Data Duplication Detection in Publication Paper Using Jaccard Indexing Algorithm

M. Pavithra[#1], Mrs. P. Shanthi[#2]
[#1]*Research Scholar, Sri Jayendra Saraswathy Maha Vidhyalaya College of arts and science, Coimbatore, INDIA*
[#2]*Associate Professor, Sri Jayendra Saraswathy Maha Vidhyalaya College of arts and science Coimbatore, INDIA*
pavihema93@gmail.com

Abstract:**Duplicate Detection is a desperate task in any database of any organization. Duplicates are nothing but the same real time entities or objects presented in the form of different structure and in the different formats. The Jaccard Indexing algorithms find out the duplicates in relational data, in complex data and hierarchical data. There are lots of works already presented in the past for finding the duplicates in the relational data. In this process, one record is compared to all other records. Different data representations, formats, terminologies and data entry mistakes make this task complex. Involvement of heavy volume databases adds more complexity. To address the problem of record comparable in such database scenario present a Jaccard Indexing Algorithm, for a given query the algorithm can effectively identify duplicates from the query result records of multiple datasets. The proposed algorithm checks the duplication in the publication paper based on author, publication year and paper name etc. This Innovative algorithm that assumes detected duplicate in sorted dataset raises the probability of finding more duplicates in neighborhood. Series of consecutive non-duplicates drops the possibility of duplicates in neighborhood. Using this concept, it adapts window both for duplicates and non-duplicates (0 or 1) and avoids unnecessary comparisons without losing effectiveness. It proves that Jaccard Indexing Algorithm is a better alternative in windowing algorithms.**

Keywords: **Duplication, Neighboring method, Non neighboring method, Jaccard Indexing Algorithm.**

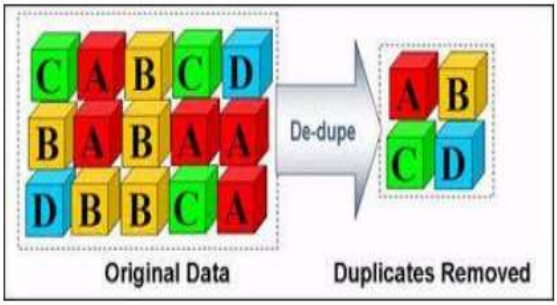
1. INTRODUTION

Data de-duplication is a kind of technology which can identify the duplicate and delete duplicate data from data set. Data duplication detection in publication paper using Jaccard Indexing algorithm is to identify duplicate values from different data records. Check the duplication in the publication papers based on author name. The publication papers contain author name, year, title of the paper and venue. Select some papers from the CORA data set and compare all papers using Jaccard Indexing algorithm. This algorithm also contains neighboring method. The publication papers sometimes contain duplicates, Jaccard Indexing algorithm check the papers and assign to one(1) value to duplicates otherwise zero(0). The neighboring method is used to check the nearest data and compare the selected data and illustrate duplicates and one is indicate duplicates and zero as non-duplicates. The non-zero neighbor is evaluating the duplicates values only. It's easy to identify the duplicates.

Various methods were introduced and discussed in the previous research works in terms of obtaining the duplicates from different sources or data sets. Mainly two types of approaches are introduces in the previous discussion. They are

- Duplicate Record Detection For Database Cleansing
- Duplicate Record Detection:survey

Duplicate Record Detection For Database Cleaningwhich arises when the data is collected from various sources and find out the duplicate usingadaptive duplicate detection algorithm is the optimal solution for the problem of duplicate record detection.



The main contribution of this research work is given as follows:

- Identity the duplicates from the publication paper using Jaccard algorithm.

- Jaccard Indexing algorithm is used to finding the duplicates accurately and time consuming is very low.it also contain neighboring method the nearest neighbor analysis is that it does not discriminate between scales. It is quite common that points are clustered at small scales.

The organization of this work is given as follows: In section 2, an overview of the previous research works is given. In section 3, proposed methodology of this work is tagged with the detailed flow. In section 4, result and discussion are verified in this part.In section 6,overall conclusion and future scope of this research work is elucidated.

2. RELATED WORKS

In this section, various previous research methodologies that have been evaluated to achieve the efficient detection of identity deception present in network environments is illustrated. They are as follows:

M.Rehman, V.Esichaikul [1] proposed Duplicate Record Detection For Database Cleaning contain adaptive duplicate detection algorithm is the optimal solution for the problem for approximate matching of data records, string matching algorithm.

L. Gu and R. Baxter[2]proposed an adaptive prefetch filtering (APF) mechanism to reduce the wasted bandwidth and energy as well as the cache pollution caused by useless prefetches.

A. Elmagarmid, P. Ipeirotis, and V. Verykios[3] present a thorough analysis of the literature on duplicate record detection. We cover similarity metrics that are commonly used to detect similar field entries.

M.Bilenko, B.Kamath, R.Mooney[5] introduced an adaptive framework for automatically learning blocking functions that are efficient and accurate effectiveness on real and simulated datasets.

V. Wandhekar, A. Mohanpurkar[4] present a novel method for XML duplicate detection, called XMLDup. XMLDup uses a Bayesian network to establish the probability of two XML elements being duplicates, considering not only the information within the elements, but also the way that information is structured.

U. Draisbach and F. Naumann[8] present a new algorithm called Sorted Blocks in several variants, which generalizes both approaches, Blocking methods partition records into disjoint subsets, while windowing methods, in particular the Sorted Neighbohood Method, slide a window over the sorted records and compare records only within the window. An advantage of Sorted Blocks in comparison to the Sorted Neighborhood Method is the variable partition size instead of a fixed size window.

U. Draisbach, F. Naumann, S. Szott, and O. Wonneberg[6] introduced some similarity measure must be defined to compare pairs of records and data sets might have a high volume making a pair-wise comparison of all records infeasible.

J. Nin, V. Mulero, N.Bazan, Josep-L. L.Pey,[7] we show that exploiting the relationships (e.g. foreign key) established between one or more data sources. Propose a new blocking method that builds groups of records based on the connection among them in the data sources, as opposed to the use of the syntactic information of its attributes, as in other classic methods.

3. JACCARD INDEXING ALGORITHM

The proposed, Jaccard Indexing Algorithm is used to detect duplicate records. The paper builds up on the idea of jaccard indexing and is an attempt to enhance the algorithm by introducing an efficient classifier to improve the accuracy rate. Jaccard algorithm assumed that there will be no duplicates within a database and only considered the result set from multiple databases for potential duplicates. This technique will classify the data's in two category duplicate and non-duplicate. The value 0 indicates the non-duplicate record and 1 indicate a duplicate record. The introduction of the jaccard indexing algorithm helps to identifying the duplicates that are present within the database. In the proposed system the plan was to develop an algorithm that uses three classifiers for detecting duplicate records.

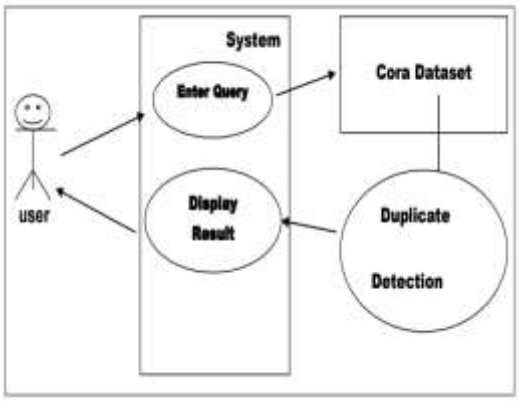


Fig 1: Flow of Duplicate Detection

Proposed system has extended work done by adding jaccard algorithm to improve the efficiency of the duplicate evaluation.

Neighbor value Calculation

The problem with nearest neighbor analysis is that it does not separate between scales. It is quite common

that points are clustered at small scales, but that these clusters are themselves over dispersed, or, conversely, that the points avoid each other locally but cluster at larger scales.

Jaccard Match Similarity Score Calculation

The distance from the hyper plane provides a measure of confidence in the pair of records being a duplicate; it can be distorted to an actual similarity value using this process of computing record similarity using multiple similarity measures over each field and a binary classifier to categorize the resulting feature vector as belonging to the class of duplicates or non-duplicates, resulting in a distance estimate. For each field of the database, two learnable distance measures, d1 and d2, are trained and used to calculate similarity for that field. The values computed by these measures form the feature vector that is then classified by a support vector machine, producing a confidence value that represents similarity between the database records.

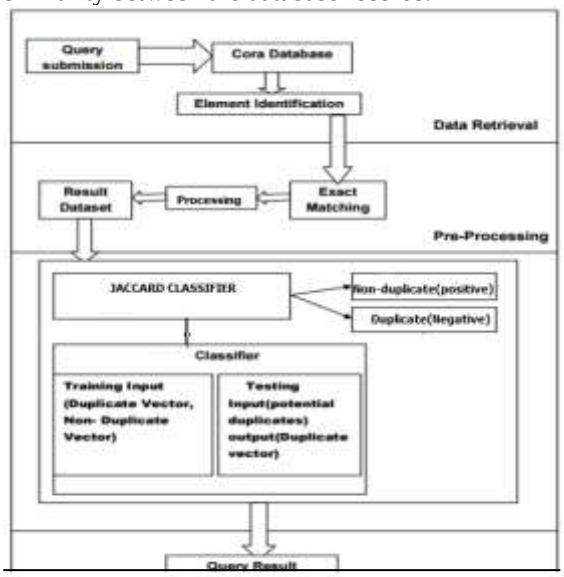


Fig 2: Architecture of duplicate detection process

Methodology

The Jaccard indexing algorithm is also known as the Jaccard similarity coefficient (originally coined coefficient denominate by Paul Jaccard), is a statistic used for comparing the similarity and diversity of sample sets.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}.$$

(If A and B are both empty, we define $J(A,B) = 1$.)

$$0 \leq J(A,B) \leq 1.$$

The **Jaccard distance**, which measures *dis*similarity between sample sets, is complementary to the Jaccard coefficient and is obtained by subtracting the Jaccard coefficient from 1, or, evenly, by dividing the difference of the sizes of the union and the intersection of two sets by the size of the union:

$$d_J(A,B) = 1 - J(A,B) = \frac{|A \cup B| - |A \cap B|}{|A \cup B|}.$$

Algorithm

To compute the Jaccard similarity coefficient for a pair of X elements, calculation of two quantities is needed.

Pass One XY

Identity Map This map method simply takes in a key/value pair and returns them unchanged.

Input: T ext, T ext

Output: T ext, T ext

X, Y → X, Y

Pass Two XY to YX

This map method takes the key/value pair and switches the String in the key with the String in the value. It then returns the modified key and value. Using the Y element as the key allows the combine stage to collect all of the X elements that correspond to each Y element. This enables the later collection of all pairs of X that occur with a single Y .

Input: Text, CPair

Output: Text, CPair X,(Y, C) → Y ,(X, C)

Pass Three

Pair Collect Map

This map takes in the empty Text key and the list of values. It then creates a series of Xi ,Xj pairs with the first element in the value list and each of the following elements. A list of size n will produce n-1 such pairs, one for each element in the list, except for the first.

Input: Text, {CPair}

Output: Pair, IntWritable

$$\underline{R}, \{(X,C)\} \rightarrow \underline{(X_i, X_j)}, C_{ij} \quad where\ C_{ij} = C_i + C_j$$

Algorithm steps

Input : Dataset→ Training Dataset(T1), Training Dataset(T2)

Output : Matching Result (j value)

Read Dataset(T1, T2)

For(i=0; i<T1.Length; i++)

For(j=0; j<T2.length; j++)

Jvalue (i, j)

End For

End For

Match Value (jvalue);

Find matching string

Publication id	Author id	Author Name	Title	Publications	Year
asfahl1992a	1	C.RayAsfahl	Robots and manufacturing Automation		1992
asfahl1992a	1	Ray Asfahl. C	Robots and manufacturing Automation		1992
benford1993a	53	Steve Benford and Lennart E. Fahlen.	A Spatial model of interaction in large virtual environments	In Proceedings of ECSCW'93	1993
buth1992a	162	B.Buth	Provably correct compiler implementation	Compiler Construction	1992
buth1992a	162	B.Buth et. al.	Provably correct compiler implementation	Compiler Construction	1992

Table 1: Table of sample attribute of Cora.

The Cora dataset Riddle dataset consists of duplicate and non-duplicates data records and the Cora data includes attributes. The experimentation starts from selecting the datasets as the input of the similarity computation by the similarity computation factors, listed in the above sections, such as distance method and Jaccard Match Similarity method. The similarity factors produce feature vectors on regard with the elements in the dataset. The feature vectors produced are represented with variables. The expressions are created from the feature vectors produced by the similarity vectors. The dataset is processed with the algorithms for a number of iterations and the best results of each algorithm for the particular iteration.

This section provides a comparative analysis of the proposed algorithm with the Jaccard Indexing algorithm. The analysis is based on accuracy, throughput and the time for execution of the algorithm. The comparative study represents the responses of the proposed Jaccard Match similarity and concept similarity measure with different datasets, namely **CORA** Database.

A. Accuracy

Accuracy is defined as the amount of correct predictions of duplication detection in Data duplication detection in publication papers. The accuracy of the proposed research work should be more than the existing work for the better system performance improvement. The following Table 2 depicts the values that are obtained while evaluating the proposed and existing mechanism.

Technique	No of records			
	200	400	600	800
NAÏVE	84%	80%	78%	76%
BOM	86%	83%	81%	78%
JACCARD	96%	94%	92%	89%

Table 2: Accuracy of algorithm

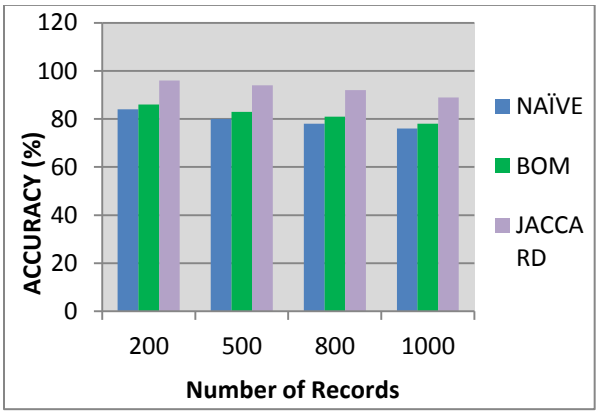


Fig 3: Accuracy result chart

B. Throughput

Throughput is defined as the correct duplications that are identifying among the total number of predictions done by the algorithm. Throughput of the proposed research work should be more than the existing scenario for better performance. The following Table 3 depicts the values that are obtained while evaluating the proposed and existing mechanism.

Technique	No of records			
	200	400	600	800
NAÏVE	80%	85%	83%	78%
BOM	83%	89%	85%	81%
JACCARD	96%	93%	91%	89%

Table 3: Throughput values

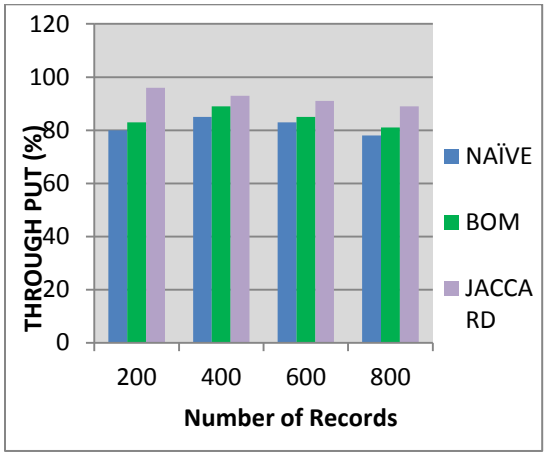


Fig 4: Throughput result chart

C. Time Complexity

Execution time is defined as the total time consumed for detection of the duplicates from the publication papers. The duplication detection time should be less in the proposed work than the existing scenario. The following Table 4 depicts the values that are obtained while evaluating the proposed and existing mechanism.

Technique	No of records			
	200	400	600	800
NAÏVE	3.5s	5s	5.5s	5.8s
BOM	3s	4.2s	4.8s	5s
JACCARD	2s	3s	3.9s	4.2s

Table 4: Processing Time

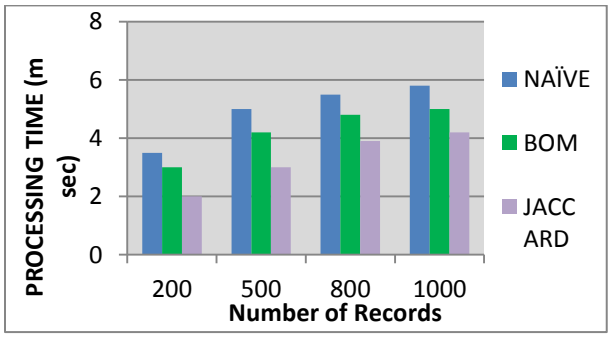


Fig 5: Processing Time Result Chart

The analysis is based on accuracy, throughput and the time for execution of the algorithm. The comparative study represents the responses of the proposed Jaccard Match similarity and concept similarity measure with different datasets, namely CORA Dataset.

5. CONCLUSION AND FUTURE SCOPE

Detection of duplicate records is an important problem in data management system. There are various techniques for record matching is explained with their advantages and disadvantages. The proposed Jaccard Indexing Algorithm is used to detect duplicate records. The paper builds up on the idea of jaccard indexing and is an attempt to enhance the algorithm by introducing an efficient classifier to improve the accuracy rate. Jaccard indexingalgorithm assumed that there will be no duplicates within a database and only considered the result set from multiple databases for potential duplicates. This technique will classify the data's in two category duplicate and non-duplicate. The value 0 indicates the non-duplicate record and 1 indicate a duplicate record. The introduction of the Jaccard indexing algorithm helps in identifying the duplicates that are present within the database. In the proposed system the plan was to develop an algorithm that uses three classifiers for detecting duplicate records.The experimental result shows that our approach is comparable to previous work and the result shows that our algorithm requires less time than Jaccard algorithm to find out the duplicates. As future work we planned to introduce a new algorithm for finding the duplicate records by using Optimization techniques. This is used for development of robust and scalable solutions. More research is needed in the area of data cleaning.

REFERENCES

[1] M.Rehman, V.Esichaikul, "Duplicate Record Detection For Database Cleansing", Second International Conference on Machine Vision, 2009.

[2] L. Gu and R. Baxter, "Adaptive filtering for efficient record linkage," in Proceedings of the SIAM International Conference on Data Mining, 2004, pp. 477–481

[3] A. Elmagarmid, P. Ipeirotis, and V. Verykios, "Duplicate record detection: A survey", IEEE Transactions on Knowledge and Data Engineering (TKDE), 2007, pp: 1-16.

[4] V. Wandhekar, A. Mohanpurkar, "A Review on Efficient and Effective Duplicate Detection in Data", International Journal for Research in Applied Science and Engineering Technology (IJRASET), ISSN: 232-9653, Volume 2 Issue XI, November 2014,pp: 103-107.

[5] M.Bilenko, B.Kamath, R.Mooney, "Adaptive Blocking: Learning to Scale Up Record Linkage", In Proceedings of

the Sixth IEEE International Conference on Data Mining (ICDM-06), Hong Kong, December 2006, pp. 87-96.
[6] U. Draisbach, F. Naumann, S. Szott, and O. Wonneberg, "Adaptive windows for duplicate detection," ACM SIGKDD interntional conference on Knowledge discover and data mining, NY, USA, 2011
[7] J. Nin, V. Mulero, N.Bazan, Josep-L.L.Pey, "On the Use of Semantic Blocking Techniques for Data Cleansing and Integration",11th International Database Engineering and Applications Symposium, 2007.
[8] U. Draisbach and F. Naumann, "A comparison and generalization of blocking and windowing algorithms for duplicate detection," in Proceedings of the International Workshop on Quality in Databases (QDB),2009.

