Token-Based Data Cleaning Technique for Job Assigning System

Yu Wai Hlaing, Khin Mar Myo
University of Computer Studies (Mawlamyine)
yuwaihlaing.1987@gmail.com, kmmyo09@gmail.com

Abstract

The problem of detecting and eliminating duplicate data is one of the major problems in broad area of data cleaning and data quality. Data cleaning is the process of identifying and removing duplicates in the database. Today, several existing data cleaning techniques have been widely used for this purpose. This paper describes how to effectively use token-based cleaning technique and positional algorithm in the development of job assigning system for detecting and eliminating duplicate job application forms. Token-based data cleaning technique is effective and efficient for data cleaning which decomposing and reassembling the data to get a unique set of string from ordinary user input data in comparing and eliminating duplicate records from database. This approach also eliminates the need to use the entire long string records with multiple passes, for duplicate identification. After eliminating duplicate data, system checks that the person has already assigned in one job position or not using Positional algorithm by identifying NRC number as match field. Positional algorithm is used because it can find errors with positional disorders and gap penalties. As a result, the system can get clean data that are not duplicated.

1. Introduction

Database plays an important role in today's IT based economy. High quality data or clean data are essential to almost any information system that requires accurate analysis of large amount of real-world data. Data cleaning is an emerging domain that aims at improving data quality. Data cleaning is a process for determining whether two or more records defined differently in a database, actually represent the same real world object. During data cleaning, multiple records representing the same real life object are identified, assigned only one unique database identification, and only one copy of exact duplicate records is retained [5]. The reliability of the data sources is not always assured when the data collection is voluminous, large amount of data can be deposited into the operational data sources in a batch mode or by data entry without sufficient checking. Given the excessive redundancies and the numerous way errors can be introduced in a database, it is not surprising that data cleaning is one of the fast evolving research interests in the 21st century [6].

Several existing data cleaning techniques and algorithms have been widely used in detecting and eliminating duplicate data. Token-based data cleaning technique also eliminates the need to rely on match threshold by defining tokens that are used for identifying duplicates. In token-based data cleaning technique, user select most important fields and rank them based on their power to uniquely identified records. The elements in the selected fields are tokenized and sorted on uniquely identifying field. And then duplicate records are detected and eliminated [2, 5]. At the token-match level, the scheme used by the positional algorithm copes with errors in tokens by charging gap penalties. Penalties are also charged for characters with positional disorder [4]. The differences between the positional algorithm and the basic recursive algorithm are that (1) position of character and word tokens are remembered and used to disallow re-matching characters/words that had participated in the previous matches, (2) positions are used to charge disorders and gap penalties to matched words that may lead to overall incorrect matches.

In this paper, token-based data cleaning technique and positional algorithm are used to detect and eliminate duplication by using tokens for detecting and removing duplicate records which makes the real data ready for mining techniques.

The rest of this paper is organized as follows. Section 2 summarizes the related work in the area of data cleaning and positional algorithm. In section 3 we introduce some background theory. Section 4 describes the system design of our job assigning system. Section 5 presents the implementation of our system. Section 6 describes the performance analysis. Finally, we conclude the paper in section 7.

2. Related Work

Author [7] proposed a framework for data cleaning that offers the fundamental services for data cleaning such as attribute selection, formation of tokens, selection of clustering algorithm, selection of similarity function, selection of elimination function and merge function. It also presented a solution to handle data cleaning process by using a new

framework design in a sequential order. Rohit Ananthakrishna [1] developed an algorithm for eliminating duplicates in dimensional tables in data warehouse, which are usually associated with hierarchies. The idea of exploiting hierarchies is to develop a high quality, scalable duplicate eliminating algorithm and evaluate it on real datasets from an operational data warehouse. Rawshan Basha [2] used similarity function to detect and eliminate duplication by using well-defined tokens for records matching in a domain-independent algorithm, for detecting and removing duplicates which makes the real data for mining techniques. C.I.Ezeife [4] proposed an algorithm that achieves a domain independent de-duplication at the attribute level and a technique for field weighting through data profiling achieves domain-independent cleaning at the record level.

According to above literature review, there have been developed many frameworks and algorithms to eliminate the duplicate records. Among them, we choose token-based data cleaning method for our system implementation.

3. Background Theory

In this section, two background theories that are applied in the job assigning system are discussed.

3.1 Data Cleaning Technique

Data cleaning is a process used to determine inaccurate, incomplete, or unreasonable data and then improving the quality through correction of detected errors and omissions.

Existing data cleaning techniques used to identify record duplicates, record and field similarities and duplicate elimination. Among data cleaning techniques, token-based data cleaning technique is a technique which meets the requirements to solve the problem of duplication. In this paper, token-based data cleaning technique is used for solving the problem of data duplication.

3.1.1 Token-Based Data Cleaning Technique. Token-based data cleaning is decomposing and reassembling the data to get a unique set of string from user input data in comparing and eliminating duplicate records from database. There are four steps included in the technique. They are:

(1) Selecting and ranking of fields
(2) Extracting and formatting of tokens
(3) Sorting of tokens
(4) Duplicate detecting and eliminating

Step 1: Selecting and Ranking of fields

The condition for "fields' selection and ranking" is that the user is very familiar with the problem domain. The system needs to select the records from the table and ranks them according to the date of job seekers applied the job.

Step 2: Extracting and Formatting of Tokens

Extract smart token for each selected field as follow. Form numeric, alphabetic or alphanumeric tokens after removing stops words and unimportant characters like "#", "(". There are three types of tokens see in [2, 5].

Step 3: Sorting of Tokens

The table of tokens is sorted separately on the uniquely identifying field according to ranking by the user. In our domain, NRC number is used as a sorting key. The purpose of sorting on tokens is to catch possible duplicate records that are not brought together.

Step 4: Duplicates Detecting and Eliminating

The main cleaning tasks are accomplished in this step. Find/detect duplicate records in the list which got after step3. Duplicate elimination task ensures that only one copy of records found to be duplicates is retained. Firstly, similarity match count (SMC) is searched from the records of the token for detecting duplicate records.

SMC=number of corresponding token fields that match/number of token fields used **(1)**

If equation (1) result is 1, the tokens are perfect match. If SMC result is less than 1, a function further computes the "similarity match ratio" SMR.

SMR=(2*number of common characters in the two tokens)/total number of characters in the two tokens **(2)**

If equation (2) result is 1, the tokens are detected as duplicates and eliminated by retaining only one of the duplicates.

3.2 Positional Algorithm

Positional Algorithm is proposed by [4] as an alternative algorithm for data cleaning. It can also be used as duplicate record eliminator. The main idea of this algorithm is to match the two strings (two fields) according to the position of characters. There are two main steps in this algorithm.

The first step is to convert the two strings into tokens to reduce the length of strings. The second step is to match the characters from two tokens. Once a character in the first token matches the character in the second token, position 1 in the second token is noted. So that the next search for the following character will begin at position 2. For example, if the first token is "ISDN" and the second token is "ISBD", the matching process will start at

position 1 of each token. They are match at position 1. The match score of 1 is noted to the match score of token. Therefore matching process is moved to position 2. The character (D) from token 1 at position 3 and the character (B) from token 2 at position 3 are not match. Therefore the position of token 2 is moved to 4. Then the character from position 3 of token 1 is matched with the character from position 4 of token 2 but there is a gap. After matching the characters from token1 and token 2 at position 3 and 4 respectively, gap penalty case is happened. In other words, the gap is found in this case. The gap is between position 3 from token 1 and position 4 from token 2.

If the gap is found in each matching process, the match score is calculated based on the following equations;

For 1st gap,

match score= (-0.2*1*number of gaps) **(3)**

For next subsequent gaps,

match score =(-0.2*2*number of gaps) **(4)**

Once a character matches a score of 1 is added to the match score of token being search for. However, if there is a character disorder in the token, a disorder penalty of -1 is charged. So, the effective match score when such a disorder occurs is 0.

For example, if the first token is "TIM" and second token is "SMITH". The match locates for 'T' in the 4th position in "SMITH", a score of 1 is given, 'T' is matched as already matched, and the match position recorded. The next character is 'I', and it is searched for, in characters positions right of 'T' in "SMITH". In this case, there is no 'I' right to 'T', so the search wraps around to the first character of "SMITH". The character 'I' is then located at the 3rd position in "SMITH". So, a disorder case is happened. If the positional disorder is found in each matching process, the match score is calculated based on the following equation;

For positional disorder,

match score= (1+(-1))=0 **(5)**

After computing equation (3, 4 and 5), the final match score is computed by summing the match score of each character and dividing them by number of characters in the first token.

Final Match Score= Total match score/number of characters in first token **(6)**

Equation (6) is computed to make the decision as to whether two tokens are match or not. A final match score lower than the match threshold returns a "no match" result for this token with the final match score of 0 for this token, but a "match" the final match score of 1 otherwise.

4. System Design

In this section, we describe phases to implement a job assigning system. A job assigning system flow is shown in figure:

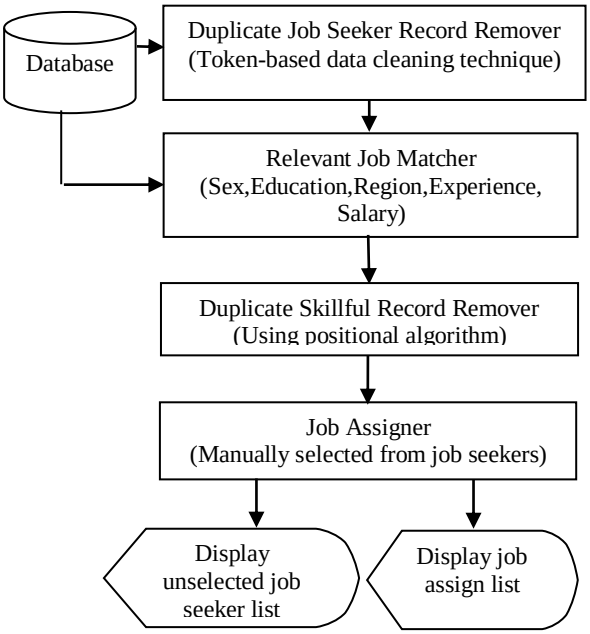

Figure 1. Job assigning system flow

The system design is constructed based on the token-based data cleaning technique and positional algorithm. It contains four main components in system design as shown in Figure 1.

(i) Duplicate Job Seeker Record Remover

The selected fields from the data set are converted into tokens in this phase. If duplicate token records are detected by using equation (1) and (2), only one record is remained the same in database. (For example, if four duplicated records are found in database, then three records are deleted and one record is retained in database).

(ii) Relevant Job Matcher

The skills of job seeker table (Table 7) that are of the duplicate free records from phase one is matched with the table of job vacancy (Table 2) announced by employers. In this phase, person who is not matched with the required skills for vacant job position is removed by matching sex, region, education, work experience and salary fields of two tables. The skillful person records are stored in Table 8.

(iii) Duplicate Skillful Record Remover

The skillful personal data (Table 8) getting after phase two are matched with the records of persons (Table 9) who have been assigned job via our agency in previous interview. The main idea is to prevent the wrong assigning the person who have already got job. This process is done by using

positional algorithm with NRC number as match field. In this phase, equation (3), (4), (5) and (6) are used for detecting duplicate personal data from Table 8 and Table 9. Then, the final skillful person list is stored as short list in Table 10. This short list is sent to Employer to select person manually according to their needs.

(iv) Job Assigner

The short list table data from phase three are sent to job assigner to manually select the job seeker from short list to be assigned or not in applied job position. And then, the lists of job assign and unselected person are displayed.

5. Implementation of Job Assigning System

To effectiveness of our system design, job assigning system is implemented based on career consultant agency. There are two main actors in this system: (i) job seekers and (ii) employers. This system is responsible to store not only the job seeker's personal data but also the employers' job vacancy announcement data. According to the nature of human being, job seeker sends job application forms many times until getting of job. So, the system is needed to detect and eliminate the duplicated application forms.

This section describes the system implementation in detail.

5.1 Data Source

There are two data sources in our system. When job seekers send job application forms, the data are stored in job seeker table. When employers announce vacant job position, the data are also stored in job vacancy announcement table. So, two data sources of our system are:

(i) Job seeker table

(ii)Job vacancy announcement table

Table 1. Job seeker table

id	Name	NRC	DOB	Apply Date	…	Regionid	Jobid
1	Ei Ei	12/ISN(N)456111	09/23/1977	12/05/2009		1	2
2	Tin Tin	14/PTN(N)030456	12/04/1983	12/07/2009		1	2
3	Hla Hla	12/ISN(N)181011	11/05/1976	12/03/2009		2	2
4	Su Su	10/BLN(N)009876	06/28/1980	12/07/2009		1	2
5	Hla Hla	12/ISN(N)181011	11/05/1976	12/08/2009		1	2

The database schema for job seeker table is (id,Name,FatherName,NRC,DOB,ApplyDate,Jobid, Regionid,Salary,WorkingExperience,Sex,Education, Address,Phone number).

Table 2. Job vacancy announcement table

Jobid	Regionid	Salary	Working Experience	Educationid
1	1	40000	No	2
2	1	60000	1	3
3	8	70000	3	5
4	3	100000	No	4

5.2 Duplicate Job Seeker Record Remover

In this phase, token-based data cleaning technique is used for detecting and eliminating duplicate records. The detailed steps of token-based data cleaning technique are as follows:

(1)Selecting and Ranking of Fields

In our system, (Name, NRC, DOB, ApplyDate) fields are selected to be tokenized and ranked the records by apply date. In this step, Table 1 is transformed into Table 3.

Table 3. Job seeker table (after selecting of fields and ranking by applydate)

id	Name	NRC	DOB	ApplyDate
1	Hla Hla	12/ISN(N)181011	11/05/1976	12/03/2009
2	Ei Ei	12/ISN(N)456111	09/23/1977	12/05/2009
3	Tin Tin	14/PTN(N)030456	12/04/1983	12/07/2009
4	Su Su	10/BLN(N)009876	06/28/1980	12/07/2009
5	Hla Hla	12/ISN(N)181011	11/05/1976	12/08/2009

(2)Extracting and Formatting of Token

In step 2, the fields that are selected in previous step are tokenized and formatted for detecting and eliminating duplicate records.

Table 4. Token table

id	Name	NRC	DOB	ApplyDate
1	HH	ISNN12181011	11051976	12032009
2	EE	ISNN12456111	09231977	12052009
3	TT	PTNN14030456	12041983	12072009
4	SS	BLNN10009876	06281980	12072009
5	HH	ISNN12181011	11051976	12082009

(3)Sorting of Tokens

The table of tokens is sorted by token NRC number. In this step, Table 4 is transformed into Table 5.

Table 5. Token table (after sorting)

id	Name	NRC	DOB	ApplyDate
1	SS	BLNN10009876	06281980	12072009
2	HH	ISNN12181011	11051976	12032009
3	HH	ISNN12181011	11051976	12082009
4	EE	ISNN12456111	09231977	12052009
5	TT	PTNN14030456	12041983	12072009

(4)Duplicate Detecting and Eliminating

In this step, duplicate records are detected by using equation (1) and (2). And then, duplicate

records are eliminated by retaining only one of them. There are two records for one person in transaction 2 and 3. In other words, duplicate records are found in transaction 2 and 3. Therefore, one record is removed from Table 5, it becomes Table 6 with four transactions. By this way, duplicate records are detected and eliminated.

Table 6. Duplicate free token table

id	Name	NRC	DOB	ApplyDate
1	SS	BLNN10009876	06281980	12072009
2	HH	ISNN12181011	11051976	12032009
3	EE	ISNN12456111	09231977	12052009
4	TT	PTNN14030456	12041983	12072009

5.3 Duplicate Skillful Record Remover

In this phase, the matching process between the skills of job seekers and the required skills of job vacancy announced by registered company (Employer) is conducted. As described in Table 1, there are 14 attributes for one person. Some of the attribute values concerning with the job seeker's qualification are extracted from Table 1 to Table 7. The record set from Table 7 is matched against with the record set from Table 2 because it stores the required skills for job seekers.

Table 7. Skills of job seeker table

id	Jobid	Region id	Salary	Working Experience	Educatio nid
1	2	1	60000	1	3
2	2	2	60000	1	3
3	2	1	60000	1	1
4	2	1	60000	1	3

The results of this step are shown in Table 8 which stored the personal data that are matched with the required skills for vacant job position.

Table 8. Skillful personal table

id	Name	NRC	DOB	ApplyDate
1	SS	BLNN10009876	06281980	12072009
2	TT	PTNN14030456	12041983	12072009

5.4 Selected Applicant Checker

In this phase, the table of skillful person is compared with job assign table that is stored the data of person who is already assigned in one job position for registered company using positional algorithm. The main idea of using positional algorithm is for data de-duplication.

Table 9. Job assign table

sid	Name	NRC	DOB	ApplyDate
1	AA	ISNN12022095	06031975	11022009
2	PP	PBDN1285320	12131988	11132009
3	SS	BLNN10009876	16281980	11172009
4	HH	PTNN14234638	10111965	11232009
5	NN	KKHN14036264	08191977	11292009
6	MM	UKMN12192453	10251985	12012009

According to the steps described in section 3.2, NRC number field from Table 8 is matched with NRC number field from Table 9. There may be happened two cases in each match. For example, if the NRC number field from Table 8 is BLNN10009876 and from Table 9 is also BLNN10009876, the final match score value calculating by positional algorithm is 1. If the NRC number field from Table 8 is BLNN10009876 and from Table 9 is ISNN12022095, the final match score value calculating by positional algorithm is 0.25. If a match score is 1, the two records are identified as duplicates and remove duplicate record from Table 8.

In this Phase, equations (3, 4, 5 and 6) are used to detect duplicate records by matching skillful person's NRC number with already job assigned person's NRC number. The result of this step (short list) is stored in Table 10.

Table 10. Short list table

id	Name	NRC	DOB	ApplyDate
1	TT	PTNN14030456	12041983	12072009

5.5 Job Assigner

In this phase, the short list table data are sent to the employer (Company) to manually select from short list table to be assigned or not in applied job position. And then, the lists of job assign and unselected person are displayed. After the person is selected for one job position, token data of this person is stored in the Table 9.

6. Performance Analysis

Other paper measures performance of duplicate removing algorithm with some parameters, namely, (i) recall (RC), (ii) false-positive error (FPE), (iii) reverse false-positive error (RFP) and (iv) threshold. In this paper, we measure the performance of duplicate removing algorithm with Recall.

Recall is a ratio indicating the number of duplicates correctly identified by a given algorithm. For example, if "x" is the number of duplicates were identified out of "y' number of duplicates, then the recall is x/y, which when expressed in percentage is (x/y)*100. After testing some appropriate amount of data set (about 2500 records), the recall rate of our system obtains nearly round about 90%.

7. Conclusion and Further Extension

Today, both individual and organization are needed to store and analyze the huge amount of data. Data preprocessing and cleaning is important to efficient and effective analysis. Token-based approach can be used as one of the data cleaning technique especially for removing duplicate records. This system is concerned with career consultant agency. There are many data not only from

employers but also from job seekers. According to nature of applicants' mind, they send application form many times. Therefore, duplicate records are accumulated in career consultant agency. Experiment shows that our system performs effectively and efficiently not only in records of duplicate elimination but also in job assigning between job seekers and employers. This system tries to eliminate the duplicate records of applicants by taking the advantages of token-based data cleaning approach. This system can not provide other facilities such as automatic finding of appropriate job for job seekers, and also automatic finding of job seekers for employers vice versa. This system can be improved as an agent-based system by providing more automatic tasks between job seekers and employers. This system can also be improved by using the token-based data cleaning technique not only in job seekers data but also in Job vacancy announcement from employers. Future work should be considered by applying token-based data cleaning technique on unstructured (like complete text file), and semi-structured (like XML file) data.

Reference

[1] R.Ananthakrishna, "Eliminating Fuzzy Duplicates in Data Warehouses", Cornell University, rohit@cs.cornell.edu.

[2] R.Basha, "Using Well Defined Tokens in Similarity functions for Records Matching in Data Cleaning Techniques", Computer Science Department, University of Sharjah, UAE.

[3] D.Bitton and D.J.Dewitt, "Duplicate Record Elimination in Large Data Files. ACM Transaction on Database System", vol 8, No.2,PP 225-265, June 1983.

[4] C.I.Ezeife, "Data Position and Profiling in Domain-Independent warehouse cleaning", School of Computer Studies, University of Windsor, Canada.

[5] C.I.Ezeife, "Token-Based data Cleaning technique for Data Warehouse Systems", School of Computer Studies, University of Windsor, Canada.

[6] H.Müller, J.C.Freytag, "Problems, Methods, and Challenges in Comprehensive Data Cleansing", Humboldt-Universität zu Berlin, 10099 Berlin, Germany {hmueller,freytag}@dbis.informatik.hu-berlin.de

[7] J.J.Tamilselvi and Dr.V.Saravanan, "A Unified Framework and Sequential Data Cleaning Approach for a Data Warehouse", Department of Computer Application, Karunya University, Coimbatore-641 114, Tamilnadu, India.