RESEARCH ARTICLE OPEN ACCESS

# A Survey Paper on Deduplication by Using Genetic Algorithm Alongwith Hash-Based Algorithm

Miss. J. R. Waykole*, Prof. S. M. Shinde**
*(PG Student, Pune University)
** (Associate Professor, Department of Computer Engineering, Pune University)

**ABSTRACT**
In today's world, by increasing the volume of information available in digital libraries, most of the system may be affected by the existence of replicas in their warehouses. This is due to the fact that, clean and replica-free warehouse not only allow the retrieval of information which is of higher quality but also lead to more concise data and reduces computational time and resources to process this data. Here, we propose a genetic programming approach along with hash-based similarity i.e, with MD5 and SHA-1 algorithm. This approach removes the replicas data and finds the optimization solution to deduplication of records.
***Keywords*** **-** Database administration, evolutionary computing algorithm i.e, genetic algorithm and MD5 and SHA-1 algorithm.

## I. INTRODUCTION

Today by increasing the volume of information created problem for duplicate records as we are collecting data from heterogeneous sources. So, finding duplicate records in those records collected from several sources are increasingly important tasks because to find the data from that sources is time consuming process and more resources are required. Record linkage and deduplication can be used to improve the data quality and integrity which in turn reduce costs and efforts in obtaining data.

In a data repository, a record that refers to the same real world entity or object is referred as duplicate records. And that duplicate record is also called as 'dirty data'. Due to this dirty data in warehouse many problem are occurred as follows:

1) Performance degradation—as additional useless data demand more processing and more time is required to answer simple queries.
2) Quality loss—the presence of replicas and other inconsistencies leads to distortions in reports and misleading conclusions based on the existing data.
3) Increased cost —because of the additional volume of useless data, investments are required on more storage media and extra computational processing power to keep the response time levels acceptable.

To avoid these problems, it is necessary to study the causes of "dirty" data in repositories. A major cause is the presence of duplicates in these repositories or warehouse is the aggregation or integration of distinct data sources. The problem of detecting and removing these duplicate records from a repository is known as record deduplication[1]. It is also referred as data cleaning [2], record linkage [1] and record matching [3].

"Data deduplication" is a data compression technique made possible by the invention in the 1980s of message digest hashes that create a "signature" for a block or file. If two signatures or hashes are equal or matched, then their corresponding blocks are considered to be equal. The second method for data deduplication is the grain of deduplication and the strategy for breaking large data sets (e.g. streams, files) into smaller chunks. New strategies for dividing a file into smaller chunks has been the focus of innovation in data deduplication over the past decade.

Deduplication is a key operation in integrating data from heterogeneous sources. The main challenge in this task is designing a function that can resolve when a pair of records refers to the same entity in spite of various data inconsistencies. Deduplication reduces the amount of storing data by eliminating redundant copy of data. Data is deduplicated as it is written, and it reduces the space for storing more and more data. Deduplication can be applied to data in primary storage, backup storage, cloud storage or data in flight for replication, such as LAN and WAN transfers.

The rest of this paper is organized as follows. In Section 2, we discuss related work i.e, literature survey. In Section 3, we present some genetic programming (GP) basic concepts. In Section 4, we describe how we can use the concept of hash algorithm for finding duplicates and by using genetic programming we can remove duplicate records. Finally, in Section 5 we present our conclusion and comment on future work.

## II. LITERATURE SURVEY

Record deduplication is a growing topic in databases as many duplicates are exist in repositories

or warehouses. This problem is arises mainly when collecting data from heterogeneous sources. To solve these inconsistencies, it is necessary to design deduplication function by combining the information available from repositories and identify whether a pair of record refers to the real world entity. Following approaches for record deduplication:

**i) Probabilistic approach**

Newcombe et al. [5] were the first ones to address the record deduplication problem as a Bayesian inference problem i.e, a probabilistic problem and proposed the first approach to automatically handle duplicates. However, their approach was considered empirical [10] since it lacks statistical ground.

After Newcombe et al.'s work, Fellegi and Sunter[4] proposed a more elaborated statistical approach to deal with this problem. Their method depends on the definition of two boundary values that are used to classify a pair of records as being duplicates or not. It is implemented with Bayes's rule and Naive based classification. This method is implemented in the tool such as, Febrl[2], usually work with two boundaries as follows:

1. Positive identification boundary—if the similarity value lies above this boundary, the records are considered to be duplicated.
2. Negative identification boundary—if the similarity value lies below this boundary, the records are considered not to be duplicated.

For the situation in which similarity values lies between the two boundaries, the records are classified as "possible matches or considered as their exist replicas" and, in this case, a human judgment is necessary.

**Limitations of Probabilistic Approach:**

- This method depends on the two boundary values definition that are used to classify a pair of records as being duplicates or not.
- Bad boundaries may increase the number of identification errors.

**ii) Machine Learning approach**

This method apply machine learning techniques for deriving record level similarity functions that combine field-level similarity functions, including the weights of records [6], [7], [8], [9]. It uses a small portion of the available data for training. This training data set is assumed to have similar characteristics to those of the test data set, which makes feasible to the machine learning techniques to generalize their solutions to unseen data.

It uses this approach to improve both the similarity functions that are applied to compare record fields and the way the pieces of evidence are combined. The main idea behind this approach is that, given a set of record pairs, the similarity between two attributes (e.g., two text strings) is the probability of finding the score of best alignment between them, so the higher the probability, the bigger the similarity between these attributes.

The adaptive approach presented in [8] consists of using examples for training a learning algorithm to evaluate the similarity between two given names, i.e., strings representing identifiers. We use the term attribute to generally refer to table attributes, record fields, data items, etc. This approach is applied to both clustering and pair-wise matching.

During the learning phase, the mapping rule and the transformation weights are defined. The process of combining the transformation weights is executed using decision trees. This system differs from the others in the sense that it tries to reduce the amount of necessary training, depending on user-provided information about the most relevant cases for training.

Active Atlas is an object identification system that aims at learning mapping rules for identifying similar records from distinct data sources. The process involves two steps as follows:

1) First, a candidate mapping generator proposes a set of possible mappings between the two set of records by comparing their attribute values and computing similarity scores for the proposed mappings.

2) Then, a mapping rule learner determines which of the proposed mappings are correct by learning the appropriate mapping rules for that specific application domain. This learning step is executed by using a decision tree.

**Limitations of Machine Learning Approach:**

- It requires large computation and memory storage requirement is high.
- Machine-learning techniques are data oriented i.e, they model the relationships contained in the training data set.

**iii) Genetic Algorithm**

Genetic Programming is one of the evolutionary programming technique which having the properties of natural selection or natural evolution. The main aspects that distinguish genetic programming from other evolutionary technique is that it represents the concept and interpretation of a problem as a computer program and even the data are viewed and manipulated in this way. This genetic programming is able to discover the variable and relationship with each other and find the correct functional form. It is having mainly three operation such as selection, crossover and mutation. All the operation has been included in the algorithm.

In this paper, we will see the overview of genetic programming and its basic concepts in Section 3. Also we will provide an approach for record deduplication by using genetic algorithm.

**Features of genetic programming:**

1) Working with multi-objective problems.
2) Good performance on searching over very large possibly infinite search spaces, where the optimal solution in many cases is not known, usually providing near-optimal answers.
3) The main aspect that distinguishes GP from other evolutionary techniques is that it represents the concepts and the interpretation of a problem as a computer program and even the data are viewed and manipulated in this way.
4) Applicability to symbolic regression problems.
5) Able to discover the independent variables and their relationships with each other and with any dependent variable.

## III. GENETIC PROGRAMMING CONCEPTS

Evolutionary computation is an area of computer science, which is inspired by the principles of natural selection as introduced by Charles Darwin. Genetic Programming is one of the best known evolutionary programming, which comes under machine learning.

To simulate the process of natural selection in a computer, we need to define the following: a representation of an individual. At each point during the search space we maintain a generation of individuals. In GP each individuals represents the possible solution for the problem. These individuals are represented by means of complex data structures such as trees, or graphs. After the initial population has been created, an actual evolutionary process starts [12].

A genetic algorithm (GA) is an evolutionary algorithm that is used to solve optimization problems. The algorithm iteratively refines an initial population of potential solutions until a solution is found. An initial population of solutions is created randomly. These solutions are then evaluated using a fitness function. A selection method is applied in order to choose a parent. Genetic operators are applied to the chosen parents to create offspring. This process of evaluation, selection and recreation is continued until either a solution has been found or a number of iterations/generations have been reached. It is well known for its best performance in searching large spaces and as well as its capability to operate over the population of individuals. It not only creates new solutions but also allows new combination of features[12]. The basic flow of genetic algorithm is shown in figure below.

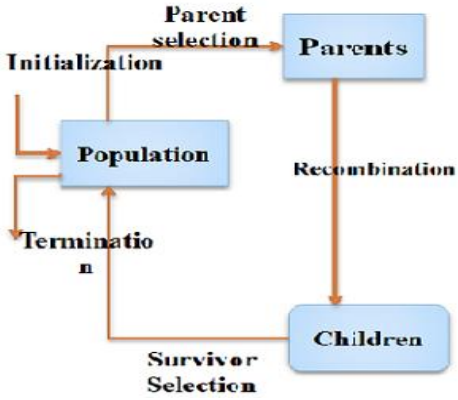


**Fig.1 Overview of genetic algorithm**

It includes individual i.e., any possible solution and population i.e., group of all individuals. Elements of chromosome are called genes. Chromosome is a set of genes and contains a solution in the form of genes. A gene contains a part of solution i.e. it determines the solution. The population size is one of the most important parameters that play a significant role in the performance of the genetic algorithms. A good population of individuals contains a diverse selection of potential building blocks resulting in better exploration.

Selection is the process of determining the number of times a particular individual is chosen for reproduction and, thus, the number of offspring that an individual will produce. In this, Roulette wheel selection is used. In roulette wheel selection, possible solutions are assigned fitness by the fitness function. This fitness level is used to associate a probability of selection with each individual. While candidate solutions with a higher fitness will be less likely to be eliminated, there is still a chance that they may be. With roulette wheel selection there is a chance some weaker solutions may survive the selection process; this is an advantage, as though a solution may be weak, it may include some component which could prove useful following the recombination process. Here, the fitness function is calculated by cosine similarity and string based similarity and then it applies to genetic algorithm[11].

## IV. GA APPLIED TO RECORD DEDUPLICATION WITH HASH-BASED ALGORITHM

In this work, we propose a representation based on trees for the individuals of the GP process which, in our case, represent possible record deduplication functions. More specifically, to perform the record deduplication, we calculate fitness function based on hash algorithm i.e, with MD5 and SHA-1 algorithm. If for two records same hash value is generated, then we consider it as duplicate record and

remove it by using GA. Following algorithmic steps are applied:

1) Initialize the population (with random or user provided individuals).
2) Evaluate all individuals in the present population, assigning a numeric rating or fitness value to each one.
3) If the termination criterion is fulfilled, then execute the last step. Otherwise continue.
4) Reproduce the best n individuals into the next generation population.
5) Select m individuals that will compose the next generation with the best parents.
6) Apply the genetic operations to all individuals selected. Their offspring will compose the next population. Replace the existing generation by the generated population and go back to Step 2.
7) Present the best individual(s) in the population as the output of the evolutionary process.

## I. CONCLUSION

The duplicate information is stored from different sources, which require more spaces to store replicas. Identifying and handling replicas is important to guarantee the quality of the information made available by digital libraries and e-commerce. These systems depend on consistent data to order high-quality services. These may be affected by the existence of duplicates or near-duplicate entries in their repositories. So by identifying and removing such duplicates from repositories is an important task. In this, we presented a GA approach with hash algorithm i.e, with MD5 and SHA-1 algorithm to record deduplication. Our approach is able to automatically suggest deduplication functions based on hash value. Also we can apply this technique for storing large data on cloud,. Also we extend this approach for small handheld devices like notepad, tab, mobile, etc.

## REFERENCES

[1] N. Koudas, S. Sarawagi, and D. Srivastava, "Record Linkage: Similarity Measures and Algorithms," Proc. ACM SIGMOD Int'l Conf. Management of Data, pp. 802-803, 2006.

[2] S. Chaudhuri, K. Ganjam, V. Ganti, and R. Motwani, "Robust and Efficient Fuzzy Match for Online Data Cleaning," Proc. ACM SIGMOD Int'l Conf. Management of Data, pp. 313-324, 2003.

[3] V.S. Verykios, G.V. Moustakides, and M.G. Elfeky, "A Bayesian Decision Model for Cost Optimal Record Matching," The Very Large Databases J., vol. 12, no. 1, pp. 28-40, 2003.

[4] I.P. Fellegi and A.B. Sunter, "A Theory for Record Linkage," J. Am. Statistical Assoc., vol. 66, no. 1, pp. 1183-1210, 1969.

[5] I. Bhattacharya and L. Getoor, "Iterative Record Linkage for Cleaning and Integration," Proc. Ninth ACM SIGMOD Workshop Research Issues in Data Mining and Knowledge Discovery, pp. 11-18, 2004.

[6] M. Bilenko, R. Mooney, W. Cohen, P. Ravikumar, and S. Fienberg, "Adaptive Name Matching in Information Integration," IEEE Intelligent Systems, vol. 18, no. 5, pp. 16-23, Sept./Oct. 2003.

[7] M. Bilenko and R.J. Mooney, "Adaptive Duplicate Detection Using Learnable String Similarity Measures," Proc. Ninth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining, pp. 39- 48, 2003.

[8] J.R. Koza, Gentic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press, 1992.

[9] W. Banzhaf, P. Nordin, R.E. Keller, and F.D. Francone, Genetic Programming - An Introduction: On the Automatic Evolution of Computer Programs and Its Applications. Morgan Kaufmann Publishers, 1998.

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

[11] Akshara k., Soorya P. "Replica free repository using genetic programming with decision tree" in International Journal of Advanced Engineering Applications, Vol.1, Iss.2, pp.62-66 (2012).

[12] S. N. Sivanandam and S. N. Deepak "Introduction to genetic algorithms" in Springer, NewYork, 2008.