Automatic Document Collection Management: The Case of Duplicates

Kyle Williams[1], C. Lee Giles[1,2]
Information Sciences and Technology[1], Computer Science and Engineering[2]
The Pennsylvania State University, University Park, PA, 16802, USA

Introduction

Automatically managing document collections is desirable for a number of reasons, such as to:

- Make information easier to discover and manage
- Reduce storage and computation requirements
- Comply with organizational policy and legal requirements for document management.

However, for large document collections, manual management is impractical since it is too time consuming. Thus, this research investigates the use of automatic methods for document collection management, focusing on duplicate detection and removal as a special case of automatic collection management.

The Case of Duplicates in the CiteSeer Digital Library

CiteSeer is a digital library of over 2 million academic papers that were automatically crawled from the Web and added to the digital library via automatic methods. The automatic detection and removal of duplicate documents, which make up approximately 2% of the documents in CiteSeer, could be considered a special case of automatic collection management.

Duplicate documents are undesirable in CiteSeer since:

- They clutter search results with the same documents [1]
- The skew automatically generated collection statistics [2]
- They require additional store and computation [1]

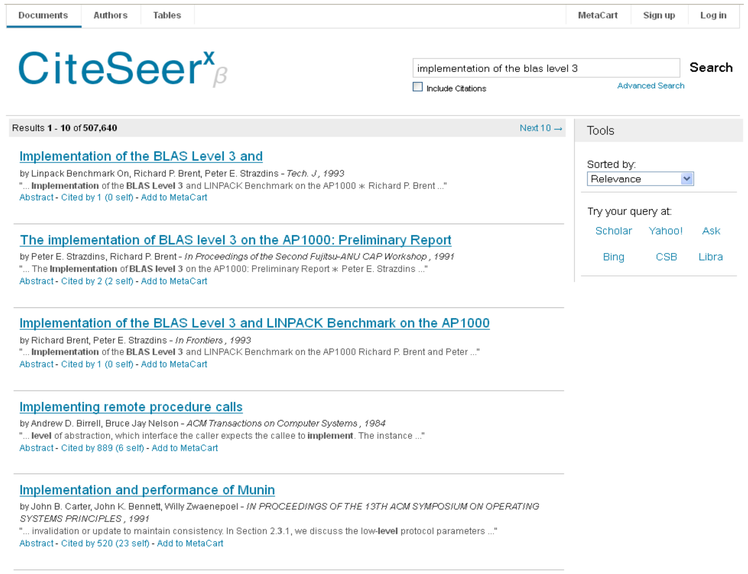

Figure: Duplicate search results in the CiteSeer digital library

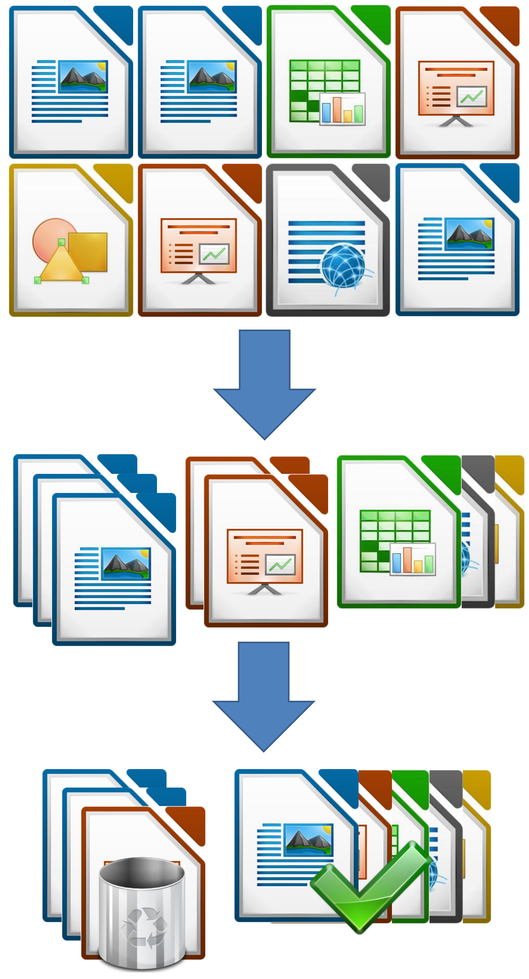

Figure: Automatic collection management applied to duplicates

Methodology and Findings

- An experiment was conducted to determine the extent to which duplicate documents exist in CiteSeer by applying two state of the art duplicate detection algorithms to a random sample of 100 000 documents from the CiteSeer collection[3]
- The first algorithm is based on “shingles”, which are sequences of words that appear in documents and duplicates documents are identified by the overlap of their shingles [4]
- The second algorithm is based on a technique where each word in a document contributes to a document *signature* and duplicate documents are identified by the similarity of their signatures [5]
- When ensuring that no errors are made in duplicate classification, 98% of known duplicates could be found using these techniques
- The different methods performed approximately equally [3]
- Approximately 2% of all of the documents in the sample were found to be duplicates [3]
- The methods scale linearly and are fast on large datasets [3]

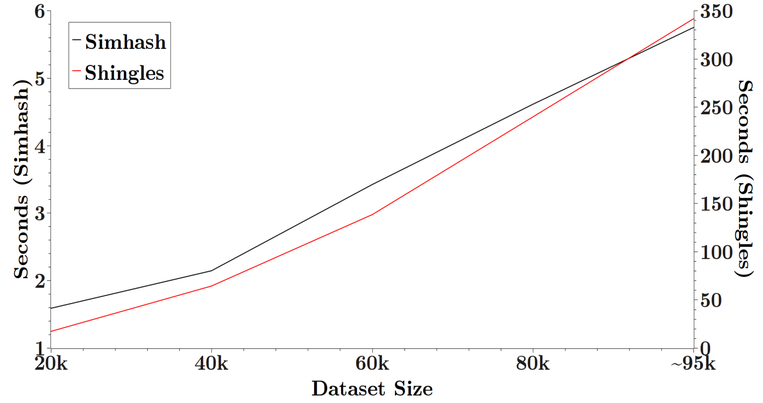

Figure: Processing times for different sized datasets and different algorithms

Future Directions

Future work related to duplicate detection involves:
- Ranking of duplicate documents
- Removal and merging of duplicates

Duplicate detection is only one process in automatic collection management. Future research to generalize automatic collection management techniques involves:

- Systems for interpreting and enforcing collection management policies
- Classification and ranking of documents according to policies so that actions can be taken on the documents
- Scaling to large collections and big data

Acknowledgments

We greatly acknowledge partial support from the National Science Foundation under grant No. 1143921 and useful suggestions by Madian Khabsa, Sagnik Ray Choudhury, and members of the CiteSeer research group.

References

[1] M. S. Manasse. 2004. Finding similar things quickly in large collections - Microsoft Research.
[2] A. Chowdhury, O. Frieder, D. Grossman, and M. C. McCabe. 2002. Collection statistics for fast duplicate document detection. Trans. Inf. Syst. 20, 2, 171-191.
[3] K. Williams and C. Lee Giles. 2013. Near Duplicate Detection in an Academic Digital Library. Under Review.
[4] A. Z. Broder, S. C. Glassman, M. S. Manasse, and G. Zweig. 1997. Syntactic clustering of the Web. Comput. Netw. ISDN Syst. 29, 8-13, 1157-1166.
[5] M. S. Charikar. 2002. Similarity estimation techniques from rounding algorithms. In Proceedings of the Symposium on Theory of computing, 380-388.