set of clusters, each cluster containing a list of clustered documents.

Figure 1 illustrates the main steps of the algorithm using a sample test collection C containing $n = 20$ documents. In the example, collection C is divided into $m = 10$ subcollections, each one containing 2 documents. Sentences in each document are represented by letters, as shown in documents 1, 2, 19 and 20. Every document contains four shingle sentences (for instance, document 1 has the shingles "$a.\ a.\ a.$", "$b.\ b.\ b.$", "$c.\ c.\ c.$", "$d.\ d.\ d.$").

Following Figure 1, in the first iteration the documents 1 and 2 (from subcollection S_0) are inserted into the hash table. Next, the shingles of the documents of C (documents 1 to 20) are searched in the hash table. Therefore, it is possible to see that document 19 is a duplicate of document 2. In the second iteration, documents 3 and 4 (from subcollection S_1) are inserted into the hash table and the shingles of the documents of collection C (documents 3 to 20) are searched in the hash table. Next iterations occur similarly.

An important point is that the shingle paragraph technique used to detect duplicates may find false matches. False matches occur when two documents have the same number of identical shingle paragraphs, but with some repeated shingle. For example, suppose that document 3 in Figure 1 has the following sentences: $e.\ e.\ e.\ d.\ d.\ d.\ e.\ e.\ e.\ d.\ d.\ d$ (the shingles are "$e.\ e.\ e.$", "$d.\ d.\ d.$", "$e.\ e.\ e.$" and "$d.\ d.\ d.$"). Since every shingle of the document 3 is found in the hash table for the document 2 and the documents have the same number of shingle paragraphs, they

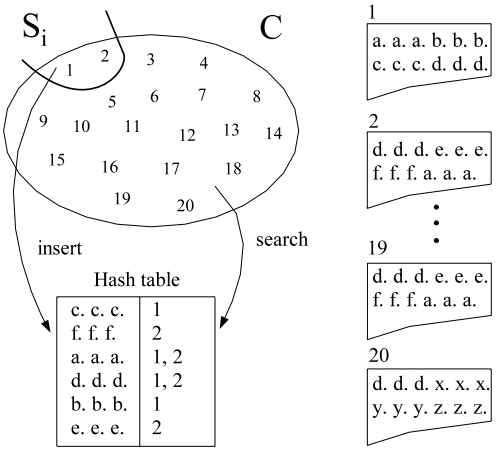


Figure 1. Process for duplication analysis.

are considered duplicates. As this situation seems to occur with a very small probability, the percentage results are not biased by false matches.

4 Experimental Results

In this section we describe our experimental results. In the experiments we considered only documents with size greater than 450 characters. This decision is due to guarantee that at least one complete shingle paragraph would represent the document (see Definition 1).

Section 4.1 presents the duplicates and near-duplicates percentage for our data set, dividing the documents of each collection into the connected or the disconnected component. Section 4.2 presents a study on how the number of clusters and duplicates grows according to each iteration of the algorithm to detect duplicates.

4.1 Study of Duplicates

Table 2 presents statistical results for the three collections presented in Table 1. Notice that the number of documents considered in each collection is smaller than the numbers shown in Table 1 because we did not consider documents with less than 450 characters. The number of clusters is the same as the number of original duplicated documents (see Definitions 2 and 3).

Table 2. General statistics about duplicates.

Col.	# docs	# clusters	# dup.	% dup.
2003	2,067,278	251,588	804,338	38.9%
2004	2,033,113	266,408	876,047	43.1%
2005	2,174,520	255,601	778,290	35.8%

According to Table 2, the number of clusters plus the number of duplicates for each collection represents more than 50% of the documents for collections 2003 and 2004. This means that for example, for collection 2004, only 48.9% of the documents are unique (i.e., do not belong to a cluster). This percentage is calculated considering the number of cluster plus the number of duplicates. The difference of the number of documents and this sum represents, for collection 2004, 48.9% of the documents.

By its turn, Table 3 shows the number of near-duplicates compared to the number of duplicates for each collection. We considered three values of minimal percentage of identical shingle paragraphs: 90%, 70% and 50% (see Definition 5).

Table 3. Data about near-duplicates.

Col.	% dup.	90% near.	70% near.	50% near.
2003	38.9	40.8	46.1	52.8
2004	43.1	44.5	47.9	53.4
2005	35.8	37.0	43.2	49.9

The percentage of near-duplicates for 90% of similarity is slightly greater than the percentage of duplicates. For instance, for collection 2003, only 1.9% of the documents share at least 90% of their shingles paragraphs and are not duplicates. On the other hand, for the same collection, 7.1% of the documents share at least 70% of their shingles paragraphs and are not duplicates, and 13.9% of the documents share at least 50% of their shingles paragraphs and are not duplicates.

Again analyzing Table 2, collection 2005 presents the smallest percentage of duplicates (35.8%) and collection 2004 presents the highest (43.1%). These figures are higher than the figures found in the literature (Shivakumar and Garcia-Molina [8] reports 27% and Fetterly, Manasse and Najork [7] reports 22%).

Our hypothesis was that the difference occurs because our collections were crawled based on a list of primary domains, which includes URLs that cannot be reached following links obtained from other pages. Most of the Web crawlers work following links, considering only documents from the connected component of the Web graph. Duplicates do not have the same inlinks that the original duplicated documents.

To study this hypothesis we developed a simple algorithm to simulate a Web crawler following links (navigational and external links). Every document reached from our crawl simulator algorithm belongs to a *connected* component of the Web graph. Documents that are not reached from the algorithm are classified as *disconnected*. Considering that we have used as seed a document from the strongly

connected component [6] of the Web graph, we classify as *connected* every document in the "central core" and in the "out" component of the Chilean Web bow tie [4]. Other components of the Chilean Web bow tie are classified as *disconnected*.

The connected components represent 51.6%, 56.8% and 63.9% of the collections 2003, 2004 and 2005, respectively. Table 4 presents the number and percentage of duplicates for the complete collection and, the connected and disconnected subcollections.

Table 4. Percentage of duplicates for the complete collection and, the connected and disconnected subcollections.

col.	# comp. (%)	# con. (%)	# discon. (%)
2003	804,338 (38.9)	276,714 (25.9)	429,965 (43.0)
2004	876,047 (43.1)	338,712 (29.3)	443,149 (50.4)
2005	778,290 (35.8)	360,903 (26.0)	323,174 (41.1)

Observing Table 4 we see that the real number of duplicates (for the complete collection) is 50% higher than the number of duplicates found for the connected subcollection, for collection 2003. On average this percentage is 45%, considering the three collections. The number of duplicates found for the disconnected collection is on average 65% higher than the number of duplicates for the connected collection. These expressive percentages show that most of duplicates occurs in the disconnected component of the Web graph. We also conclude that the absolute real number of duplicates is two or three times the absolute number of duplicates in the connected component.

Now we study relations between duplicates and clusters sizes. Figures 2, 3 and 4 present a logarithmic scale distribution of the number of duplicates per clusters for the collections 2003, 2004 and 2005, respectively. For every collection we present the distribution for the complete collection and, the connected and disconnected subcollections. The plots follow Zipf-like distributions, where many clusters have few documents and few clusters have many documents.

Initially we analyze only the letters *a)* of Figures 2, 3 and 4, which represent the distributions for the complete collections.

For collection 2004 we found nine clusters with more than 10,000 documents and two of them had more than 20,000 documents. The duplicates belonging to these clusters represent 7.1% of the documents of the collection. This explains the high number of duplicates for collection 2004 in relation to the other collections studied, as shown in Table 2.

For collection 2003, 95.7% of the clusters have ten or less documents. Documents in these small clusters represent 63.3% of clustered documents and only 40.5% of duplicate documents. The same figures were found for the other collections, which means that large clusters have more influence in the number of duplicates than short clusters.

Clusters with two documents (with only one duplicate) are very frequent. Collections 2003, 2004 and 2005 have 158,288, 166,929, 159,695 clusters containing only two documents, respectively. In every case these values represent about 63% of the clusters, but only 19% of the duplicates, approximately.

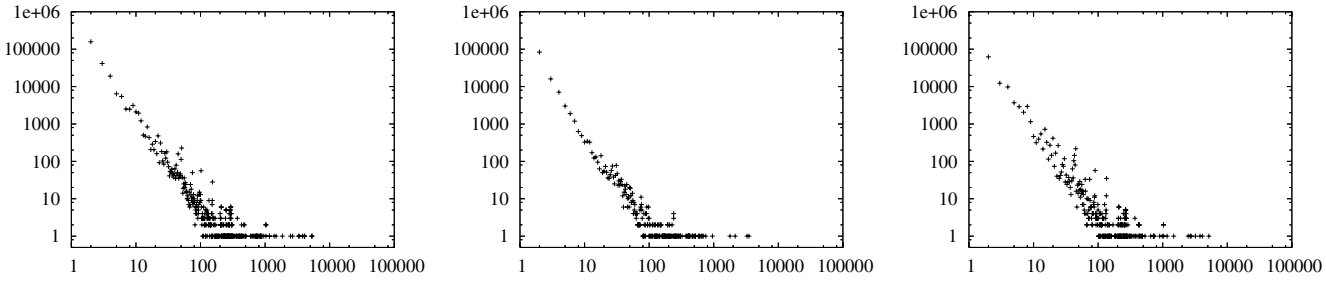
We manually analyzed some clusters of duplicates, aiming to investigate the reasons for the duplications. For the large clusters, we identified that the duplicates belong to the same site, probably replicated by the publisher of the original document. On the other hand, small clusters most of the times have documents that belongs to distinct sites. This type of replication characterizes plagiarized documents or documents in a mirrored site.

Now we analyze the distributions for the connected and disconnected subcollections of Figures 2, 3 and 4 (letters *b* and *c* of the figures).

For the three collections the number of clusters with only two documents is greater for the complete collection than for the sum of the two subcollections. For instance, in collection 2003, the number of clusters with only two documents for the complete collection is approximately 158,000, while for connected and disconnected subcollections the values are 83,000 and 63,000, respectively. In general, the same behavior is observed for the clusters with up to 12 documents.

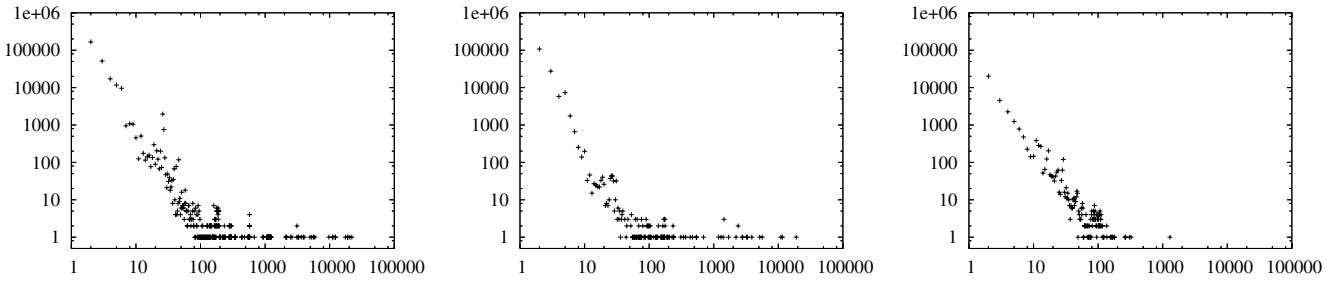
Considering that clusters in a complete collection can be divided into two other clusters when the collection is divided into connected and disconnected subcollections, it is impossible to find a cluster in a subcollection with more documents than the same cluster in the complete collection. As clusters with less than 12 documents occur more frequently in the complete collection, we conclude that a portion of the clusters disappeared when a collection is divided. Notice that only clusters with two documents can disappear. Other clusters, when divided, become two smaller clusters or one cluster with one less documents.

Comparing the three plots for each year, we see that the tail for the complete collections are always longer than for the connected and disconnected subcollections. If a point is found in a given position in the complete collection plot and it is not found in the same position in one of the subcollections, it means that the cluster is divided into the two subcollections. By the other hand, observing Figures 2 and 4 for collections 2003 and 2005, we see that most of the clusters with more than 1,000 documents remains its number of documents in the disconnected subcollection (it is not true for collection 2004).



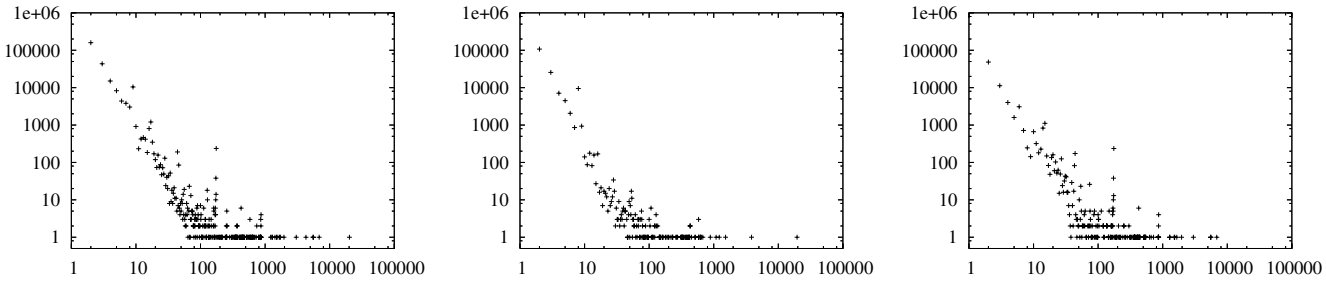
(a) Complete 2003 collection. (b) Connected 2003 subcollection. (c) disconnected 2003 subcollection.

Figure 2. Distribution (log scale) of the number of documents (axis x) per cluster (axis y) for collection 2003.



(a) Complete 2004 collection. (b) Connected 2004 subcollection. (c) Disconnected 2004 subcollection.

Figure 3. Distribution (log scale) of the number of documents (axis x) per cluster (axis y) for collection 2004.



(a) Complete 2005 collection. (b) Connected 2005 subcollection. (c) Disconnected 2005 subcollection.

Figure 4. Distribution (log scale) of the number of documents (axis x) per cluster (axis y) for collection 2005.

Duplicates are unwanted in Web databases for search engines. Thus, the important portion of our Web collections is the union of the unique documents with the original duplicated document from each cluster (i. e., one of the documents from each cluster). The difference between the number of documents and the number of duplicates in Table 2 represents the number of important documents in each collection. For this portion of the documents we will refer to as *useful* documents.

Now we study the occurrence of useful documents in connected subcollections. Our hypothesis is that a collection composed by just following links in the Web contains a very representative portion of the useful documents, in terms of coverage. Table 5 presents the number of clusters in the complete collections with at least one document in the connected subcollections and Table 6 presents the occurrence of unique documents in the connected subcollections, for collections 2003, 2004 and 2005.

Table 5. Number of clusters with at least one document in the connected subcollections.

col.	# clusters	# docs in con.	% docs in con.
2003	251,588	180,563	71.8%
2004	266,408	202,562	76.0%
2005	255,601	208,648	81.6%

Table 6. Occurrence of unique documents in the connected subcollections.

col.	total # unique	# unique in connected	% unique in connected
2003	1,011,352	609,501	60.3%
2004	890,658	613,362	68.9%
2005	1,140,629	819,578	71.9%

According to Table 5, for collection 2005, 81.6% of the clusters have at least one document in the connected subcollection 2005. According to Table 6, for collection 2005, 71.9% of the unique documents belong to the connected subcollection 2005. Now we are able to calculate the number of useful documents found in the connected subcollection. For collections 2003, 2004 and 2005, respectively 62.6%, 70.5% and 73.6% of the useful documents are found in the connected subcollections.

4.2 Growth of Clusters and Duplicates

In this section we study how the number of clusters and duplicates grows according to each iteration of the algorithm to detect duplicates. Remember that collection C is divided into m subcollections S_i, $0 \leq i < m$, and the documents in C are compared with the documents in S_i.

Figure 5 presents the growth of the number of clusters and duplicates according to each iteration, for the collections 2003, 2004 and 2005. The axis x represents each iteration, according to the variable m in the algorithm described in Section 3. For collection 2003 we used $m = 15$, for collection 2004 we used $m = 40$ and for collection 2005 we used $m = 50$. We chose different values to observe if the growth changes. The linear curve represents the sum of the number of documents considered in each subcollection S_i.

According to Figure 5 the number of clusters and duplicates grow as logarithmic-like functions (the first value in the axis x is 1). It occurs because in the first iterations the largest clusters are identified, once it is very probable that one of the duplicates in the largest clusters belongs to the first subcollections used. Notice that the algorithm works by comparing a subcollection of documents with the complete collection. For the same reason, all documents of a given cluster are identified when the first document of that cluster belongs to the subcollection used in an iteration.

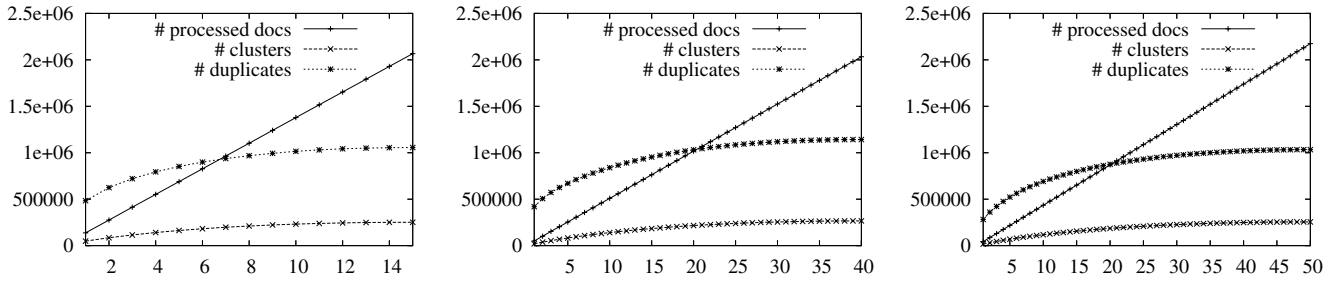
Figure $5-b$ (collection 2004) shows that about 420,000 duplicates are identified in the first iteration. If we compare it with the collection 2005 (with 280,000 duplicates in the first iteration), this value is considerable higher because the collection 2004 has many clusters with many documents.

We observe that in general the number of clusters grows gentler than the number of duplicates. It occurs because there are many clusters with few documents that are not identified in first iterations. Since these clusters have only one or few duplicates, the documents in new clusters do not increase much the number of duplicates.

In every collection, with 50% of the iterations, 90% of the duplicates are identified. As the percentage growth of duplicates per iteration, for each collection, is very similar, observing the percentage growth of one collection it is possible to estimate the number of duplicates of another collection by processing only few iterations.

As an example we estimated the number of duplicates in the collection 2005 using percentage data from 2004. Initially, we used only 20% of the iterations (10 iterations) for the collection 2005. Using the absolute number of clusters and duplicates found until iteration 10 for the collection 2005 and considering that these values represent the same percentage that iteration 8 (20% of the iterations) for the collection 2004, we are able to estimate the percentage of duplicates in collection 2005. The found percentage was 34.4%, against 35.8% for the real percentage of duplicates.

Performing the same calculus for 50% of the iterations, we found 35.6% of duplicates, which is very close to the real value, with an error of 0.20%. Since with 50% of the iterations we find more than 90% of the duplicates, the precision is very high. These estimations would be important



(a) Collection 2003. (b) Collection 2004. (c) Collection 2005.

Figure 5. Growth of the number of documents, clusters and duplicates (axis y) per iteration (axis x).

if we are interested to estimate the number of duplicates for large collections, performing only few iterations.

5 Related Work

In this section we present works related to finding and eliminating duplicates and near-duplicates on the Web. Broder et al. [3] used shingle to estimate the text similarity among 30 million documents retrieved from a walk of the Web. The similarity was evaluated using a sample of fixed size (a fingerprint) for each document. Considering a resemblance of 50%, they found 2.1 million clusters of similar documents, a total of 12.3 million documents.

Shivakumar and Garcia-Molina [8] crawled 24 million Web documents to compute the overlap between each pair of documents. Pieces of the documents are hashed down to a 32-bits fingerprint and stored into a file. A similarity is detected if two documents share a minimal number of fingerprints. The number of replicas is estimated as approximately 27%. Cho, Shivakumar and Garcia-Molina [5] combined different heuristics to find replicated Web collections. They used 25 million Web documents and found approximately 25% of duplicates.

Fetterly, Manasse and Najork [7] extended the Broder et al.'s work [3] in terms of the number of compared documents and investigated how clusters of near-duplicate documents evolve with the time. They found that clusters of near-duplicate documents are fairly stable and estimated the duplicates in approximately 22%.

Our work differs from the above mentioned papers in three main aspects: i) we study duplicate documents in collections where documents of all sites under a given domain (.cl, from Chile) were crawled, that represent accurate and representative subsets of the Web, ii) we compare the number of duplicates for the complete collection and, for the connected and disconnected subcollections, and iii) once our collections are not very large, we did not use fingerprints, improving the precision of the results.

6 Conclusions

In this paper we have presented a study on duplicates in the Web. We have suggested that the Web has many more duplicates than previously reported in the literature. Other works use collections crawled by following links in the Web. The number of duplicates found for the disconnected collection is on average 65% higher than the number of duplicates for the connected collection. Once we have used accurate and representative subsets of the Web, we suppose that our conclusions can be extended to other Web collections.

Our results have an important impact for search engine Web crawlers. Once the Web grows very fast, is very dynamic and has many replication, search engines have to heuristically decide which pages to crawl. In this paper we have shown that the connected component of our Web graphs contain a representative portion of the Web, in terms of coverage. To project a Web crawler many aspects must be considered. Considering the coverage and elimination of duplicates, Web crawlers designers may choose to crawl only pages reached by links, instead of listing every found directory and crawling every document in a directory.

As future work it is interesting to study the characteristics of the documents in the disconnected component of the Web. We know that they are many times new documents [4]. Maybe it is heuristically possible to separate the interesting new documents from other documents that are many times replications of documents in the connected component of the Web graph.

Acknowledgements

This work was partially funded by Spanish MEC Grant TIN 2005-09201 (R. Baeza-Yates and A. Pereira Jr) and by Brazilian GERINDO Project–grant MCT/CNPq/CT-INFO 552.087/02-5 (N. Ziviani and A. Pereira Jr), and CNPq Grants 30.5237/02-0 (N. Ziviani) and 14.1636/2004-1 (A. Pereira Jr).

References

[1] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press/Addison-Wesley, England, 1999.

[2] A. Broder. On the resemblance and containment of documents. In *Compression and Complexity of Sequences (SEQUENCES'97)*, pages 21–29. IEEE Computer Society, 1998.

[3] A. Broder, S. Glassman, M. Manasse, and G. Zweig. Syntactic clustering of the Web. In *Sixth International World Wide Web Conference (WWW'97)*, pages 391–404, 1997.

[4] A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins, and J. Wiener. Graph structure in the web. In *Ninth International World Wide Web Conference (WWW'00)*, pages 309–320, Amsterdam, Netherlands, May 2000.

[5] J. Cho, N. Shivakumar, and H. Garcia-Molina. Finding replicated Web collections. In *ACM International Conference on Management of Data (SIGMOD)*, pages 355–366, May 2000.

[6] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to algorithms*. MIT Press/McGraw-Hill, San Francisco, CA, 1990.

[7] D. Fetterly, M. Manasse, and M. Najork. On the evolution of clusters of near-duplicate Web pages. In *First Latin American Web Congress*, pages 37–45, Santiago, Chile, November 2003.

[8] N. Shivakumar and H. Garcia-Molina. Finding near-replicas of documents on the Web. In *International Workshop on the World Wide Web and Databases (WebDB'98)*, pages 204–212. Lecture Notes in Computer Science, 1998.