

## A Novel Approach for Locating Identification of Similar Records

Satish B. Wagh<sup>1</sup>, Prof. Dr. Varsha H. Patil<sup>2</sup>

<sup>1</sup>PG Student, <sup>2</sup>Professor, Department of Computer Engineering, MCOERC, Nashik, (UOP, pune), India.

**Abstract--** With rapid advancement in technology enables high uses of database which causes duplication of database management. The replicated data records generate multiple copies of similar data is associated with record, in completed and also outdated or same data is duplicated in the database. For removing the similar data using various data cleaning software and applications are used. The proposed system for locating identification of similar record helps in combining several different pieces of evidence extracted from the data duplication function helps for identifying whether two entries are in repository same or not. In this approach for automatic adaptation of function are given fixed similar identification boundary for improving accuracy in terms of number of similar records found versus the actual number of duplicate records. This work has been tested on the cora dataset with similar records. System used for locating replica with the help of genetic programming helps in combining evidence extracted from the content and duplication function enables in identification weather two entries present in repository are replicas or not. Additionally genetic programming is capable for automatically adapting this function to a specified similar record identification boundary this genetic programming approach is applied for various database management to find similar records.

### I. INTRODUCTION

Several systems such as digital libraries and other database systems like organization databases are affected by the duplicates. The system for locating Identification of similar record using genetic programming approach that create duplication function which is able to identify whether two entries in a repository are replicas or not. Similar records is a task of identifying the similar record in a repository that refer to the same real world entity or object and systematically substitutes the reference pointers for the redundant blocks; also known as storage capacity optimization. Similar record identification is defined in various categories performance, data demand on more processing, more time is required to answer user queries; reduce the overall performance. Inconsistencies leads to distortions in reports and misleading conclusions based on the existing data; increasing operational costs because of the additional volume of useless data, extra costs are required on more storage media and extra computational processing power to keep the response time. To avoid such problems, it is necessary to study the causes of similar records in repositories.

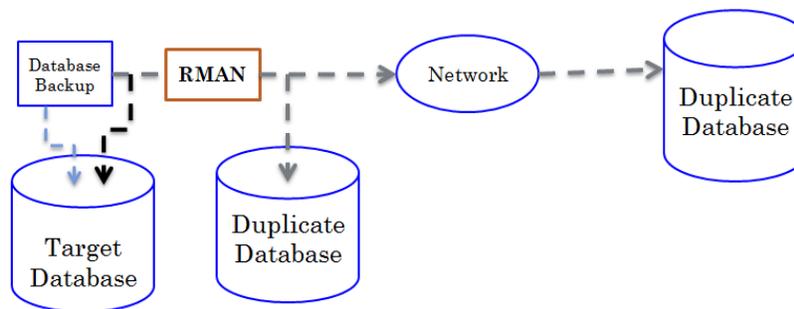


Fig.1.0 Database with Replica

A major root is the presence of similar, or near duplicates in these repositories, those constructed by the aggregation or integration of distinct data sources. The issue of detecting and removing duplicate entries in a repository is generally known as similar records. In this system the replica of dataset will be removed. As a part of genetic programming approach the gaining concepts and the entropy calculations are used to de-duplicate the records.

In Section 2, to describe literature survey. Section 3, briefly describes Implementation Details. To describe system architecture in Section 4. Section 5 describe genetic algorithm for Implementation. Specify dataset require for system and system experiment is described in Section 6, implantation detail in section 7 and conclude in Section VII.



## II. LITERATURE SURVEY

A literature has been dedicated to the record de-duplication and tremendous progress has been made ranging from efficient and scalable algorithm record de-duplication in Cora and Restaurant dataset. Replica identifications a growing research topic in database and related fields such as digital libraries. This problem arises mainly when data are collected from disparate sources using different information description styles and metadata standards. Common place for replicas are found in data repositories created from documents. The situations can lead to inconsistencies that may affect many systems such as those that depend on searching and mining tasks. To solve these inconsistencies it is necessary to design a duplication function that combines the information available in the data repositories in order to identify whether a pair of record entries refers to the real-world entity. In bibliographic citations, for instance, this problem was discussed by Lawrence et al. [13], [14]. Lawrence proposed a number of algorithms for matching citations from different sources based on word matching, phrase matching, and field extraction. As more strategies for extracting disparate pieces of evidence become available, Elmagarmid et al. [21] has proposed following two categories:

### 2.1 Probabilistic Approaches

W.W. Cohen (2002)[19,20] was proposed ones to address the record de-duplication problem as a Bayesian inference problem and proposed the first approach to automatically handle replicas and record de duplication problem as a Bayesian inference problem and proposed the first approach to automatically manage duplication. To Elaborated statistical method. Work, Fellegi and Sunter [5] proposed an elaborated statistical approach to deal with the problem of evidence. These methods rely on the definition of two boundary values that are used to classify a pair of records as being duplicate records or not. Febrile [2] to implement Tools for this method, such as, work with two boundaries as follows:

*Pros-* The positive identification boundary is the similar value lies above the boundary, the records are considered as replicas.

*Cons-* the negative identification boundary is the similar value lies below this boundary, the records are considered as not duplicate records.

### 2.2 Training-based Approaches

This category includes all approaches that depend on some sort of training supervised or semi-supervised in order to identify the replicas.

Machine learning approaches fall into this category. Next, brief comment on some works based on these two approaches (domain knowledge and training-based), those that exploit the domain knowledge and those that are based on probabilistic and machine learning techniques.

*Pros-* Closest matching approach.

*Cons-* For record matching high weight tokens are required.

### 2.3. Identification of record replication is done on individual basis.

Information Extraction using Learning Object Identification Rules When integrating information from multiple database, the same data objects can exist in consistent text formats [3] it is difficult to identify matching objects using exact text match. It has developed an attribute identification system called Active Atlas, which compares the different attributes in order to identify matching attributes. Certain attributes are more important for deciding if a mapping should exist between two attributes. Previous methods of object identification have required manual construction of attribute identification rules or mapping rules for determining the mappings between objects [5]. This process is manual, time consuming and error-prone. In this approach, Active Atlas learns to tailor matching rules, through limited input, to the specific application domain [16]. The experimental results demonstrate that we achieve higher accuracy and require less user involvement than previous methods across various application domains. Maximum problems arise when integrating data from multiple data sources [9]. One of these problems is that data attributes can exist in inconsistent text formats across several sources.

*Pros -*

- This method addresses the problem of mapping objects between structured web sources where same objects can appear. In similar yet inconsistent text format

- This method achieves high accuracy while limiting user input.

*Cons -*This method also does not provide high accuracy .needs more techniques.

## III. MATHEMATICAL MODEL

System for Replica Identification using genetic programming approach illustrates the mathematical model that contains duplicate records. By using Genetic Programming cosine similarity can be derived from this data set and de-duplicate functions are formed and it is being constructed in the form of tree.

*Dataset (ds)*:- Any Dataset containing different record from different domain

*Extraction (fe)*:- Extract Feature from input dataset

*Similarity Function (sf)*:- Find the Cosine Similarity between Records.

*Tree Construction (tc)*:- deduplication Function

$S = \{D, FE, SF, TC, GO, O\}$

Let D: -  $\{d_1, d_2, \dots, d_n\}$  where D consists of no of records

Let FE: -  $\{f_1, f_2, f_3, \dots, f_n\}$  where FE is function which extract features

Let SF: -  $\{v_1, v_2, v_3, \dots, v_n\}$  where v consists of similarity function records

Let TC: -  $\{n_1, n_2, \dots, n_n\}$  where tc is a tree constructor a function which constructs a tree

Let GO: -  $\{r_1, r_2, \dots, r_m\}$  where GO function is used to generate best fitness result

Let O: -  $\{o_1, o_2, \dots, o_n\}$  where O consist of output

Function F1 returns the features extracted from dataset record

$F1(d) \rightarrow FE$  for downloaded dataset

$F1(f) \rightarrow \{f_1, f_2, \dots, f_n\}$  to FE

Function F2 returns the similarity values of each record

$F2(FE) \rightarrow SF$

e.g:  $F2(FE) \rightarrow \{v_1, v_2, \dots, v_n\}$  to SF. Function F3 returns the Constructed tree or node.

$F3(SF) \rightarrow TC$

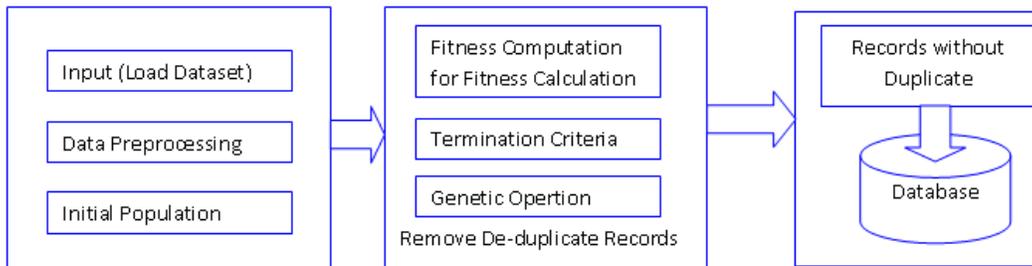
e.g:  $F3(SF) \rightarrow \{n_1, n_2, \dots, n_n\}$  to TC. Function F4 returns the Genetic operation result.  $F4(SF) \rightarrow GO$

e.g:  $F4(SF) \rightarrow \{r_1, r_2, \dots, r_m\}$  to GO. Function F5 returns the output

$F5(GO) \rightarrow O$

e.g:  $F5(GO) \rightarrow \{o_1, o_2, \dots, o_n\}$  to O. Functional Dependency of the above functions

#### IV. SYSTEM DESIGN



**Fig.2.0 System Architecture**

#### 4.1. Data Preprocessing

Cosine similarity is a measure of similarity between two vectors of an inner product space that measures the cosine of the angle between them. The cosine similarity value of 0 is 1, and less than 1 for any angle; the minimum value of the cosine is -1. The cosine angle between two vectors is determines whether two vectors are pointing in roughly the same direction. Cosine similarity between two strings is calculated using Levenstein distance and Soft-TFIDF similarity method. The cosine similarity function using Genetic programming approach automatically selects populations. Cosine similarity function also used to in training phase to capture the characteristics of dataset. Based on the characteristic of data set only populations are selected.

#### 4.2. Feature Vector Extraction

Populations are the feature vectors selected based on cosine similarity functions.

If the function does not reach fitness value then populations are changed. e.g :( att1,att2, att3) Selected fitness function by machine learning approach has to reach the fitness value. If fitness function does not reach fitness value then have to change the fitness function using genetic operations Genetic operations is mutation, crossover, and reproduction. Selected fitness function had represented in tree format, for applying genetic operations easily.

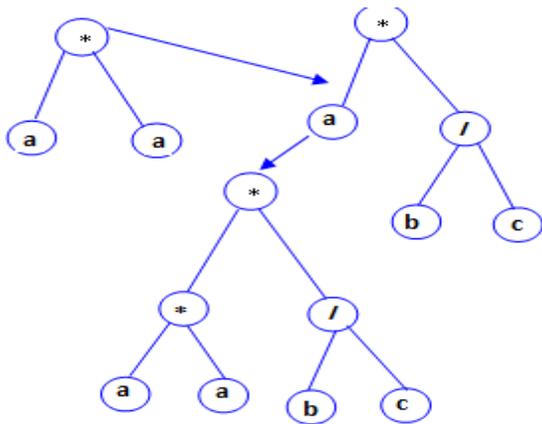
#### 4.3. Genetic Operations

If the function does not reach a fitness value have to apply genetic operations again to change the fitness function.

#### 4.4. Reproduction

Reproduction is the operation of Genetic programming that copies individuals without modifying them. The operator is used to implement an elitist strategy that is adopted to keep the genetic code of the fittest individuals across the changes in the generations.

If a good individual is found in previous generations, it will not be lost while executing the evolutionary process. The crossover operation allows genetic content which are the process of exchanging two parents, in this process that can generate two or more children. In a Genetic programming, two parent trees are selected according to a matching (or pairing) policy and, then, a randomly selected subtree is selected to every parent.



**Fig.3.0 Tree Construction for Objective Function**

#### 4.5 Mutation

The mutation operation has been implemented for keeping a minimum diversity level of individuals in the population thus it avoids premature convergence.

#### 4.6 Removing Duplicate Records

After selecting a fitness function firstly calculate fitness value for all records using a fitness function. The Fitness Function selected using existing machine learning approach. The selected fitness function can remove the records. Before storing records into a database calculate fitness values for all records if two records match with same fitness value then remove one record.

### V. GENETIC ALGORITHM

In production scheduling, population of solutions consists of many answers that may have different sometimes conflicting objectives. It operates on a population of solutions rather than a single solution. For example, in one solution it may be optimizing a production process to be completed in a minimum time. In another solution it may optimize for a minimum defects. By cranking up the speed at which it produce run into an increase in defects in our final product.

As increase in the number of objectives system are trying to achieve and also increase the number of constraints on the problem and similarly increase the complexity. Genetic programming is better for these types of problems where the search space is larger and the number of feasible solutions is very small. Modeling the Record Deduplication Problem using GP while using Genetic Programming to solve a problem, there are some ordinary requirements that must be fulfilled, order to successfully explore this technique which are based on the data structure used to represent the solution of record de duplication. It chosen a tree based GP representation for the evidence combination function, since it is a natural representation for this type of defined functions.

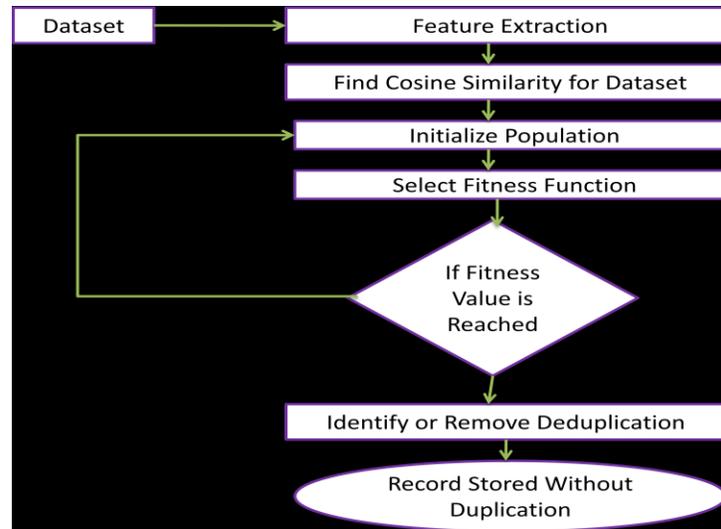
These requirements are the following:

- 1) The problem solution must be modeled as a tree structure.
- 2) The evolutionary operations applied over the modeled tree must be at the end of the result and converted into a valid tree finally.
- 3) The modeled tree must be automatically evaluated in order to make the use of this technique viable.

Evidence E is a pair  $\langle$ attribute, similarity function that represents the use of a specific similarity function over the values of a specific attribute found in the data repositories being analyzed. E.g, it wants to deduplicate a database table with four attributes (e.g., name, surname, address, and postal code) using a specific similarity function.

The system initially get the input of dataset are preprocessed and the Features are extracted. These preprocessed dataset are used to reduce the size of support vector model, Cosine similarity on this vector data is measure and a tree is constructed. Finally following de duplication function applied to genetic model,

1. Initialize and set the population (random or user provided individuals).
2. Evaluate all individuals in the current population; apply a numeric rating or fitness value to each one.
3. If the termination criterion is fulfilled, then execute the last step and continue.
4. Reproduce the best n individuals in the next generation population.
5. Select n individuals that will process the next generation with the best parents.
6. Apply the genetic operations to all individuals selected and compose the next population. Replace the existing generation of the generated population and go back to Step2
7. Present best individuals in the population as the output of the evolutionary process.



**Fig.4.0 System Flow**

Genetic algorithms (GAs) begin with a set of solutions which is represented by chromosomes, called population. Solutions from population are taken and used to form a new population, which is motivated by the possibility that the new population will be better than the old one. Further, solutions are selected according to their fitness to form new solutions.

## VI. DATASET

Apart from the tasks related to the logical and the physical side of the database system, Database Administrator may also take part in database System operations. Main role is to give developers recommendations about the DBMS specificity, other important work of the DBAs are related to data modeling at optimizing the system, as well as to the creation and analysis of new databases. Computer science is evolutionary computation sub field of artificial intelligence (more particularly computational intelligence) that involves combinatorial optimization problems. Restaurant and Cora datasets are will be used to analyze the proposed algorithm and the performance of the proposed algorithm is compared against the genetic programming technique with the help of evaluation metrics in this experiment

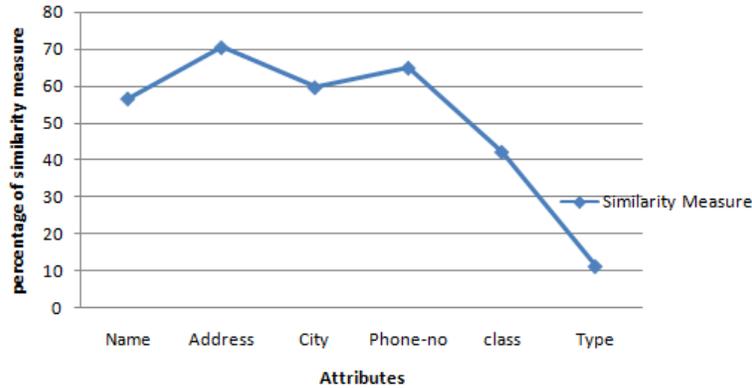
Select datasets from the cora dataset data repository and the datasets used is Restaurant dataset. The details about datasets are given below.

*Dataset1 (CORA):* The Cora dataset consists of duplicate and non-duplicates data records which is cited to 122 Conference paper, it is divide into various attributes(name, year, title, and other information)

*Dataset2 (Restaurant):* This dataset contains 500 records (400 originals and 100 duplicates), duplicates record based on one original record (using a Poisson distribution of duplicate records) and with a maximum of two modifications in a single

## VII. EXPERIMENTAL RESULT

The result of the system is in term of Feature extraction, similarity measure Cora dataset, discriminative attribute selection, experiment with replica identification, fitness calculation and computation time are presented and discussed. This module elaborate the object identification system that aims at learning mapping rules for identifying similar objects or records from data sources. The system process two steps first, a records matching generator proposes a set of possible match between the two sets of objects by comparing their attribute values and computing similarity scores for system; this is shown in fig 5.0

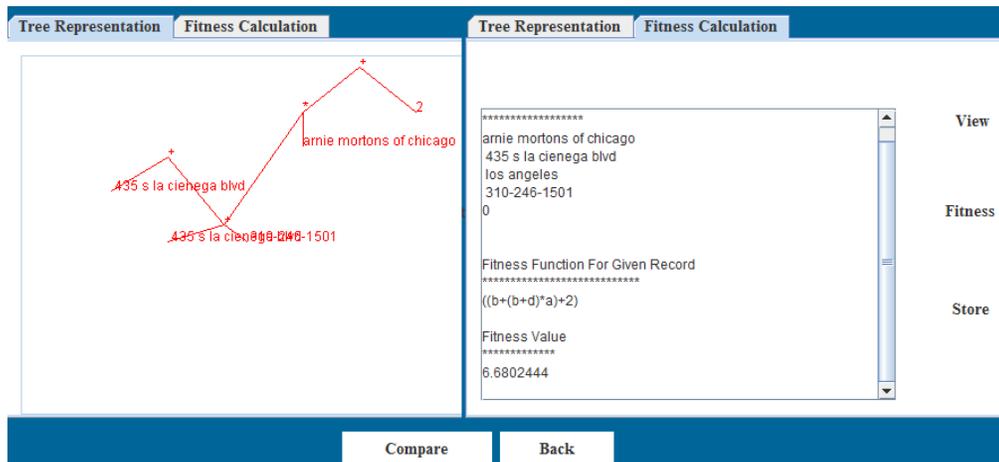


**Fig. 5.0 Similarity Measure**

**7.1 Tree Construction and Fitness Calculation**

This result described representation based on trees for the individuals of the GP process, it represent possible record duplication functions and perform the similar records identification.

Combination functions are in all sets E is a pair <attribute, similarity function> that represents the application of a specific similarity function on the values of a given attribute of the data repository.



**Tree Representation**

```

graph TD
    Root(( )) --- Node1((arnie mortons of chicago))
    Root --- Node2((435 s la cienega blvd))
    Node1 --- Node3((435 s la cienega blvd))
    Node1 --- Node4((310-246-1501))
    
```

**Fitness Calculation**

```

*****
arnie mortons of chicago
435 s la cienega blvd
los angeles
310-246-1501
0

Fitness Function For Given Record
*****
((b+(b+d)*a)+2)

Fitness Value
*****
6.6802444
        
```

Compare
Back

**VI. CONCLUSION AND FUTURE WORK**

This system helps in presenting system for Replica Identification via genetic programming. The approach is combination of several pieces of evidence that has been extracted from the data content for the production of de-duplication function. That enables for identification whether two or more entries are in the repository or replicas or not. In future this approach will help in finding the complex matches in three different data repository scenarios where the repositories partially share some data or some common data.

**REFERENCES**

- [1] oises G. de Carvalho, Alberto H.F. Laender, Marcos Andre Goncalves, and Altigran S. da Silva ,A Genetic Programming Approach to Record DeduplicationIEEE Transaction on knowledge and Data Engineering vol.24,No.,3, March 2012
- [2] Agrawal, S. Chaudhuri, G. Das, A. Gionis, Automated ranking of database query results, in: Proceedings of the First Biennial Conference on Innovative Data System Research, 2003Ahmed, C. F., Tanbeer, S. K., Jeong, B.-S., Lee, Y.-K. (2008). Mining weighted frequent patterns in incremental databases. In: Proceeding of the 10th Pacific Rim international conference on artificial intelligence (pp. 933938). Hanoi, Vietnam, December 1519.
- [3] Bhattacharya and L. Getoor, Iterative Record Linkage for Cleaning and Integration, Proc. Ninth ACM SIGMOD Workshop Research



**International Journal of Recent Development in Engineering and Technology**  
**Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 3, Issue 3, September 2014)**

- [4] 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.
- [5] P. Fellegi and A.B. Sunter, A Theory for Record Linkage, J. Am. Statistical Assoc., vol. 66, no. 1, pp. 1183-1210, 1969.
- [6] 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.
- [7] Bell and F. Dravis, "Is Your Data Dirty? and Does that Matter?", Accenture Whiter Paper, <http://www.accenture.com>, 2006.
- [8] R. Koza, Genetic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press, 1992.
- [9] M. de Almeida, M.A. Gonalves, M. Cristo, and P. Calado, A Combined Component Approach for Finding Collection-Adapted Ranking Functions Based on Genetic Programming, Proc. 30th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval, pp. 399-406, 2007.
- [10] Zhang, Y. Chen, W. Fan, E.A. Fox, M. Gonalves, M. Cristo, and P. Calado, Intelligent gp Fusion from Multiple Sources for Text Classification, Proc. 14th ACM Int'l Conf. Information and Knowledge Management, pp. 477-484, 2005.
- [11] G. de Carvalho, M.A. Gonalves, A.H.F. Laender, and A.S. da Silva, Learning to Deduplicate, Proc. Sixth ACM/IEEE CS Joint Conf. Digital Libraries, pp. 41-50, 2006.
- [12] 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.
- [13] Lawrence, L. Giles, and K. Bollacker, Digital Libraries and Autonomous Citation Indexing, Computer, vol. 32, no. 6, pp. 67-71, June 1999.
- [14] 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.
- [15] W. Cohen, Data Integration Using Similarity Joins and a Word Based Information Representation Language, ACM Trans. Information Systems, vol. 18, no. 3, pp. 288-321, 2000.
- [16] C.P. Carvalho and A.S. da Silva, "Finding Similar Identities among Objects from Multiple Web Sources," Proc. Fifth ACM Int'l Workshop Web Information and Data Management, pp. 90-93, 2003.
- [17] B. Newcombe, J.M. Kennedy, S. Axford, and A. James, Automatic Linkage of Vital Records, Science, vol. 130, no. 3381, pp. 954-959, Oct. 1959.
- [18] Freely Extensible Biomedical Record Linkage, [http:// sourceforge.net/projects/febrl](http://sourceforge.net/projects/febrl) , 2011.
- [19] W. Cohen and J. Richman, Learning to Match and Cluster Large High-Dimensional Data Sets for Data Integration, Proc. Ninth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining, pp. 475- 480, 2002.
- [20] Tejada, C.A. Knoblock, and S. Minton, Learning Object Identification Rules for Information Integration, Information Systems, vol. 26, no. 8, pp. 607-633, 2001.
- [21] Guha, N. Koudas, A. Marathe, and D. Srivastava, Merging the Results of Approximate Match Operations, Proc. 30th Int'l Conf. Very Large Data Bases, pp. 636-647, 2004.
- [22] J. Angeline, Genetic Programming's Continued Evolution, Advances in Genetic Programming, vol. 2, ch. 1, MIT Press, 1996.