

Essential Deduplication Functions for Transactional Databases in Law Firms

Jack G. Conrad
Research & Development
Thomson Legal & Regulatory
St. Paul, Minnesota 55123 USA
Jack.G.Conrad@Thomson.com

Edward L. Raymond, Jr.
Content Operations
Thomson–West
Rochester, New York 14694 USA
Ed.Raymond@Thomson.com

ABSTRACT

As massive document repositories and knowledge management systems continue to expand, in proprietary environments as well as on the Web, the need for duplicate detection becomes increasingly important. In business enterprises such as law firms, effective retrieval applications depend upon such functionality. Today's Internet-savvy users are not interested in search results containing numerous sets of duplicate documents, whether exact duplicates or near variants.

This report addresses our work in the domain of legal information retrieval, working with a large, *transactional* knowledge management system. We specifically explore the occurrence and treatment of identical, near-identical, and fuzzy duplicate sub-documents ('clauses') in a contracts database. To our knowledge, we are the first to use principled methods to construct a test collection of transactional documents for such research purposes, one which identifies a variety of duplicate types and is deployed to establish baseline algorithmic approaches to deduplication.

We subsequently investigate the application of digital signature techniques to characterize and compare similar clauses in order to identify duplicates and near duplicates. This approach establishes a baseline using methods and algorithms first developed in a parallel domain. It produces a set of promising results following an extensive assessment phase involving direct comparisons with gold training and test data created by expert attorneys working in the transactional domain.

Categories and Subject Descriptors

H.2.4 [Information Systems]: Database Management—*Systems—Textual Databases*; H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval—*Selection Process*; H.3.m [Information Storage and Retrieval]: Miscellaneous—*Test Collections*

General Terms

Experimentation, Measurement, Design, Algorithms

Keywords

data management, duplicate detection, document signatures

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ICAIL '07, June 4-8, 2007, Palo Alto, California USA
Copyright 2007 ACM 978-1-59593-680-6/07/0006 ...\$5.00.

1. INTRODUCTION

Both on the World Wide Web and in proprietary data environments, it is currently possible to have tens of millions of textual objects indexed as part of the same collection.¹ Transactional databases are particularly challenging in that the contracts they contain consist of a highly hierarchical structure where the same text object may appear at several levels across documents. In large knowledge management environments like law firms, there may be terabytes of information stored. In such environments, the identification of duplicate documents is an important factor for a practical and robust data delivery platform.

One goal of this work is to leverage domain expertise in order to characterize the duplication existing in such large textual collections. We subsequently try to validate the completeness and reliability of this effort with analyses of assessor agreement, error rates, and significance.

This work makes two significant contributions. First, it creates and deploys a deduping test collection by harnessing:

- (a) real user queries;
- (b) a significant collection from an operational setting;
- (c) professional assessors possessing substantial knowledge of the domain and its clients.

In addition, this work expands the discussion of online (real time) deduping in Cooper, et al. [10]. Other recent work has often been syntax rather than lexical-based, Web-based (focusing on issues such as URL replication and instability), and conducted offline (e.g., examining large numbers of permutations before constructing a feature set). Previous research is substantially different than our current efforts which target a dynamic law firm environment. The novelty of this work thus derives from its being the first to focus on duplicate document detection for *transactional* documents in the legal domain.

The remainder of this paper is organized as follows: Section 2 reviews related work in duplicate document detection. In Section 3, we present the methodology used to assemble our duplicate document detection collection. Section 4 describes a baseline deduping algorithm for non-identical duplicates and the preliminary trials to assess it. Section 5 delves into performance and evaluation issues associated with the algorithm. We present our conclusions in Section 6 and discuss Future Work in Section 7.

¹In this paper, we will use "collection" to refer to a database of textual documents, and "deduping" to refer to duplicate document detection and subsequent removal or suppression.

or even term distribution information. It may be effective for general Web-based information about which we may know little, but for directed domains for which we do know quite a bit, it may work at a disadvantage.

Both of the above approaches rely on hash values for each document sub-section, and both prune these hash values to reduce the number of comparisons that the algorithms must perform. The computational complexity and thus resultant efficiency of the schemes are therefore quite dependent on the manner and extent to which the pruning is performed. The more aggressive the pruning, the more efficient are the algorithms, at the cost of increasing the prospects for identifying false positive duplicates.

Chaudhuri, Ganti and Motwani recently approached duplicate detection from a record merging perspective and focused on eliminating the problem based on two fundamental properties of duplicate tuples: compact set and sparse neighborhood [6].

Shivakumar and Garcia-Molina describe factors in identifying nearly identical documents on the Web for the benefit of Web crawlers and Web archivers [26]. They consequently concentrate on computing pairwise document overlap among pages commonly found on the Web. Their workshop draft specifies Web-based applications for the identification of near replicas: (1) more efficient web-crawling, focusing on speed and richer subsets rather than time-consuming comprehensiveness; (2) improved results ranking (or re-ranking), inspecting the environments from which Web documents originate; and (3) archiving Web documents, enabling greater compression of shorter pages that replicate more complete document sets. The authors reveal that there is a much greater incidence of (a) server aliasing; (b) URL aliasing; and (c) replication of popular documents such as FAQs and manuals than initially believed. Some of the resource-saving concepts they propose have been harnessed by a number of Web search engines, including Google [2].

In one of the most comprehensive works to date, Chowdhury, Frieder, Grossman and McCabe [7] refine their collection statistic, idf-based deduping algorithm for efficiency and effectiveness on both Web-based and non-Web-based test collections. They also compare its performance to other state-of-the-art techniques such as shingling and super-shingling. The authors demonstrate that their approach, called I-Match, scales in terms of number of documents and works well for documents of diverse sizes. They claim that in addition to improving accuracy over competing approaches like shingling, it executes in one-fifth the time. The authors briefly describe how the collection statistics for the algorithm can come from training collections in rapidly changing data environments.

In more recent work, Kolcz, Chowdhury and Alspector offer an alternative to I-Match that relies upon a set of digital signatures for a document created from randomized subsets of the global lexicon [17]. The motivation for this approach is to compensate for the case where the fraction of terms participating in the I-Match signature (hash) relative to the terms in the lexicon used is small. The significance of the approach stems from the fact that I-Match may result in false positive matches if a large document has a small term intersection with the lexicon used. The authors show that this approach outperforms traditional I-Match with an improvement in overall recall of 40% to 60%. An advantage of the scheme is its increased insensitivity to word permutations

and its document length independence. The authors do not quantify, however, the additional cost associated with generating the multiple lexicons, creating the multiple $(K + 1)$ signatures, and comparing one $(K + 1)$ tuple with another.⁴ The computational cost of this improved performance appears to be implementation dependent. For non-critical applications such as that mentioned by the authors—reducing spam by a significant percentage in a large ISP provider e-mail system—the benefits of the technique may outweigh its costs and justify its deployment.

The Web-related research of Park, Pennock, Giles and Krovetz relies heavily on the notion of lexical signatures, consisting of roughly five key identifying words in document, based either on their low df or high tf properties [23]. What distinguishes this work is that its eight signature variations are designed and evaluated for their ability either to retrieve the associated document in question in the top ranks of a search result (unique identification) or to retrieve alternative relevant documents should the document be lost (e.g., due to a broken link) (relevance properties). They determine that hybrid signatures consisting of only a couple of low df terms plus several high tf or high $tf \cdot idf$ terms produce the most effective unique and relevant properties for Web page signatures.

Cooper, Coden and Brown discuss methods for finding identical as well as similar documents returned from Web-based and internal IBM enterprise searches [10]. The techniques are based upon the creation of a digital signature composed of the sum of the hash codes of the “salient” terms found in a document. The document signatures are intended to provide a short-hand means of representing the top terms in documents to facilitate fast comparisons. Their tests generally rely upon a single query and may warrant more comprehensive evaluation. The authors describe their approach as the “logical extreme of super-shingl[ing],” yet characterizing a document by summing its Java hash codes for hundreds or more terms may raise questions about the principled, dependable nature of the technique.⁵

The significance of this overview is that there has not yet been established a standard information retrieval (IR) test collection for duplicate document detection. As we approached the problem, this was our first essential step, since without a validated test collection, we could not have confidence in the approaches and performance measures that followed.

3. METHODOLOGY

3.1 Background

Initially the Thomson business unit responsible for law firm knowledge management (West km) asked us for technologies to identify and treat duplicate documents in transactional databases. In response, we began characterizing the distribution of duplicate types across a representative contracts collection that we constructed from documents obtained through an acquisition. The collection statistics for the resulting contracts database are shown in Table 1.

⁴ $(K + 1)$: 1 represents the original and complete I-Match signature and K represents the number of permutations of the original lexicon. Kolcz, et al. experimented with K ranging from 1 to 10.

⁵The test to determine whether a technique is principled, in this case, depends upon whether it avoids leaving anything to chance or probabilistic uncertainty. In short, is the approach highly reliable?

3.3 Collection Generation and Domain Expert Assessments

To test our approach, we selected a total of 50 real user information requests from a query log that also included a human-assigned transactional query *category*. These logs originated from a production environment that was incorporated into West km. The queries were randomly selected with the exception that a results list of at least 20 documents was required. A sample of these categories is shown in Table 4 while a sample of the queries is shown in Table 5. The average query contained roughly three terms. Each query was run using the West km Transactional system which provides natural language search capability, depending on the preference of the user. After running these queries against our test collection, which consisted of approximately 82,500 clauses, we assembled the top twenty clauses returned from each query. While the cumulative result set consisted of nearly 1,000 clauses, each set of twenty clauses was reviewed by two attorney-editors,⁶ in order to identify their duplicate subsets.⁷ This process helped us produce standard training and test sets against which computational approaches would be compared.⁸ Collection statistics for the resultant training and test sets are shown in Table 6.

No.	Category of Contract (Selected)
A.	Standard Provisions (All Contracts)
B.	Acquisition Agreements
C.	Employment Agreements
D.	Escrow Agreements
E.	Investor Rights Agreements
F.	Joint Ventures
G.	License Agreements
H.	Limited Partnership Agreements
I.	Loan Agreements
J.	Merger Agreements
K.	Real Estate (REIT/Partnerships)
L.	Reorganization Agreements
M.	Security Agreements
N.	Underwriting Agreements

Table 4: Transactional Law–Sub-Categories

Query Type	Transactional Law Queries
Acquisitions	“excluded liabilities”
Employment	“put option”
Joint Venture	“event of default”
Limited Partnership	“initial capital contribution”
Loan Agreement	“fixed charge coverage ratio”
Security Agreement	“sale of collateral”

Table 5: Sample Qrys–Duplicate Set Construction

3.3.1 Details of the Document Inspections

In this trial, we applied definitions of non-identical duplicates that were drafted by customer and business unit work groups. The resulting definition states that two texts are duplicates if they retain much of the same language and

⁶Our attorney-editors, who are required to have law degrees, spend a significant portion of their day working closely with essential analytical legal texts.

⁷Inter-assessor agreement is discussed in Section 3.4.

⁸“Training” is not used here in the Machine Learning sense involving automatic learning; rather, it signifies an initial round in which we were permitted to establish the algorithm’s optimal parameter settings.

are at least 90% similar.⁹ To formally review the duplication status of our result sets, we assembled twelve attorney-editors. The 50 sample queries were divided into two sets of 25, the first set to be used to train a prototype system and the second set to test it. The process by which the query results were judged was scheduled over four weeks time (as indicated in Table 7). During week 1, results from the training queries were assessed for their duplication status. Each team reviewed the results from 25 queries, approximately 5 queries per team per day. Although members of the same team reviewed the same results, they did so independently.

Assessor Pair	Team A	Team B
Week 1	25 Qrys	25 Qrys
Week 2	<i>Arbitration</i>	<i>Arbitration</i>
Total	25 Qrys	25 Qrys
Combined	50 Qrys	

Table 7: Scheduling of Assessments

The assessors also had access to the term counts available in the core documents (which excluded only a limited amount of metadata, such as name of source file, as shown in Table 8). Week 2 served as an arbitration week. When members of the same team disagreed about a duplicate set, a senior attorney-editor not on that team would serve as an arbitrator or tie-breaker. In this way, a virtual voting system was established. Every result set would thus be reviewed by a minimum of two assessors, and sometimes by three. This approach was intended to produce dependable judgments from the process.

Type	Sample Instantiations
ClauseTitle	Section 7.9 WAIVER OF JURY TRIAL
DocTitle	Pledged Bonds Custody & Security Agreement
DMSFile	PledgeCustody.1March00.doc
ClauseTitle	(m) LEVERAGE RATIO
DocTitle	Letter of Credit & Reimbursement Agreement
DMSFile	NNZX18!.doc
ClauseTitle	6. Compensation of Escrow Agent
DocTitle	EXHIBIT C ESCROW AGREEMENT
DMSFile	0587980.doc

Table 8: Metadata Classifications for Clauses

To further help ensure judgment reliability and consistency, a training document was prepared for the assessors that included illustrations and detailed instructions. In addition, a preliminary training exercise was developed for each team that included real user query result sets and the opportunity for the participants to discuss their judgments as well as the granularity of their inspection. All of the assessors participated in the same initial training session and were asked to apply their knowledge to the same pair of sample result sets. Training guidelines were amended as a result of these sessions in order to clarify the level of granularity of analysis necessary for the task. In general, the assessors found these training cases quite instructive. As beneficial

⁹(a) I.e., 90% of the words in one text are contained in the other (in terms of overall *terminology* rather than individual term *frequency*).

(b) For texts that do not meet a working threshold for similarity or *resemblance*, Broder, et al. monitor a second looser relationship described as *containment* [3].

Assessor Pair	Editor–Editor	System–Editor–Arb.
Training (First 25 Queries)	$\kappa = 0.87^*$	$\kappa = 0.95$
Test (Second 25 Queries)	$\kappa = 0.92^+$	$\kappa = 0.94$
Combined (50 Qrys) (Train & Test)	$\kappa = 0.898$	$\kappa = 0.943$

Table 11: Kappa Statistics for Inter-assessor Agreements for Duplicate Set Identification (macro-averaged scores)

The above value of κ for the combined query set yields $z = 3.925$ (Team A, Queries 1-25)* and $z = 5.191$ (Team B, Queries 26-50).⁺ These values exceed the $\alpha = 0.001$ significance level (where $z = 3.090$). Therefore, we may conclude that the assessors exhibit significant agreement on this categorization task. It is important to note that these results were produced *before* we introduced the arbitration round, wherein another attorney-editor not on the team resolved differences in judgments between the two original assessors. Given a third expert casting a “vote” on these differences, the final duplication judgments are arguably more reliable than those examined during the Kappa analysis.

4. OVERVIEW OF INITIAL ALGORITHM

Sections 4 and 5 are included to examine the utility of the resulting transactional duplicate detection collection when designing, developing, and testing algorithmic approaches to deduplication of fuzzy duplicates.

Note that there have been efforts to completely automatically detect “redundancy” in result sets [31], but these appear to eliminate the role of the client and focus exclusively on mathematical models of content, even in highly dynamic retrieval environments. In order to determine our ability to identify and characterize such non-identical duplicate documents using the contributions from our client base, we began investigating reliance upon an expanded multi-dimensional feature set or “digital signature.” This feature set includes:

- magnitude component (doc_length);
- hash of the top N rarest terms and their locations (hash_key);
- core content component (term_vector).

The role of the first two is to provide heuristics to reduce the need for more costly term comparisons. They do not reduce the number of candidate pairs as much as reduce the search space for valid duplicate candidates. In addition to document length (excluding metadata) and top-term hash, a document’s term_vector is represented by its top n idf words, where n falls somewhere between 30 and 60 words. We determined empirically that 60 words would serve as an optimal default vector size for *documents* of moderate length, because (a) it offers substantially finer granularity to the process, and (b) it does not exceed the short length limits of the vast majority of such documents. For clause-level deduplication, however, where the texts can and are often considerably shorter, the lower bound of 30 terms was found to be more practical, but also invites an algorithmic means of smoothly accommodating still shorter length texts.

The percent overlap between two documents’ term_vectors served as our de facto similarity measure. In practice, once the heuristics completed their reduction of eligible candidate pairs, the algorithm then used as its matching criterion a

90% vector overlap.

Aside from core content from contract clauses, metadata indicating law firm, key indexing terms, source file, etc. (some shown in Table 8) is not used. We have determined that such supplemental content tends to increase the number of false positives, since related but dissimilar documents may possess similar metadata and classification terms.

It is worth noting that even though these metadata classification indexes are not considered part of the core document, they were not suppressed from our assessors (though the assessors were generally discouraged from using them in their determination of duplication status, because of the false positive risk discussed above). Nonetheless, in the comprehensive collection that resulted, these fields are still viewed as intrinsic to the corpus and are therefore retained.

5. COLLECTION DEPLOYMENT AND PERFORMANCE EVALUATION

5.1 Test Corpus and Algorithm Assessment

Figures 1 (a) and (b) and Table 12 show the performance of the algorithm outlined above relative to the gold data standard established by the attorney assessors, in terms of agreement (correct identification), false negatives (misses), and false positives (over-generation). An idf table constructed from a separate training collection of over 2 million documents is used to identify the rarest terms. A number of modifications were made to the algorithm during the training phase. Most notable is how it treats short texts (with fewer than 30 terms). A variety of options exist, including (i) comparing vectors of unequal length, (ii) comparing only the rarest n terms, where n is the size of the shortest text’s vector, and (iii) padding the short text’s term vector with entries not found in the table (in a manner that facilitates comparisons with similar docs). In the end, we found that amendments to the last approach yielded the best results.

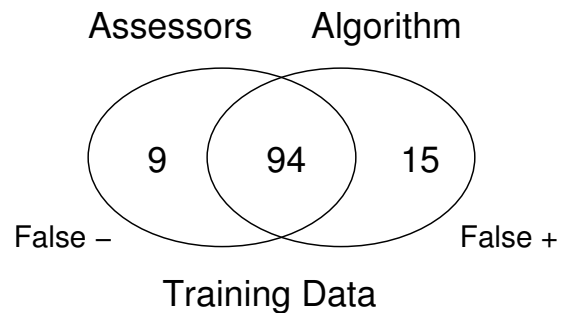


Figure 1 (a). Dup Sets Identified in Training Round

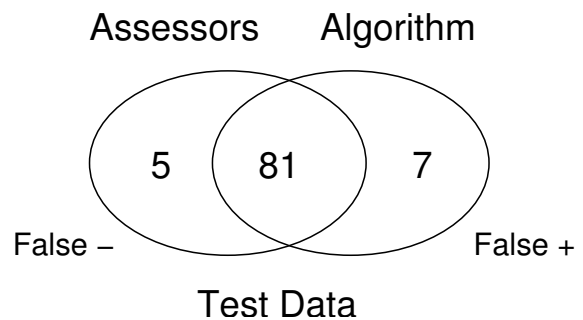


Figure 1 (b). Dup Sets Identified in Test Round

set reflect true accuracy on the complete set? Mitchell has addressed this problem in the context of Machine Learning [22]. For a collection C , $error_C$ can be defined as the ratio of false positives and false negatives in the algorithm's results on C . Our evaluation test set of sample S produces $error_S$. Mitchell assumes that the probability of having a specific ratio of errors (r) is approximated by a normally distributed random variable with a mean $error_S$ and standard deviation:

$$\sigma_{error_S} = \frac{\sigma_r}{|S|} \approx \sqrt{\frac{error_S(1 - error_S)}{|S|}}$$

where $|S|$ is the size of the sample. The true error can be viewed as drawing a bell curve that is centered on the observed error. So with probability $N\%$, $error_C$ is within z_N standard deviations of $error_S$, where z_N is the z-value. In our case, there is a 95% chance of $error_C$ is within 1.96 standard deviations of $error_S$. For instance, for an observed error ratio of 2.3% (22 errors among 942 document-clauses), there is a 95% chance that the error on the full collection is within the range $2.34\% \pm 0.49\%$. For 44 errors among 1,884 document-clauses, the interval would be $2.34\% \pm 0.35\%$. This analysis likely warrants further investigation since as one moves beyond consideration of a result set consisting of 20 documents, the number of pair-wise comparisons required per query increases exponentially. It would be instructive to determine whether this fanout has any appreciable impact on error rate. In subsequent tests on result sets consisting of approximately 1,000 documents coming from the business domain, we found no deviation in performance.

6. CONCLUSIONS

The accelerated growth of massive electronic data environments, both Web-based and proprietary, has expanded the need for various forms of duplicate document detection. Depending on the nature of the domain and its customary search paradigms, this detection can take any of several forms, but may be largely characterized by either identical or non-identical deduplication. Our own exploration addresses a real world replication problem occurring in the transactional law domain. We designed a methodology that invited our clients, both internal and external, to define the scope of the problem, and then commissioned pairs of professional legal assessors to use our working definition together with additional principled methods to construct a test collection in which non-identical duplicates are identified. We have also attempted to validate the decisions of our assessors using a follow-up Kappa analysis. For non-identical duplicate text detection, our applied test collection proved beneficial and the subsequent dedicated trials suggest that a multi-dimensional feature set approach to characterizing and comparing clause-level texts can provide a solid indication of the degree of duplication between two texts. The treatment of its multi-dimensional feature set frees it from reliance upon singular features and permits heuristics to save on more costly comparisons.

7. FUTURE WORK

In subsequent work, we plan to add a layer of Entity Recognition (ER) in order to address the "near identical" duplicates category. Such ER research would include the categorization of entity (e.g., party, organization, location,

financial amount, etc.) as well as whether two entities would warrant resolution to a single canonical form. Once such a follow-up process were added, we would be better able to improve the granularity of our existing evaluation measures.

8. ACKNOWLEDGMENTS

We appreciate the duplicate assessment efforts of Rodney Brown, Scott Ratcliffe, Cara Cardinale, Frank Wozniak, Yasmin Alexander, Joanne Rhoton, Lora Thody, Bill Bremer, Elizabeth Randisi, Stephanie Harth, Kevin Duerinck and Lisa Kless. We thank Ely Razin and Kingsley Martin for their invaluable contribution of domain expertise. We are also grateful to George May, Anudeep Parhar and Matt Canavan who supported our non-identical duplicate research. And lastly, we acknowledge the assistance of Bart Matzek and Doug Heger in handling computability and real-time processing issues in the production environment.

9. REFERENCES

- [1] Sergey Brin, James Davis, and Héctor García-Molina. Copy detection mechanisms for digital documents. In *Proceedings of the Special Interest Group on Management of Data (SIGMOD '95) (San Francisco, CA)*, pages 398–409. ACM Press, May 1995.
- [2] Sergey Brin and Larry Page. The anatomy of a large-scale hypertextual Web search engine. In *Proceedings of the Seventh Int'l World Wide Web Conference (WWW7 '98) (Brisbane, Australia)*, pages 107–117. Elsevier Science, April 1998.
- [3] Andrei Z. Broder, Steven C. Glassman, Mark S. Manasse, and Geoffrey Zweig. Syntactic clustering of the Web. In *Proceedings of the Sixth Int'l World Wide Web Conference (WWW6 '97) (Santa Clara, California)*, pages 391–404. Elsevier Science, April 1997.
- [4] Robert Burgin. Variations in relevance judgments and the evaluation of retrieval performance. *Information Processing and Management*, 28(5):619–627, 1992.
- [5] Jean Carletta. Assessing agreement on classification tasks: The kappa statistic. *Computational Linguistics*, 22(2):249–254, 1996.
- [6] Surajit Chaudhuri, Venkatesh Ganti, and Rajeev Motwani. Robust identification of fuzzy duplicates. In *Proceedings of the 21st International Conference on Data Engineering (ICDE05)*, page 12, Tokyo, Japan, April 2005. IEEE Computer Society.
- [7] Abdur Chowdhury, Ophir Frieder, David Grossman, and Mary Catherine McCabe. Collection statistics for fast duplicate document detection. *ACM Transactions on Information Systems (TOIS)*, 20(2):171–191, April 2002.
- [8] Jack G. Conrad, Xu S. Guo, and Cindy P. Schriber. Online duplicate document detection: Signature reliability in a dynamic retrieval environment. In *Proceedings of the 12th Int'l Conference on Information and Knowledge Management (CIKM'03) (New Orleans, LA)*, pages 443–452. ACM Press, Nov. 2003.
- [9] Jack G. Conrad and Cindy P. Schriber. Managing déjà vu: Collection building for identifying non-identical duplicate documents. In *Journal of the American*

