IJCSE International Journal of Computer Sciences and Engineering Open Access
Survey Paper Vol.-7, Special Issue, 4, Feb 2019 E-ISSN: 2347-2693

An Efficient Duplicate Detection Algorithm Using Data Cleansing

J. Selvi[1*], R. Gayathri[2]

[1,2]M.Sc Computer Science, Idhaya College for Women, Kumbakonam, Tamilnadu, India

Corresponding Author: selvi12@gmail.com

Available online at: www.ijcseonline.org

***Abstract*—**The aim of the technique is to minimize the data duplication in the web mining patterns during the time of web based search in large data mining applications. Although there is a long line of work on identifying duplicates in relational data, only a few solutions focus on duplicate detection in more complex hierarchical structures, like XML data. In this system present a novel method for XML duplicate detection, called XML Dup. XML Dup uses a Bayesian network to determine the probability of two XML elements being duplicates, considering not only the information within the elements, but also the way that information is structured. In addition, to improve the efficiency of the network evaluation, a novel pruning strategy, capable of significant gains over the un optimized version of the algorithm, is presented. Through experiments, we show that our algorithm is able to achieve high precision and recall scores in several data sets. XML Dup is also able to outperform another state-of-the-art duplicate detection solution, both in terms of efficiency and of effectiveness.

***Keywords*—**Duplicate Detection, Network Evaluation, Efficiency, Effectiveness.

I. INTRODUCTION

Web search is one of the most prominent Information Retrieval (IR) applications. Typical question-answering scenarios are well supported by ranking highly the documents that not only look relevant by their content, but also receive external support such as by incoming links and anchor text references. In these applications, looking at one or a few of the highest ranked result documents might be sufficient, and if it is, the search process can be stopped. Commercial web search engines are optimized for this scenario and much IR research is focused on improving performance in the top, say 10, results. However, if the objective is to carry out a comprehensive review for a particular topic, search cannot be stopped after finding a few relevant documents. In particular, reviews aim for very broad coverage of a topic, and seek to minimize any bias that might arise as a result of missed or excluded relevant literature. But the typical tensions in IR continue to apply, and if more relevant documents are to be found, more irrelevant documents will also need to be inspected. In the biomedical domain, systematic reviews of the whole corpus of published research literature (the largest collection, MEDLINE, currently indexes more than 17 million publications) are used to provide medical practitioners with advice to assist their case by case decision-making. To seed the reviews, complex Boolean queries are used on different citation databases to generate a set of documents which are then triaged by multiple assessors. In this domain, it becomes crucial to find as much of the relevant literature as possible for any given level of effort, because each item of overlooked evidence adds to the possibility of suboptimal outcomes in terms of patients' health-care. The interest for information retrieval has existed long before the Internet. The Boolean retrieval is the most simple of these retrieval methods and relies on the use of Boolean operators. The terms in a query are linked together with AND, OR and NOT. This method is often used in search engines on the Internet because it is fast and can therefore be used online. This method has also its problems. The user has to have some knowledge to the search topic for the search to be efficient, e.g., a wrong word in a query could rank a relevant document non relevant. The retrieved documents are all equally ranked with respect to relevance and the number of retrieved documents can only be changed by reformulating the query. The Boolean retrieval has been extended and refined to solve these problems. Expanded term weighting operations make ranking of documents possible, where the terms in the document could be weighted according to their frequency in the document. Boolean information retrieval has been combined with content-based navigation using concept lattices, where shared terms from previously attained documents are used to refine and expand the query. The Boolean operators have been replaced with fuzzy operators. Weighted query expansion using a thesaurus. A model based on fuzzy set theory allows the interpretation of a user query with a linguistic descriptor for each term. The traditional Boolean retrieval model has been studied intensively in IR research. While it has straightforward semantics, it also has a number of disadvantages, most notably the strictly binary categorization

of documents, and the consequent inability to control the result set size except by adding or removing query terms. For example, it is often the case that too many, or too few, or even no documents are returned, and no matter how the query terms are juggled, the "Goldilocks" point might be impossible to attain. In contrast, the broad adoption of ranking principles based on bag-of-word queries, and the resultant ability to order the set of documents according to a heuristic similarity score, means that for general IR applications users can consciously choose how many documents they are willing or able to inspect. Now the drawback is that bag-of-word keyword queries do not offer the same expressive power as Boolean queries do. Although extensions to the Boolean retrieval system have been suggested that produce a ranked output based on Boolean query specifications, they have not been broadly adopted for practical use – perhaps because, to date, simple keyword queries have typically been able to produce similar results, and, for lay users, are easier to generate. Although ranking has the advantage of identifying a monotonically increasing total number of relevant documents as more documents are inspected, typical IR ranking functions face the difficulty that their ranking is dependent on properties of the whole collection, and can thus be difficult to reproduce, or even understand. Reproducibility helps in assessing review quality, and is thus often stipulated as a key requirement of comprehensive reviews. But if ranked queries are used, reproducibility can only be assured if all aspects of the computation are reported, including term weights and within-document term frequencies. With Boolean queries, all that is required is publication of the query that was used, together with the date or other identifying version numbers of the collections it was applied to. Moreover, previous work did not show improved retrieval results with ranked keyword queries compared to complex Boolean queries.

Advantages of Boolean retrieval

- Complex information need descriptions: Boolean queries can be used to express complex concepts;
- Compensability and Reuse: Boolean filters and concepts can be recombined into larger query tree structures;
- Reproducibility: Scoring of a document only depends on the document itself, not statistics of the whole collection, and can be reproduced with knowledge of the query;
- Scrutability: Properties of retrieved documents can be understood simply by inspection of the query;
- Strictness: Strict inclusion and exclusion criteria are inherently supported, for instance, based on metadata.

II. RELATED WORK

Extended Boolean Models

Extended Boolean models can supports document ranking facility for the conventional Boolean retrieval system by calculating the similarities between documents and Boolean queries. An IR system based on extended Boolean models can be defined by the quadruple <T, D, Q, F>, Where

- T is a set of index terms used to represent queries and documents.
- D is a set of documents. Each document d ε D is represented by $\{(t_1,w_1),\ldots\ldots(t_n,w_n)\}$ where w_i designates the weights of term t_i in document d and wi may take any value between zero and one i.e $0<w_i<1$.
- Q is a set of queries that can be recognized by the system. Each query q ε Q is a legitimate Boolean expression composed of index terms and Boolean operators AND, OR and NOT. In some extended Boolean models, queries can be also formulated with term and clause weights.
- F is a ranking function

$$F : D * Q \rightarrow [0,1]$$

Which assign to each pair (d,q) a number in the closed interval [0,1]. This number is a measure of similarity between document d and query q is called the document value for document d with respect to query q. The retrieval function F is defined as follows:

1. For each term t_i in query q the function $F(d,t_i)$ is defined as the weight of term t_i in document d, i.e. w_i.
2. Boolean operators, i.e AND, OR and NOT are then evaluated by applying the corresponding formulas. The evaluation formulas of the operators are an important factor to determine the quality of ranked output. F(d,NOTti) is evaluated as 1-wi. For Boolean queries containing more than one Boolean operator, the evaluation proceeds recursively from the innermost clause.

Conventional Retrieval Strategies

In conventional information retrieval, the stored records are normally identified by sets of key words or index terms, and requests for information are expressed by using Boolean combinations of index terms. The retrieval strategy is normally based on an auxiliary inverted-term index that lists the corresponding set of document references for each allowable index term. The Boolean retrieval system is designed to retrieve all stored records exhibiting the precise combination of key words included in the query: when two query terms are related by an and connective, both terms must be present in order to retrieve a particular stored record; when an or connective is used, at least one of the query terms must be present to retrieve a particular item. In some systems where the natural language text of the documents or the document excerpts is stored, the user queries may be formulated as combinations of text words. In that case, the queries may include location restrictions for the query terms--for example, a requirement that the query terms occur in the same sentence of any retrieved document or within some specified number of words of each other.

Boolean retrieval systems have become popular in operational situations because high standards of performance

are achievable. The retrieval technology which is based on list intersections and list unions to implement Boolean conjunction ("A and B") and Boolean disjunction ("A **or** B"), respectively, is now well understood. The conventional Boolean retrieval technology is however also saddled with various disadvantages:

1. The size of the output obtained in response to a given query is difficult to control; depending on the assignment frequency of the query terms and the actual term combinations used in a query formulation, a great deal of output can be obtained or, alternatively, no output might be retrieved at all.

2. The output obtained in response to a query is not ranked in any order of presumed importance to the user; thus, each retrieved item is assumed to be as important as any other retrieved item.

3. No provisions are made for assigning importance factors or weights to the terms attached either to the documents.

III. METHODOLOGY

The proposed system is Extended Boolean retrieval (EBR) model for retrieving the top k documents.We present a scoring method for EBR models thatdecouples document scoring from the inverted list evaluation strategy, allowing free optimization of the latter. The method incurs partial sorting overhead, but, at the same time, reduces the number of query nodes that have to be considered in order to score a document. We adopt ideas from the max-score and wand algorithms and generalize them to be applicable in the context of models with hierarchical query specifications and monotonic score aggregation functions. Further, we show that the p-norm EBR model is an instance of such models and that performance gains can be attained that are similar to the ones available when evaluating ranked queries. Term-independent bounds are proposed, which complement the bounds obtained from max-score. Taken alone, term-independent bounds can be employed in the wand algorithm, also reducing the number of score evaluations. Further, in conjunction with the adaption of max-score, this novel heuristic is able to short-circuit the scoring of documents.

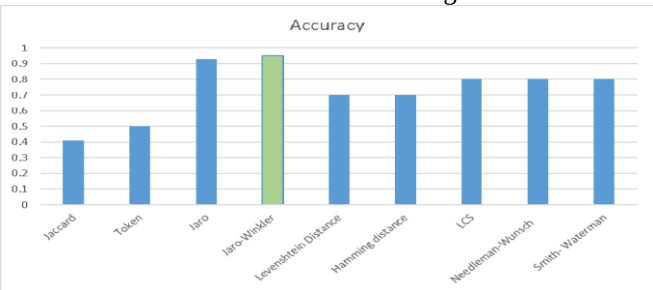

FIGURE 1: PERFORMANCE OF THE SIMILARITY MEASUREMENTS

IV. CONCLUSION

A comparative analysis was performed in this study based on the most popular similarity measurements and approaches that exist in the literature for duplication detection of records in databases. The strengths associated with each of the approaches with regards to similarity measurement are discussed and examined for performance accuracy, the speed of execution and computational time required to process duplications. The experiment demonstrated that Jaro-Winkler similarity measurement outperformed all other popular similarity measurement approaches and techniques. Furthermore, the evaluation of the transposition step in the Jaro-Winkler approach was found to be more convenient and practical in achieving further improvement in detecting similarity measurements of records contained in the database applying this algorithm. The study has motivated our intention to implement this algorithm.

REFERENCES

[1] S. R. Alenazi and Kamsuriah, "Record Duplication Detection in Database: A Review," Int. J. Adv. Sci. Eng. Inf. Technol., vol. 6, no. 6, pp. 838–845, 2016.

[2] F. N. Mahmood and A. Ismail, "Semantic Similarity Measurement Methods: The State-of-the-art," Res. J. Appl. Sci. Eng. Technol., vol. 8, no. 18, p. 1923–1932., 2014.

[3] A. Osama, Helmi, "A Comparative Study of Duplicate Record Detection Techniques," Middle East, 2012.

[4] D. Vatsalan and P. Christen, "Privacy-Preserving Matching Of Similar Patients," J. Biomed. Inform., vol. 59, pp. 285–298, 2016.

[5] Christenp and Timc, "Freely Extensible Biomedical Record Linkage," 2013. [Online]. Available: https://sourceforge.net/projects/febrl/.

[6] M. G. Elfeky, V. S. Verykios, and A. K. Elmagarmid, "TAILOR: A Record Linkage Toolbox," Proc. 18th Int. Conf. Data Eng., pp. 17–28, 2002.

[7] W. E. Yancey, "Big Match: A Program For Extracting Probably Matches From A Large File For Record Linkage," Computing, vol. 1, no. 1, pp. 1– 8, 2002.

[8] A. K. Elmagarmid, P. G. Ipeirotis, and V. S. Verykios, "Duplicate Record Detection: A Survey," IEEE Trans. Knowl. Data Eng., vol. 19, no. 1, pp. 1–16, Jan. 2007.

[9] W. H.Gomaa and A. A. Fahmy, "A Survey of Text Similarity Approaches," Int. J. Comput. Appl., vol. 68, no. 13, pp. 13–18, Apr. 2013.

[10] R. T. Nakatsu and E. B. Grossman, "A Task-Fit Model of Crowdsourcing□: Finding the Right Crowdsourcing Approach to Fit the Task," J. Inf. Sci., pp. 1–11, 2014.

[11] "SemEval-2015 The 9th International Workshop on Semantic Evaluation," New York 12571 USA, 2015.

[12] Nirmalrani V, E. P. Sim, and Arun PR, "Detection of near duplicate web pages using four stage algorithm," in 2015 International Conference onCommunications and Signal Processing (ICCSP), 2015, pp. 0644–0648.

[13] Y. Jiang, G. Li, J. Feng, and W. Li, "String Similarity Joins□: An Experimental Evaluation," Vldb, pp. 625–636, 2014.

[14] P. A. V. Hall and G. R. Dowling, "Approximate String Matching," ACM Comput.Surv., vol. 12, no. 4, pp. 381–402, 1980.

[15] J. L. Peterson, "Computer Programs For Detecting And Correcting Spelling Errors," Commun. ACM, vol. 23, no. 12, pp. 676–687, Dec.1980.
[16] V. Wandhekar, "Validation of Deduplication in Data using Similarity Measure," Int. J. Comput.Appl., vol. 116, no. 21, pp. 18–22, 2015.
[17] K. Williams and C. L. Giles, "Near Duplicate Detection In An Academic Digital Library," Proc. 2013 ACM Symp. Doc. Eng. - DocEng '13, pp. 91–94, 2013.
[18] A. Z. Broder, S. C. Glassman, M. S. Manasse, and G. Zweig, "Syntactic Clustering of the Web," Comput.Networks ISDN Syst., vol. 29, no. 8, pp. 1157–1166, 1997.
[19] K. Dreßler and A.-C. N. Ngomo., "On the Efficient Execution of Bounded Jaro-Winkler Distances," Semant. Web 8, vol. 0, no. 0, pp. 1–13, 2017.
[20] S. B. Needleman and C. D. Wunsch, "A General Method Applicable To The Search For Similarities In The Amino Acid Sequence Of Two Proteins," J. Mol. Biol., vol. 48, no. 3, pp. 443–453, 1970.
[21] T. F. Smith and M. S. Waterman, "Identification Of Common Molecular Subsequences," J. Mol. Biol., vol. 147, no. 1, pp. 195–197, Mar. 1981.

