MATHEMATICAL REPRESENTATION AND IMPLEMENTATION OF ALLIANCE RULES FOR DATAWAREHOUSE CLEANSING

Developed by:

Ms. Shamsa Wahid
Registration # 329-FAS/MSCS/F06

Supervised by:
Mr. M. Imran Saeed

Co-Supervised by:
Ms. Zakia Jalil

Department of Computer Science
& Software Engineering
Faculty of Basic and Applied Sciences
International Islamic University Islamabad

(2011)

DEPARTMENT OF COMPUTER SCIENCE & SOFTWARE ENGINEERING
INTERNATIONAL ISLAMIC UNIVERSITY ISLAMABAD

FINAL APPROVAL

Date: 30-12-2011

This is to certify that we have read the thesis submitted by **Shamsa Wahid, Registration # 329-FAS/MSCS/F06.** It is our judgment that this thesis is of sufficient standard to warrant its acceptance by International Islamic University, Islamabad for the degree of **MS COMPUTER SCIENCE.**

COMMITTEE

EXTERNAL EXAMINER

Dr. Abdus Sattar
Ex-Chairman,
Pakistan Computer Bureau

INTERNAL EXAMINER

Dr. Ayyaz Hussain
Assistant Professor,
Department of Computer Science
IIUI, Islamabad

SUPERVISOR:

Mr. Muhammad Imran Saeed
Assistant Professor

Co-Supervisor:

Ms. Zakia Jalil
Lecturer

Department of Computer Science & Software Engineering,
Faculty of Basic and Applied Sciences,
International Islamic University, H-10, Islamabad.

Dedication

Dedicated to
my parents, sister, brother
and my husband,
whose affection has always been
the source of encouragement for me,
and whose prayers have always been
a key to my success.

A dissertation Submitted To

Department of Computer Science & Software Engineering,

Faculty of Basic and Applied Sciences,

International Islamic University, Islamabad

As a Partial Fulfillment of the Requirement for the Award of the Degree of *MS Computer Science*

Declaration

I hereby declare that this Thesis "***Mathematical Representation and Implementation of Alliance Rules For Data Warehouse Cleansing***" neither as a whole nor as a part has been copied out from any source. It is further declared that I have done this research with the accompanied report entirely on the basis of my personal efforts, under the proficient guidance of my teachers especially my supervisor *Mr. M. Imran Saeed, and* co- supervisor *Ms. Zakia Jalil.* If any part of the system is proved to be copied out from any source or found to be reproduction of any project from any of the training institute or educational institutions, I shall stand by the consequences.

Shamsa Wahid.

Registration# 329-FAS/MSCS/06

Acknowledgement

I am very grateful to Almighty Allah, the Compassionate and the Merciful, who knows about whatever is there in the universe, hidden or evident and has enabled me to elucidate a drop from the existing ocean of knowledge. All praises be to the Holy Prophet Muhammad (Peace Be Upon Him), a star brightening the path of faith and knowledge, luminary to the truth and justice.

I wish to extend my sincere gratitude to my supervisor Mr. M. Imran Saeed and co-supervisor Ms. Zakia jalil for their favorable concentration and support in this research work.

I owe a debt of appreciation to my husband Mr. Kashif Javaid for his immense cooperation at various stages, when I needed his guidance without which the research work would not have been accomplished.

Shamsa Wahid

Registration# 329-FAS/MSCS/06

Project In Brief

Project Title:	Mathematical Representation and Implementation of Alliance Rules for Data Warehouse Cleansing
Undertaken By:	Shamsa Wahid. Registration # 329-FAS/MSCS/06
Supervised By:	Mr. M. Imran Saeed.
Co- Supervised By:	Ms. Zakia Jalil.
Start Date:	January, 2011
Completion Date:	August, 2011
Tools & Technologies	VISUAL STUDIO .NET 2010 VISUAL BASIC .NET SQL SERVER 2005
Documentation Tools	MICROSOFT WORD MICROSOFT EXCEL
Operating System:	WINDOWS 7
System Used:	DELL INSPIRON N5010 CORE i3

Abstract

The Alliance Rules are used for error identification and revealing errors in a data warehouse. These Alliance Rules are based on the perception of Mathematical Association Rules for detecting corrupt data in datasets. Data cleansing, also known as data scrubbing. is the course of action ensuring that a set of data is correct and accurate. During data cleansing, records are checked for correctness and uniformity, and either corrected, or deleted as required. Furthermore to deal with data cleansing, a manual work is evidently not a good alternative as it is painstaking, time consuming and requires a huge amount to be spent during the entire operation. The automation of the process of data cleansing for large datasets may be a good option with practical approach, which is also cost effective, time saving, and provides excellence level of data in a dataset. The role of application of Alliance Rules to the problem of data cleansing and automatically identifying potential errors in datasets is vital and has gained significant attention from researchers in the recent years. These rules also identify the outliers in data. In addition algorithms for different data types present in a dataset are also proposed along with their implementation for the identification of potential errors. Data cleansing approaches have been compared and a deep literature survey is carried out as a little domain knowledge for automated data cleansing acquaintance.

Table of Contents

List of Tables

List of Figures

Chapter 1

INTRODUCTION

The purpose of this research is to emphasize the characteristics of the Alliance Rules based on the idea of Mathematical Association Rules for detecting corrupt data in datasets. The Association Rules are useful for handling the problem of data cleansing so that we can dig out automatically the potential faults in datasets which is very important and has gained major consideration from researchers in the recent years. This research done, particularly focuses on the given literature highlighting the Binary Association Rules like quantitative, ratio, generalized, constraint based, distance based and composite Association Rules. These rules classify outliers in data. In addition to this, different algorithms that find these rules and identify possible mistakes in data are projected as well.

1.1 Motivation and Challenges

The control and exploration of data is provoked by diverse goals. Data are figurative representation of statistics, i.e., evidences or objects from the real domain, portrayed by descriptive values. The entirety, accuracy and uniformity of any large dataset depend upon number of factors. Abnormality is a feature of data values that solidifies them as an incorrect picture of the real world. It might be an outcome of the incorrect dimensions, lethargic input behavior, oversights during gathering and sustaining facts and figures etc. It might likewise originate from misunderstandings in data examination or owing changes in the real world that are undetected or not mirrored by variations to the demonstrating data. A distinct type of irregularity is duplicity, i.e., composite records demonstrating the similar information or corresponding fragments of it. The entrance of data and completion generally faces problems both in nature simple and composite. A lot of efforts are brought about but a chance of mistakes rests as it is in large datasets. The problem to decrease errors in large datasets effects the organization due to greater working costs, reduced policy making, on the rise distrust evidences within the association and as a result, the distraction of emphasis of management from major issue, to somewhat an unimportant matter. This problem can be solved by cleaning the data by using some cleansing approach.

The manual work cannot be suggested as it is painstaking, laborious and needs an enormous expense to be consumed throughout the whole process. The mechanization of the procedure of data cleansing for huge datasets is a good idea with practical approach, which is also expense effective, less time and effort consuming , and delivers excellent standard of data in

a dataset. The researchers have given a serious attention to this problem and are working to deal these issues of datasets. When data is gathered from different sources, the main problem the researchers come across is data cleansing.

1.2 Background

The data cleansing approach which is also called data scrubbing, is the procedure of safeguarding the set of data which is correct and precise. Throughout the data cleansing process, the records are checked for correctness and uniformity, and are modified, or cleaned which is equally essential. The process of data cleansing can be applied within a single set of records, or between numerous sets of data which need to be pooled together, or which will work as one unit. Basically data cleansing means to go through a set of records and authenticate their correctness. The spelling mistakes are removed, mislabeled data is appropriately categorized and filled, and lacking or erroneous entries are done. Data cleansing procedures frequently eliminate out-dated or not-recovered records, so that they do not take up space and form the basis of incompetent processes.

A person requires data cleansing for improving the data quality. The quality of the data can be enhanced by eliminating unreliable data, eradicating duplicate values and re-indexing current data in order to attain the precise and accurate database. Data cleansing is also compulsory when the data is amalgamated from various parent databases. Data cleansing can be achieved substantially or through explicit software programs.

1.2.1 Data Warehouse

A data warehouse is a consolidated source that stocks data from various information sources and alters them into a shared, multidimensional data model for efficient questioning and examination. As conferred by Inmon, a well-known author for a number of data warehouse books, A data warehouse is a:

1. Subject-based
2. Assimilated
3. Time-varying
4. Non-volatile

range of data required in the maintenance of organization's judgment making practice. [21]

It stores the transactional as well as the chronological data. It differentiates examination capacity from transaction work load and supports an association to combine data from various sources in order to support organization's inference making and development. In a data warehouse, the data should be consistent, correct and precise. Amendments have a great influence on developments. The quality of data deteriorates due to continuous updates, which leads to depletion of assets like time, currency and human power, etc. It also disturbs the data mining practice.

- **Components of a Data Warehouse**

A data warehouse system includes three modules:

1. A Database
2. A Database Interface.
3. A Query Tool

The data of the warehouse databases is extracted from existing operational databases. To make it useful and result oriented, the warehouse system rearranges the data.

A warehouse database can be updated by database interface facilities on daily, weekly, monthly or yearly basis. In order to shift the data from working database to the data warehouse, these interfaces are used.

The query generation is supported by the query tools for making presentations, reports and helping data specialists by permitting administrators and other user's real time access to the data warehouse database.

- **Data Warehouse Architecture**

The architecture is the correlation between the parts of a structure. The figure1.1. shows the architecture of a Data Warehouse.

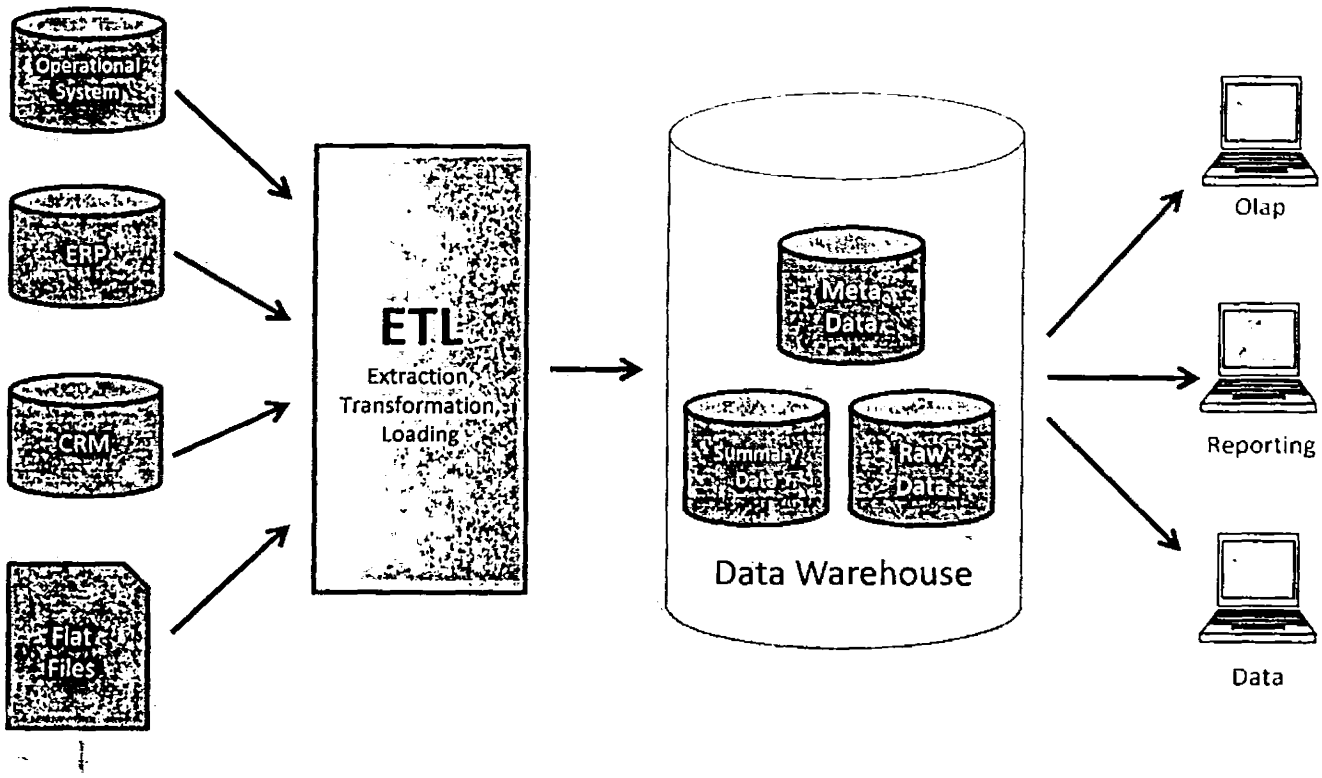


Figure 1.1: Data Warehouse Architecture. [21]

The tools for mining data from multiple working data sources (databases), are included in a data warehouse architecture as shown in figure 1.1. The key strength of these tools is to clean the data, then convert the data, and lastly stocking the data into the data warehouse. It also helps for infrequently refreshing the warehouse to reproduce apprises at the sources and to move data from the warehouse, possibly on to slower backups.

In addition to the main data warehouse, there are a number of departmental data marts. One or added warehouse servers are used to store and organize the data in the warehouse and data marts. The multidimensional notions of data can be checked by the help of numerous front-end tools like " Query Tools", "Report Writers", "Analysis Tools" and "Data Mining Tools". At the end repository is used to stock and organize the metadata for monitoring and administration of the warehouse system.

- **Data Mining**

Data Mining or knowledge discovery is the computer-assisted procedure of digging information from end to end and investigating massive groups of data and then drawing out the good judgment of the data. Data mining is a time intolerable and expensive procedure. Due to out of order data, wrong results will be produced. The tools used for data mining predict the performances and upcoming developments, letting industries to make information driven conclusions. Data mining tools are used to answer corporate queries that conventionally stood time unbearable to determine.

1.2.2 Data Mart

The Data Mart is a subgroup of the Data Warehouse, typically concerned with specific commercial line.

"Data Marts are analytical data stores designed to spotlight on specific business function for a specific community within an organization. A data mart may have tens of gigabytes of data rather than hundreds of gigabytes for the entire enterprise." [21]

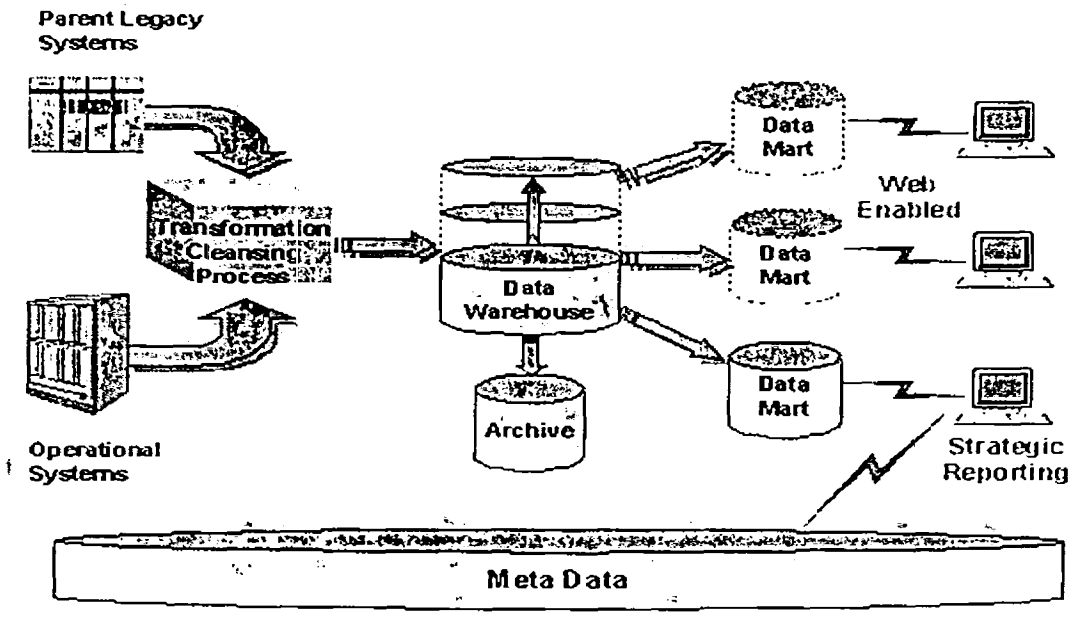


Figure 1.2:Architected Decision Support System. [22]

A DW is a dominant storehouse which can be actually dispersed; DM is a data store that may or may not be an outcome of a data warehouse. An organization may have one or more data marts; each data mart is associated to one or more professional tasks for which it was planned. It is promising that the data marts are dependent on other data marts of the same organization.

Generally the associated data marts are considered using common evidences and measurements. The professional unit is supposed to be the property owner of the data mart comprising all the hardware, software and data.

1.2.3 Extract, Transform and Load

The terms Extract, Transform and Load describe a collection of tools that assist in safeguarding the data that has latest records before being entered into the data warehouse. The ETL implements the subsequent main jobs:

1. Acknowledgment of significant, correct and interrelated evidences at the source side.
2. Data mining.
3. Transferring this information to the data staging Area (DSA).
4. Transformation of the facts taken from numerous sources into a common and standard layout.
5. Washing out the dataset, on the basis of database and commercial rules.
6. Stacking the data to the data warehouse and refreshing the data marts.

1.2.4 Data Cleansing

The process of Data Cleansing is defined as:

"Data cleansing or data scrubbing is the process of detecting and correcting (or removing) corrupt or inaccurate records from a record set, table, or database. Used mainly in databases, the term refers to identifying incomplete, incorrect, inaccurate, irrelevant etc. parts of the data and then replacing, modifying or deleting this dirty data."[21]

The procedure of Data Cleansing is likewise compulsory, when two systems of data need to work together. If an organization has two subdivisions, sharing many of the same customers, then it is mandatory for both the subdivisions to have corresponding data and data must be

complete and precise. If a customer updates his records in one office, the system should inevitably update the data at the other office to ensure competence and steadiness among the different datasets.

A Data Warehouse or the Data Marts deal with enormous amount of data, so mistakes can be produced during the process of insertion and deletion. The data cleansing process is applied for reducing these mistakes. A useful and meaningful information can only be extracted from the data when it is error free. Without undergoing through cleansing process, the system can produce useless results, leading to less knowledgeable work and complex difficulties.

A wide-ranging framework for data cleansing [18] is to:

- "Identify and determine error types."
- "Explore and categorize error instances."
- "Correct the errors."
- "Document the error instances."
- "Transform data entrance processes to reduce future mistakes."

The process of data cleansing consist of three major areas:

- Data Warehouse.
- Information supply in Databases.
- Complete excellence management.

1.2.5 The Need for Data Cleansing

The process of data cleansing is compulsory to recover the data quality by noticing and removing mistakes in the data. Classifying the faults in data, modifying precious records and eradicating bad records is a tiresome procedure but it is essential for decision making.

1.2.6 Data Excellence

The following points should be considered for maintaining the data quality. [21]. This includes:

Correctness: "The criteria of reliability, consistency and density."

Reliability: "The criteria of entirety and validity."

Comprehensiveness: "Achieved by correcting data, containing anomalies."

Legitimacy: "Approximated by the amount of data fulfilling integrity constraints."

Steadiness: "Concerns contradictions and syntactical analysis."

Regularity: "Directly related to irregularities."

Compactness: "The quotient of missing values in the data and the number of total values ought to be known."

Exceptionality: "Related to the number of duplications in the data."

1.3 Research Domain

A generalized algorithm is required for each data type present in the data warehouse for further research. Research completed by Arora et al [1] is in initial stage currently as no single algorithm solves the problem of error detection of different data types in an ordinary computerized way.

The research largely concentrates on the analysis of Alliance Rules based on the principle of data mining association rules to test out its viability for applying on a data warehouse for data cleaning. Foremost the goal of this research is to survey all the cleansing methodologies testified in preceding work.

It is also worth mentioning that it is not proposed to re-plan the previously proven cleansing principles for identifying errors in the datasets but a try to deliver and implement a generalized algorithm for different data types present in the data warehouse.

1.4 Proposed Approach

In order to deliver a solution for all the data types present in the data warehouse, the requirements are systematically studied and deep examination is brought about to develop and implement automated generalized algorithms utilizing the Alliance Rules based on the principal of data mining association rules. Main focus is to target the correctness of the

prevailing algorithm by the provision of a generalized mathematical model with enhanced precision.

1.5 Thesis Outline

The following structure is followed for the organization of the thesis:

Chapter 2: Literature Survey is brought about to inspect the prevailing data cleansing methodologies used in a data warehouse. Important outcomes are taken out and reconvened in the form of Research Outcomes Table at the end of this chapter.

Chapter 3: Requirement analysis provides an examination of the preceding and existing data cleansing requirements.

Chapter 4: System Design focuses on the design requirements and methodology of the proposed model utilizing the Alliance Rules in detail.

Chapter 5: This chapter will focus on the deployment environment in which the focused research scenario will be implemented. The flow control and pseudo code will be provided along with the implementation details.

Chapter 6: This chapter will provide the details of Testing and Performance Evaluation along with the testing software and hardware environment.

Chapter 7: Conclusion and results along with the achievements and recommendations for future research will be provided in this chapter.

Chapter 2

LITERATURE SURVEY

Research is done in order to know about the new knowledge or discovery. It helps the researchers to make it clear by providing the contextual knowledge for accepting the current information on the subject and brightens the implication for the new study.

The term Data Warehouse is used to define a system for gathering data, from one or more data sources, such as the databases of a transactional system. This data is moved into a central data location, the Data Warehouse, and later report this data in an accumulated way, to the commercial users in the association. These users can use this information for strategic decision making and analysis for end user applications. Data cleansing is mainly used to bring consistency to different sets of data merged from multiple sources.

This chapter will provide the literature survey. Foremost the main objective is to deliver a concrete base for what this research is demanding to examine. Section 2.1 will provide the research associated to data warehouse cleansing. Section 2.2 arranges the concept matrix in tabular form. In section 2.3 the complications found in literature survey are given. Section 2.4 classifies Research Findings. Lastly Section 2.5 provides the limitations of different data warehouse cleansing approaches explored in the previous sections.

2.1 Related Research

Data cleansing becomes mandatory when several databases are combined together for data mining. A number of problems arise when tuples belonging to the same entity are characterized in different layouts in the dissimilar datasets. Difficulties stated in preceding papers are conversed in detail.

- **Data Cleansing**

According to Arora et al [01] Data Cleansing is supposed to be the principal step for Knowledge Data Discovery. In this research various KDD and Data Mining schemes are carried out for cleansing data in a specific style.

- **Inappropriate Data Values**

According to Agrawal et al [20] data cleansing is the process that implements electronic procedures for exploring the databases and then checking out the incorrect and missing data for final removal.

- **Domain Format Errors**

The research by Muller et al [10] discusses and defines the Domain Format Errors as when the assumed value for an attribute does not support the possible field format.

- **Irregularities in Data**

As discussed by Muller et al [10], irregularities in data arise due to non-uniform usage of values, components and acronyms. This happens for example if one uses dissimilar currency formats to identify an employee's salary, the currency is not clearly computed with each value, and is supposed to be uniform.

- **Integrity Constraint Violations**

According to Muller et al [10], a limitation is a rule which demonstrates the information about the particular field and the values permitted for expressing certain facts, like age ≥ 0. When the records do not satisfy the applied rules or constraints, the problem of violating integrity constraint arises.

- **The Merge/ Purge Problem**

Duplicity occurs when dissimilar databases are amalgamated. Data cleansing techniques are applied basically to identify and remove these duplicates. The merge/purge problem described by Marcus et al [14], is not practical to evaluate each element of one database with the elements from the other; it also over burdens the main memory. It gathers all the records from the databases in one table and then allots a key for each record. The elements are sorted in the next stage with respect to their key and finally a fixed sized window is moved through the list. Every time the window is passed over the list, a new element is compared with the ones already existing in the window. The matching records are combined together. This process continues till the matching of all the records. The Merge/ Purge problem applies the Sorted Neighborhood Method.

- **Repetition of Data**

The literature [12, 14] examines the difficulty of data cleansing and classifies the possible mistakes in datasets. An impression of the small amount of existing literature concerning data

cleansing is provided. The research highlights the methods for error detection that go beyond integrity analysis. The research focuses on different date formats to cleanse the data. It does not identify the error of duplicity.

- **Illegal Records**

According to Muller et al [10], unacceptable records characterize by far the most difficult class of irregularity depicted in data groups. This type of issues are related to the errors that occur at the point of data entry into the source system. Typical errors involve misspelling, default values for mandatory attributes, wrong entry of data and misfielded values etc.

- **Misplaced Data values**

Misplaced values are the consequence of oversights while collecting the data [10]. This is a restriction infringement for the attributes having null values where a NOT NULL limitation is applicable. Furthermore there exist cases where user might not find such limitations thus permitting null values for an attribute. This problem can produce wrong results in KDD process.

- **Irregularities in Data**

According to Maletic et al [12], "Data Cleansing is the process of eliminating errors and inconsistencies in data, and solving the object identity problem." Association Rules are used for this purpose. To identify interesting patterns for fields like market basket analysis or census data, Association Rule Mining is quite helpful.

- **Data Manipulation**

The research findings by Karlsteen [09], assume that data is not changed throughout the merging procedure of data from multiple sources, so there exists a chance of stimulation for altered or wrong data entry while updating data in a data warehouse which has not been debated yet.

- **Problem of Data Mining**

This research highlights the problem of "Mining" a large compilation of basket type data dealings for association rules among sets of elements with least amount of stated sureness by Agrawal et al [20].

- **Disintegrating and Reuniting Data**

The term data cleansing is used to decompose and then reassemble the data. The key objective is to update a tuple with good quality data. This difficulty is not being discussed in the literature. The research has partially discussed the process management problem while keeping in mind the data quality issue, although other researchers have just focused on the definition of data quality.

- **Data Quality Problems**

The research by Karlsteen [09] finds out data quality difficulties when a user switches from the old running database environment to a new database environment. Some of the concepts work properly in older environments but produce some problems in the new environment.

S. #	Title	Year	Author	Research Method	Technique	Ref
04.	"A Unified Framework and Sequential Data Cleaning Approach for a Data Warehouse."	2008	J. Jebamalar Tamilselvi, Dr. V. Saravannan.	Framework	The data quality should be enhanced before the process of data mining so a framework is required to clean the data. The available frameworks do not provide all the services required for data cleaning, so this paper proposes and implements a new framework to comprise all the previous data cleaning methodologies and new techniques to decrease the complications of data cleaning process.	[4]
05.	"Quantitative Data Cleaning for Large Databases."	2008	Joseph M. Hellerstein.	A Survey on quantitative data attributes.	The paper presents an analysis on data cleansing methods that center the quantitative attributes of large databases. An arithmetical vision of data quality is also discussed with a stress on outlier detection and investigation methods based on statistics. The algorithms and implementations that best suit the large databases are also analyzed.	[5]
06.	"Duplicate Record Detection: A Survey."	2007	Ahmed K. Elmagarmid, Panagiotis G. Vassilios S.	Survey	A logical study is presented on Duplicate Record Detection. It also covers the duplicate record detection algorithms along with the methods for enhancing the proficiency and scalability of these algorithms.	[6]
07.	"Quality and Complexity Measures for Data Linkage and De-duplication."	2007	Peter Christen. Karl Goiser.	Algorithm	The research provides an impression of the issues implicated in assessing data linkage and de-duplication quality and difficulty. A range of procedures are debated and references are given for assessing the data quality during data linkage.	[7]

S. #	Title	Year	Author	Research Method	Technique	Ref
08.	"Efficient Algorithms for Grouping Data to Improve data Quality."	2006	Wing Ning Li, Johnson Zhang, Roopa Bheemavaram..	Algorithm.	The research introduces a record grouping problem called Transitive Closure Problem to speed up the process of improving data quality, and proposes algorithms to solve this problem. The proposed algorithms have been implemented efficiently in several ways.	[8]
09.	"Automation of Metadata Updates in a Time Critical Environment."	2006	Johan Karlsteen.	Algorithm.	The goal of this research is to design and implement an automated system for the metadata updates. The system would gather the necessary information about the weather stations and update the database with the new information. This system will save time and lower the risk for human errors. Data merging problem is solved by Brute Force Algorithm.	[9]
10.	"Problems, Methods and Challenges in Comprehensive Data Cleansing."	2003	Heiko Muller, Johann Christoph Freytag.	Survey	This paper classifies the different types of anomalies occurring in data that have to be removed and defines a set of quality criteria for clean data. The paper also evaluates and compares the already existing approaches for data cleansing with respect to the types and impurities handled and eliminated by them.	[10]
11.	"A Token-Based Data Cleaning Technique for Data Warehouse	2003	Timothy E. Ohanekwu, C.I. Ezeife.	Technique using Smart Tokens.	This paper presents a technique that removes the requirement of relying on match threshold by defining smart tokens which are used for identifying duplicates. This strategy also eliminates the need to use the entire long string records with multiple passes, for duplicate identification.	[11]

S. #	Title	Year	Author	Research Method	Technique	Ref
	Systems."					
12.	"Data Cleansing Beyond Integrity Analysis."	2000	Jonathon I. Maletic, Andrian Marcus.	Framework	In this paper the methods of error detection that are away from integrity analysis are discussed. These techniques comprise of pattern matching, statistical outlier detection, clustering, and data mining. The results of these methods after experimentation on real world datasets are also provided.	[12]
13.	"Automated Identification of Errors in Datasets."	2000	Jonathon I. Maletic, Andrian M.	Tool	An outline is provided by the paper on the present research approaches which are applied for solving the problem of data cleansing. It also provides a tool for computerized data cleansing of datasets.	[13]
14.	"Utilizing Association Rules for the Identification of Errors in Data."	2000	Andrian Marcus, Jonathon I. Maletic.	Algorithm	The addition of Boolean and Ordinal Association Rules is introduced to identify the outliers in data. An algorithm for finding these rules and identification of probable errors in datasets is also presented.	[14]
15.	"Data Cleansing: Problems and Current Approaches."	2000	Erhard Rahm, Hong Hai Do.	Survey.	The paper classifies the data quality issues which are pointed out by data cleaning and gives an overview of the main data cleansing approaches. A survey is presented for the classification of source (Single or Multiple) and location (Schema Level or Instance Level) errors to be solved.	[15]

S. #	Title	Year	Author	Research Method	Technique	Ref
16.	"Matching Algorithms within a Duplicate Detection System."	2000	Alvaro E. Monge.	Algorithm.	In this research the duplicate detection problems occurring at the record level are discussed. The paper reviews the methods to identify imprecise duplicate records in a database and provides some features that an algorithm requires in order to accomplish a successful duplicate record detection system.	[16]
17.	"ARKTOS: A Tool for Data Cleansing and Transformation in Data Warehouse Environments"	2000	Vassiliadis Vagena, S. Skiadopoulos, T. Sellis, N. Karayannidis.	Tool	The research presents a tool for demonstration and implementation of numerous data cleansing events which can also help in reducing the complexity and increasing the efficiency of data transformation activities.	[17]
18.	"Real World Data is Dirty: Data cleansing and the Merge/ Purge Problem."	1998	Hernandez, S. Stolfo	Programming Module.	The research presents an intelligent equation theory established to classify alike elements by a multipart, domain independent matching process.	[18]
19.	"The Impact of Poor Data Quality on the Typical	1998	Thomas C. Redman.	Summary.	The paper provides the summary of the influences of poor data quality on a typical enterprise.	[19]

S. #	Title	Year	Author	Research Method	Technique	Ref
	Enterprise."					
20.	"Mining Association Rules Between Sets of Items in Large Databases."	1993	Agrawal, Imielinski, Swami Arun.	Algorithm	The paper presents an effective algorithm to create association rules between different items of a database. The said algorithm unites "Buffer Management", "Novel Estimation" and "Pruning Techniques".	[20]

Table 2.1: Concept Matrix

2.3 Problems in the Existing Literature

S.#	Problem Reported	Recommended Solution	Ref
01.	Discovering unclean and out of order data in a data warehouse for effective decision making and higher operational costs.	This research proposed an algorithm for discovering errors and dirty data in the data warehouse because cleansing is the only solution for maintaining the quality level of a data warehouse. It utilizes Alliance Rules based on the concept of Mathematical Association Rules for data cleansing.	[1]
02.	How different kinds of information can be extracted from libraries which would eventually assist in decision making process. There exists a need to create large library databases for managing them electronically.	As the data of the libraries is growing with exponential rate so the best way to get information from this redundant data is the application of mining techniques. The DW technique helps in determining knowledge and refining facilities.	[2]
03.	When data is collected from various sources, the problem of "Duplicate Record Detection" arises.	The research presents a prototype which shows that the "Adaptive Duplicate Detection Algorithm" is the best way out for recognizing the duplicity.	[3]
04.	In the prevailing research, some of the data cleansing approaches are implemented for error detection. The similarity cleaning tools only check the field and record similarity, so these methods are good for some part of the cleaning progression not the all.	The paper proposes and implements a new framework in a sequential order which comprises of all the methods as a solitary data cleaning tool.	[4]
05.	The computer practitioners who manage large databases of quantitative information have to face the severe problem of errors in quantitative attributes in the form of outliers.	Due to the main focus on quantitative data, the statistical view of data quality is analyzed in this research along with the application of analytical methods based on vital statistics.	[5]

S.#	Problem Reported	Recommended Solution	Ref
06.	The paper addresses the problem of lexical heterogeneity. It happens when the records have identically designed fields throughout the database, but the data uses dissimilar demonstrations for the same object.	The paper presents a detailed survey on different techniques used for individual fields, approaches for refining the competence of the duplicate record detection procedures and a few profitable, off-the-shelf tools used in the business for duplicate record detection.	[6]
07.	During data preparation steps of different data mining developments the problem of identifying and eradicating duplicate records related to the same entity within the same datasets or several different datasets arises.	Different data linkage techniques are discussed and the matters concerned in quantifying equally the worth and complexity of linkage algorithms are presented.	[7]
08.	The data sources acquire huge sizes so efficient methods and algorithms are required to speed up the process of improving data quality.	To speed up the process of improving data quality, the research paper gives a two-step process; in step one the possible candidate records are grouped together and in step two, each group is further examined and treated.	[8]
09.	Automation of metadata of an already running online database produces a number of problems. There exist a number of abnormalities in the data itself, if the database schema is not perfect so there will be a need to remove the syntactic and semantic errors.	The old databases should be purged in such a way that the chances of errors are reduced. Before starting work on the new system, there is a need to cleanse the old data.	[9]
10.	The presence of irregularities and anomalies in data causes abnormalities and reduces its effective operation. There is a need to classify these impurities for elimination.	This research paper evaluates and compares the existing approaches for data cleansing and presents a classification of numerous types of abnormalities happening in data that have to be removed and a set of quality criteria for the cleansed data is also defined.	[10]

S.#	Problem Reported	Recommended Solution	Ref
11.	It is hard to determine the best match score threshold as well as comparing long strings with a number of passes so there is a need of an efficient and accurate string matching approach.	This paper proposes the idea of smart tokens that can eliminate the need to rely on match threshold as well as the requirement to use the entire long string records with multiple passes, for checking duplicity.	[11]
12.	Problem is to automatically identify and elementize the potential errors in the datasets.	The error detection methods that are away from integrity analysis are presented. These approaches comprise of statistical outlier detection, clustering, pattern matching, and data mining techniques.	[12]
13.	The problem is to develop an efficient tool for data cleansing.	The research presents a tool for computerized data cleansing of datasets which is domain independent and establishes the first part in a recommended framework for automated data cleansing.	[13]
14.	The problem is to clean and automatically identify potential errors in datasets.	A new addition to the Boolean Association Rules, Ordinal Association Rules, that incorporate ordinal relationships among data items, are introduced to identify the outliers in data.	[14]
15.	There is a need to classify the major data quality problems to be resolved by the process of data cleaning and transformation of data.	The research provides the classification of data quality problems in data sources distinguishing among single and multi-source as well as between schema and instance level problems. An overview of commercial data cleaning tools is also provided.	[15]
16.	When data is gathered from different sources, a number of problems are faced as the recognition of multiple illustrations of a single entity, detection of mistakes found during data entry, incorrect acronyms, or alterations in the	An algorithm is presented to notice estimated duplicate records in a database and present the properties that a pair-wise record matching algorithm ought to have for an effective identical record detection system.	[16]

S.#	Problem Reported	Recommended Solution	Ref
	schemas of records from multiple databases.		
17.	There is a problem to handle the efficiency and complexities of the ETL process for pulling out the data from numerous sources, their clean-up, modification and inclusion into a data warehouse.	A tool "ARKTOS" is developed to provide the graphical as well as declarative services, dimension of data quality through excellence factors and optimized implementation of complicated sequences of data transformation and cleaning tasks.	[17]
18.	The problem of merging multiple databases for information regarding common entities is regularly encountered in KDD and decision support applications. This problem is termed as the Merge/Purge problem and is difficult to fix equally in scale and correctness.	The Sorted Neighborhood Method is proposed as a solution to the merge/purge problem. It can be summarized in the following three steps: 1. Generate Keys, 2.Sort Data, 3.Merge.	[18]
19.	While implementing data quality programs, the practitioners have to face the impacts of poor data quality.	The paper discusses different issues which the practitioners have to face like dissatisfaction of the customer, amplified price, uncreative choice making, administrative uncertainty, problems in bringing into line the enterprise and the possession issues.	[19]
20.	The problem of "Mining" a large compilation of basket type data type transactions for association rules among sets of items with least amount of specified confidence is discussed.	The presented algorithm was quite efficient on sales data taken from a vending corporation. The approximation process presented higher exactness and the pruning techniques were able to crop out a very large part of item sets devoid of measurement.	[20]

Table 2.2: Problems In the Existing Literature

2.4 Research Outcomes

S. #	Research Outcomes	Ref
01.	Difficulties and faults related to the 'name' field are debated.	[1]
02.	The information revealed from one library should be different from the other library according to the user characteristics so there is a need of suitable research to be familiar with the requirements of Urban and Rural membership.	[2]
03.	"Adaptive Duplicate Detection Algorithm" is much better in performance as compared to other algorithms in terms of correctness but not in competence as it takes additional time for implementation because of both the activities of strict and accurate match.	[3]
04.	This framework presents six steps to clean data in a sequential order as Attributes Selection, Tokens Formation, Clustering, Similarity Computation, Elimination and Merge. It is valid for a relational database. It should be drawn-out for other databases.	[4]
05.	Outlier detection in a quantitative data, should be a human driven process with an analyst using their judgment and domain knowledge to validate information provided by the algorithms. If an analyst finds that a dataset is not normal then he should be careful to choose outlier approaches that accommodate their non-normality.	[5]
06.	The lack of consistent, large-scale benchmarking of datasets blocks the new techniques to emerge. A new direction for future research is to build up techniques by merging statistics and machine learning.	[6]
07.	The quality of data should be calculated by the precision-recall or F-measure graphs instead of single mathematical values.	[7]
08.	The number of tuples fed to the analysis tools are reduced through transitive closure handling by combining related records into smaller groups. By using numeral pairs as record identifiers the performance of transitive closure computation can be enhanced.	[8]
09.	Research has to be done on intelligent string matching functions.	[9]
10.	Data Cleansing is defined as an arrangement of operations planning to boost up the overall data quality of a data collection. Data Cleansing is extremely explorative and domain dependent.	[10]
11.	The smart tokens are related to domain-independent data cleaning . By using smart tokens as warehouse identifiers the process of incremental cleaning can be improved. This approach should be applied on un-structured and semi-structured data as a future work.	[11]

S. #	Research Outcomes	Ref
12.	There is a need to design and build useful software tools to update the data cleansing practice. Dissimilar approaches to identify errors should be incorporated.	[12]
13.	Using outlier detection procedures to identify possible errors in the data is different from the existing data cleansing tools. Further research is required on error correction issue along with extensive domain specific knowledge.	[13]
14.	Association rule mining ascertains to be supportive in finding not only the interesting patterns for fields such as market basket analysis or census data, but also patterns that expose errors in other kinds of datasets.	[14]
15.	Future work is required on the application of the best language approach for both schema and data transformations. Data cleaning is essential for data warehousing as well as for query handling on dissimilar data sources, for example in web based systems.	[15]
16.	Domain independence ought to be attained. The algorithm must be effective for checking the match or non-match between records.	[16]
17.	1. An impact analyzer should be designed to show how changes take place in the definition of a table or an activity influences other tables of a data warehouse. 2. A metadata repository should be connected to utilize its enhanced query activities. 3. An optimizer should be created to achieve improved efficiency.	[17]
18.	To enhance the merger of records, consider two options. The first one is to simply expand the scanning window size. Second is to perform several tracks of the Sorted-Neighborhood Method, using a different key each time with a small window. This policy is termed as the multi-pass approach.	[18]
19.	The quality of data can only be improved by overcoming the client's problems like enlarged price, fruitless decision making, the reduced ability to make policy, managerial uncertainty, problems in aligning the enterprise and the possession issues.	[19]
20.	The database competency to categorize queries must be improved.	[20]

Table 2.3: Research Outcomes

2.5 Limitations

- **"Alliance Rules for Data Warehouse Cleansing"**

Alliance Rules for Data Warehouse Cleansing are not refined systematically to gratify the necessities of cleansing of all the fields. Algorithm devised in this paper is limited to only "name" field of string data type. Alliance Rules show its applicability by taking into account a scenario but no application is provided.

- **"Managing Very large Databases and Data Warehousing."**

As the data of libraries is constantly growing with exponential rate, so the main problem of referencing the important information from large amount of redundant information of library can be reduced using mining techniques. Further study is required on the mining techniques to search through categorization of content of the library and acquirement of books through data mining by organizing large databases and data warehousing.

- **"Duplicate Record Detection for Database Cleansing."**

The "Adaptive Duplicate Detection Algorithm", produces improved outcomes in terms of correctness. But it takes extra time for execution due to careful and exact match activities so seems to lack competence.

- **"A Unified Framework and Sequential Data Cleaning Approach for A Data Warehouse."**

The prevailing data cleaning approaches covered only some data cleaning difficulties. But this new structure offers suitable and powerful algorithm to comprise all the techniques as a single data cleaning tool. The only limitation of this framework is that it is flexible for data in a relational database. It should be drawn-out for other databases.

- **"Quantitative Data Cleaning for Large Databases."**

To improve data quality by detecting and eliminating outliers in quantitative data, further research is required on exploratory data analysis methods, algorithms and arithmetical applications that can be effortlessly and competently implemented in large databases. Efficient and approximate computation of summary statistics in a single pass to detect incorrect and inconsistent data is a topic which needs further study.

- **"Duplicate Record Detection: A Survey."**

The lack of consistent, large-scale benchmarking of datasets blocks the new techniques to emerge. A new direction for future research is to build up techniques by merging statistics and machine learning.

- **"Quality and Complexity Measures for Record Linkage and De-duplication."**

The research presents an outline to reflect the issues involved in calculating data linkage and de-duplication quality and complexity. Further research is required on the overall complexity of linkage techniques keeping in mind the potential size of the datasets.

- **"Efficient Algorithm for Grouping Data to Improve Data Quality."**

Further research is required to speed-up and scale-up the projected algorithm by parallel and distributed processing using grid computers. A prototype should be developed for distributed algorithms for assessing the performance of the algorithm.

- **"Automation of Metadata Updates in a Time Critical Environment."**

Research must consider the intelligent string matching function. The presented approach is appropriate for one department only. Other departments must be considered for implementation. Further research is required on n-grams and significant functions. The proposed framework requires implementation.

- **"Problems, Methods and Challenges in Comprehensive Data Cleansing."**

A number of open problems and trials related to data cleansing are still under consideration. The areas which need further study are the organization of multiple, substitute values, the documentation of cleansing processes and the development of an appropriate framework for data cleansing.

- **"A Token-Based Data Cleaning Technique for Data Warehouse Systems."**

The technique of smart tokens is related with cleansing of domain independent data and these tokens can be used as identifiers for enhancing the process of incremental data cleansing. In future work, this technique must be applied on semi-structured and un-structured data.

- **"Data Cleansing: Beyond Integrity Analysis."**

A number of methods are devised for resolving the dilemma of automatic error detection in datasets. The investigation of groups of correlated fields should be planned. Integration of various methods to detect errors is not covered in this paper. High quality software tools should be developed for data cleansing process.

- **"Automated Identification of Errors in Datasets."**

Further research is required on error correction issue along with extensive domain specific knowledge.

- **"Utilizing Association Rules for the Identification of Errors in Data."**

A new addition of the Boolean Association Rules and ordinal association rules will be used to classify the outliers in data. An algorithm that finds these rules and classifies probable errors in data is presented. Association rule mining reveals to be helpful in classifying not only stimulating patterns for fields such as market basket analysis or survey data, but also the patterns that uncover errors in other kinds of datasets. Further research is required for generalization of the projected association rules on sets of data items .

- **"Data Cleaning Problems and Current Approaches."**

The research provides the classification of data quality problems in data sources differentiating single and multi-source as well as schema and instance level problems. To update both schema and instance level alterations, distinct applications are required for the plan and the application of the best language approach is compulsory.

- **"Matching Algorithms within a Duplicate Detection System."**

The research addresses the duplicate detection problem at the record level. A system to detect approximate duplicate records in a database is reviewed to have a successful duplicate detection system. Different record matching algorithms are presented in the paper for checking the equivalence of records from the sources. The proposed algorithm must achieve

domain independence and should point out the strength of match or non-match between the records.

- **"ARKTOS: A Tool for Data Cleansing and Transformation in Data Warehouse Environments".**

Additional functionality can be added to the "ARKTOS", to offer a user with more affluent transformation primitives. Some research issues are:

- An impact analyzer should be designed to show how changes in the definition of a table or an activity influence other tables or activities in the data warehouse.
- A metadata repository should be connected to utilize its enhanced query activities.
- An optimizer should be created to achieve improved efficiency.

- **"Real World Data is Dirty: Data cleansing and the Merge/ Purge Problem."**

The Sorted-Neighborhood Method is costly due to categorization phase, as well as the need to look for in large windows for extraordinary precision. Alternative methods based on data clustering modestly improve the process in time. Finally, the results reported here outline the basis of a Data Blade Module marketed as the Data Cleanser Data Blade.

- **"The Impact of Poor Data Quality on the Typical Enterprise."**

The paper provides the outline of the influences of poor data quality. It discusses different issues which the practitioners have to face like disappointment of the client, enlarged price, fruitless decision making, the reduced ability to make and perform policy, structural uncertainty, problems in aligning the enterprise and the possession issues. Further study is required to reduce the impact of above mentioned issues.

TH-8586

- **"Mining Association Rules between Sets of Items in Large Databases."**

The proposed algorithm is applied on auction's data taken from a vending corporation. The algorithm yields effective and accurate results on the given dataset. This process can further be enhanced with improvements of database capability using classification queries.

Chapter 3

REQUIREMENT ANALYSIS

Requirements analysis is the procedure of finding out the user potential for a new and customized invention. These requirements are ought to be confirmed, pertinent and complete. In software engineering, these requirements are termed as functional specifications. It is a significant feature of project management.

To bring about the requirement analysis, an understanding of data cleansing practice is necessary. It is an important issue while amalgamating multiple working data sources in a single system. The databases are dispersed across an enterprise, and search engines are used to build a Data Warehouse. This research needs suitable platforms for cleansing purposes, to lessen human reliance and produce efficient algorithms. The only way out is to clean the data. The major issue of data cleaning is "Duplicity", which is emphasized in this research.

3.1 Requirements for Data Cleansing

Following requirements are the prerequisite for data cleansing.

- **Precision**

Precision of data is the extent to which data suitably reflects the real world object or an event being described.

- **Competence**

Data Competence denotes the capability of many procedures applied to data such as storage, right to use, purifying, distribution, etc., and the extent to which these procedures lead to the favored results inside the source restriction.

- **Comprehensiveness**

Data comprehensiveness is an aspect which discovers all the desired data required to meet the current and the upcoming difficulty present in the data sources.

- **Mechanization**

To reduce human intervention , the complete automation of the system is required. By decreasing the human interference, high degree of mechanization and precision can be achieved.

- **Uniformity**

The redundancy should be removed for decreasing the occurrence of faulty data. If a single instance of a record is considered in a database, then a modification will make the new value instantaneously accessible to all the users. If multiple occurrences of the tuples are considered, the system must guarantee that all the copies of the records are retained uniform.

- **Individuality**

Individuality means a unique collection without duplicate values.

- **Domain Independency**

It is data transparency that matters for a centralized database. It advises the confrontation of the user applications to mark modifications in the meaning and organization of data.

3.2 Constraint Investigation over Important Problem Circumstances

In this section the requirement analysis based on critical problem scenarios is given.

- **"Alliance Rules for Data Warehouse Cleansing."**

This paper addresses the issues related with dirty data, entrance of dirty data and detection of dirty data in the data warehouse. It provides an algorithm for finding errors and dirty data in the datasets of an already existing data warehouse. The paper characterizes the Alliance Rules based on the concept of Association Rules for finding out the faulty data in the data warehouse. Algorithm proposed in this paper is limited to only "name" attribute of string data type. The research provides no implementation. The duplicity in the "name" field of the data warehouse has been purified and worked out, still there is a need to cover all other data fields. By using the concept of integer domain, the domain independency is acquired.

Referenced Paper	Constraint 1 Precision	Constraint 2 Completeness	Constraint 3 Mechanization	Constraint 4 Uniformity	Constraint 5 Domain Independency	Constraint 6 Individuality
"Alliance Rules for Data Warehouse Cleansing." [1]	Fulfilled	Not Fulfilled	Partially Fulfilled	Fulfilled	Fulfilled	Partially Fulfilled

Table 3.1: Req. Analysis of "Alliance Rules for Data Warehouse Cleansing."

- **"Managing very Large Databases and Data warehousing."**

This research classifies the alterations that had taken place in libraries due to electronic equipment and how the DW expertise could help them to find out the information and get improved services. This research reveals that organizing libraries by electronic means has brought about a change in the formation and organization of large library databases, so the libraries should prepare to develop these digital collections for decision making and provide services to suit the digital civilization. Information provided from one library should be different from the other one. Appropriate research to distinguish between the needs of rural and urban membership is not provided.

Referenced Paper	Constraint 1 Precision	Constraint 2 Completeness	Constraint 3 Mechanization	Constraint 4 Uniformity	Constraint 5 Domain Independency	Constraint 6 Individuality
"Managing Very Large Databases and Data Warehousing." [2]	Fulfilled	Partially Fulfilled	Partially Fulfilled	Fulfilled	Fulfilled	Fulfilled

Table 3.2: Req. Analysis of "Managing Very Large Databases and Data Warehousing."

- **"Duplicate Record Detection for Database Cleansing."**

When data is collected from various sources, the problem of duplicate record detection arises. This research is brought about on the evaluation of standard duplicate detection algorithms and it is realized that the "Adaptive Duplicate Detection Algorithm", produces improved results in terms of precision. But takes additional time for completion due to strict and approximate match undertakings.

Referenced Paper	Constraint 1 Precision	Constraint 2 Completeness	Constraint 3 Mechanization	Constraint 4 Uniformity	Constraint 5 Domain Independency	Constraint 6 Individuality
"Duplicate Record Detection for Database Cleansing." [3]	Fulfilled	Not Fulfilled	Fulfilled	Fulfilled	Fulfilled	Partially Fulfilled

Table 3.3: Req. Analysis of "Duplicate Record Detection for Database Cleansing."

- **"A Unified Framework and Sequential Data Cleaning Approach for Data Warehouse."**

This paper presents a framework for fulfilling the cleansing requirements for different kinds of data in the relational databases only. Services for data cleansing like attributes selection, formation of tokens, selection of clustering algorithm, selection of similarity function, elimination function selection and merge functions are delivered. It must be drawn-out for other types of databases.

Referenced Paper	Constraint 1 Precision	Constraint 2 Completeness	Constraint 3 Mechanization	Constraint 4 Uniformity	Constraint 5 Domain Independency	Requirement 6 Individuality
"A Unified Framework and Sequential Data Cleaning Approach for a Data Warehouse." [4]	Fulfilled	Not Fulfilled	Not Fulfilled	Partially Fulfilled	Not Fulfilled	Fulfilled

Table 3.4: Req. Analysis of "A Unified Framework and Sequential Data Cleaning Approach for a Data Warehouse."

- **"Quantitative Data Cleaning for Large Databases."**

The paper presents an analysis on data cleansing methods that center the quantitative attributes of large databases. An arithmetical vision of data quality is also discussed with a stress on outlier detection and investigation methods based on statistics. The algorithms and implementations that best suit the large databases are also analyzed.

Referenced Paper	Constraint 1 Precision	Constraint 2 Completeness	Constraint 3 Mechanization	Constraint 4 Uniformity	Constraint 5 Domain Independency	Constraint 6 Individuality
"Quantitative Data Cleaning for Large Databases." [5]	Fulfilled	Fulfilled	Not Fulfilled	Partially Fulfilled	Fulfilled	Fulfilled

Table 3.5: Req. Analysis of "Quantitative Data Cleaning for Large Databases."

- **"Duplicate Record Detection: A Survey."**

A logical study is presented on the text of duplicate record detection. The research also covers the duplicate record detection algorithms and methods for improving the competence and scalability of these algorithms. An attractive direction for future research is to build up techniques by merging machine learning and statistics.

Referenced Paper	Constraint 1 Precision	Constraint 2 Completeness	Constraint 3 Mechanization	Constraint 4 Uniformity	Constraint 5 Domain Independency	Constraint 6 Individuality
"Duplicate Record Detection: A Survey." [6]	Fulfilled	Fulfilled	Not Fulfilled	Fulfilled	Fulfilled	Partially Fulfilled

Table 3.6: Req. Analysis of "Duplicate Record Detection: A Survey."

- **"Quality and Complexity Measures for Record Linkage and De-duplication."**

The paper provides an outline of the problems involved in measuring data linkage and de-duplication quality and difficulty. Different procedures are debated and clearances are given on how to assess the data quality while linking data. Further research is required on the overall complexity of linkage techniques keeping in mind the potential size of the datasets.

Referenced Paper	Constraint 1 Precision	Constraint 2 Completeness	Constraint 3 Mechanization	Constraint 4 Uniformity	Constraint 5 Domain Independency	Constraint 6 Individuality
"Quality and Complexity Measures for Data Linkage and De-duplication." [7]	Fulfilled	Fulfilled	Not Fulfilled	Partially Fulfilled	Fulfilled	Partially Fulfilled

Table 3.7: Req. Analysis of "Quality and Complexity Measures for Data Linkage and De-duplication."

- **"Efficient Algorithm for Grouping Data to Improve Data Quality."**

The research introduces a record clustering problem called Transitive Closure Problem to improve data quality. The planned algorithms have been applied competently in different ways. The further research is required to speed-up and scale-up the proposed algorithms by

parallel and distributed processing using grid computers, and developing a prototype of a distributed algorithm to assess the performance of the algorithms empirically.

Referenced Paper	Constraint 1 Precision	Constraint 2 Completeness	Constraint 3 Mechanization	Constraint 4 Uniformity	Constraint 5 Domain Independency	Constraint 6 Individuality
"Efficient Algorithms for Grouping Data to Improve data Quality." [8]	Fulfilled	Fulfilled	Fulfilled	Partially Fulfilled	Fulfilled	Fulfilled

Table 3.8: Req. Analysis of "Efficient Algorithms for Grouping Data to Improve data Quality."

- **"Automation of Metadata Updates in a Time Critical Environment."**

The main objective of this research is to project and apply an automated system for the metadata updates. The system would collect the desirable material about the weather stations and update the database with the new facts. By the help of this system time will be saved and the risks of errors produced by humans will be lowered. Brute Force Algorithm solves the merging problem. Further research is required on intelligent string matching function. The projected method is appropriate for one department only. It can be drawn-out for other departments.

Referenced Paper	Constraint 1 Precision	Constraint 2 Completeness	Constraint 3 Mechanization	Constraint 4 Uniformity	Constraint 5 Domain Independency	Constraint 6 Individuality
"Mechanization of Metadata Updates in a Time Critical Environment." [9]	Fulfilled	Fulfilled	Not Fulfilled	Partially Fulfilled	Not Fulfilled	Not Fulfilled

Table 3.9: Req. Analysis of "Automation of Metadata Updates in a Time Critical Environment."

- **"Problems, Methods and Challenges in Comprehensive Data Cleansing."**

This paper classifies the different types of irregularities happening in data that have to be removed and defines a set of quality benchmarks for clean data. The paper also evaluates and compares the already prevailing methods for data cleansing with reference to the categories and abnormalities controlled and eradicated by them. The researchers mostly fear from the

management of multiple, alternative values, the management and documentation of performing cleansing operations as well as the specification and development of a suitable framework supporting data cleansing.

Referenced Paper	Constraint 1 Precision	Constraint 2 Completeness	Constraint 3 Mechanization	Constraint 4 Uniformity	Constraint 5 Domain Independency	Constraint 6 Individuality
"Problems, Methods and Challenges in Comprehensive Data Cleansing." [10]	Not Fulfilled	Not Fulfilled	Not Fulfilled	Fulfilled	Not Fulfilled	Not Fulfilled

Table 3.10: Req. Analysis of " Problems, Methods and Challenges in Comprehensive Data Cleansing."

- **"A Token-Based Data Cleaning Technique for Data Warehouse Systems."**

A procedure that eliminates the requirement of relying on matching standards and for identifying duplicates by defining smart tokens is presented. This plan also removes the need to use the entire long string records with multiple passes, for checking duplicity. Future work is scoped out for applying this token-based cleaning approach on un-structured and semi-structured data.

Referenced Paper	Constraint 1 Precision	Constraint 2 Completeness	Constraint 3 Mechanization	Constraint 4 Uniformity	Constraint 5 Domain Independency	Constraint 6 Individuality
"A Token-Based Data Cleaning Technique for Data Warehouse Systems."[11]	Fulfilled	Fulfilled	Fulfilled	Fulfilled	Not Fulfilled	Partially Fulfilled

Table 3.11: Req. Analysis of " A Token-Based Data Cleaning Technique for Data Warehouse Systems"

- **"Data Cleansing Beyond Integrity Analysis."**

In this paper the methods of error detection that are away from integrity analysis are discussed. These methods include clustering, statistical outlier detection, pattern matching and data mining techniques. The results obtained after experimentation of these methods to a dataset are also given. Incorporation of various methods to detect errors is not covered in this paper. It has become important to design and build high quality, useful software tools.

Referenced Paper	Constraint 1 Precision	Constraint 2 Completeness	Constraint 3 Mechanization	Constraint 4 Uniformity	Constraint 5 Domain Independency	Constraint 6 Individuality
"Data Cleansing Beyond Integrity Analysis." [12]	Not Fulfilled	Not Fulfilled	Not Fulfilled	Fulfilled	Not Fulfilled	Not Fulfilled

Table 3.12: Req. Analysis of "Data Cleansing Beyond Integrity Analysis."

- **"Automated Identification of Errors in Datasets."**

An outline is provided by the paper on the present research approaches which are applied for solving the problem of data cleansing. It also provides a tool for computerized data cleansing of datasets. Further research is required on error correction issue along with extensive domain specific knowledge.

Referenced Paper	Constraint 1 Precision	Constraint 2 Completeness	Constraint 3 Mechanization	Constraint 4 Uniformity	Constraint 5 Domain Independency	Constraint 6 Individuality
"Automated Identification of Errors in Datasets." [13]	Fulfilled	Not Fulfilled	Not Fulfilled	Fulfilled	Fulfilled	Not Fulfilled

Table 3.13: Req. Analysis of "Automated Identification of Errors in Datasets."

- **"Utilizing Association Rules for the Identification of Errors in Data."**

The addition of Boolean and Ordinal Association Rules is introduced to identify the outliers in data. An algorithm for finding these rules and identification of probable errors in datasets is also presented. Further research is required for generalization of the projected association rules on sets of data items .

Referenced Paper	Constraint 1 Precision	Constraint 2 Completeness	Constraint 3 Mechanization	Constraint 4 Uniformity	Constraint 5 Domain Independency	Constraint 6 Individuality
"Utilizing Association Rules for the Identification of Errors in Data." [14]	Fulfilled	Fulfilled	Not Fulfilled	Fulfilled	Fulfilled	Fulfilled

Table 3.14: Req. Analysis of "Utilizing Association Rules for the Identification of Errors in Data."

- **"Data Cleaning Problems and Current Approaches."**

The research provides the classification of data quality problems in data sources differentiating between single and multi-source as well as between schema and instance level problems. To upkeep both schema and data alterations, distinct efforts are required on the plan and application of the best language approach is a prerequisite.

Referenced Paper	Constraint 1 Precision	Constraint 2 Completeness	Constraint 3 Mechanization	Constraint 4 Uniformity	Constraint 5 Domain Independency	Constraint 6 Individuality
"Data Cleaning: Problems and Current Approaches." [15]	Not Fulfilled	Fulfilled	Not Fulfilled	Fulfilled	Not Fulfilled	Not Fulfilled

Table 3.15: Req. Analysis of "Data Cleaning: Problems and Current Approaches."

- **"Matching Algorithms within a Duplicate Detection System."**

The research addresses the duplicate detection problem at the record level. A system to detect approximate duplicate records in a database is reviewed to have a successful duplicate detection system. Different record matching algorithms are presented in the paper for checking the equivalence of records from the sources. This algorithm must achieve domain independence and should point out the strength of match or non-match between the records.

Referenced Paper	Constraint 1 Precision	Constraint 2 Completeness	Constraint 3 Mechanization	Constraint 4 Uniformity	Constraint 5 Domain Independency	Constraint 6 Individuality
"Matching Algorithms in a Duplicate Detection System" [16]	Not Fulfilled	Not Fulfilled	Not Fulfilled	Not Fulfilled	Not Fulfilled	Fulfilled

Table 3.16: Req. Analysis of " Matching Algorithms in a Duplicate Detection System."

- **"ARKTOS: A Tool for Data Cleansing and Transformation in Data Warehouse Environments".**

The paper presents a tool for demonstration and execution of several data cleansing accomplishments which can also help in reducing the complexity and increasing the

competence of data transformation activities. Extra features can be added to the "ARKTOS", to provide a user with richer conversion primitives. A few issues are:

- An impact analyzer should be designed to show how changes in the definition of a table or an activity influence other tables or activities in the data warehouse.
- A metadata repository should be connected to utilize its enhanced query activities.
- An optimizer should be created to achieve improved Competence.

Referenced Paper	Constraint 1 Precision	Constraint 2 Competence	Constraint 3 Mechanization	Constraint 4 Uniformity	Constraint 5 Domain Independency	Constraint 6 Individuality
"ARKTOS: A Tool for Data Cleansing and Transformation in Data Warehouse Environments" [17]	Not Fulfilled	Not Fulfilled	Not Fulfilled	Fulfilled	Not Fulfilled	Not Fulfilled

Table 3.17: Req. Analysis of" ARKTOS: A Tool for Data Cleansing and Transformation in Data Warehouse Environments."

- **"Real World Data is Dirty: Data cleansing and the Merge/ Purge Problem."**

The Sorted-Neighborhood Method is costly due to categorization phase, and the need to look for in large windows for high precision. Alternative methods based on data clustering modestly improve the process in time. Finally, the results reported here outline the basis of a Data Blade Module available for the Informix Universal Server and are marketed as the Data Cleanser Data Blade.

Referenced Paper	Constraint 1 Precision	Constraint 2 Completeness	Constraint 3 Mechanization	Constraint 4 Uniformity	Constraint 5 Domain Independency	Constraint 6 Individuality
"Real World Data is Dirty: Data cleansing and the Merge/ Purge Problem." [18]	Not Fulfilled	Not Fulfilled	Not Fulfilled	Fulfilled	Not Fulfilled	Not Fulfilled

Table 3.18: Req. Analysis of "Real World Data is Dirty: Data cleansing and the Merge/ Purge Problem."

- **"The Impact of Poor Data Quality on the Typical Enterprise."**

The paper provides the outline of the influences of poor data quality. It discusses different issues which the practitioners have to face like disappointment of the client, enlarged price, fruitless decision making, the reduced ability to make and perform policy. structural uncertainty, problems in aligning the enterprise and the possession issues. Further study is required to reduce the impact of above mentioned issues.

Referenced Paper	Constraint 1 Precision	Constraint 2 Completeness	Constraint 3 Mechanization	Constraint 4 Uniformity	Constraint 5 Domain Independency	Constraint 6 Individuality
"The Impact of Poor Data Quality on the Typical Enterprise." [19]	Not Fulfilled	Not Fulfilled	Not Fulfilled	Fulfilled	Not Fulfilled	Not Fulfilled

Table 3.19: Req. Analysis of "The Impact of Poor Data Quality on the Typical Enterprise."

- **"Mining Association Rules between Sets of Items in Large Databases."**

The research presents an algorithm on sales data taken from a vending corporation. The algorithm produces well-organized and precise results on the given dataset. The procedure can further be upgraded with enhancement of database capability using classification queries.

Referenced Paper	Constraint 1 Precision	Constraint 2 Completeness	Constraint 3 Mechanization	Constraint 4 Uniformity	Constraint 5 Domain Independency	Constraint 6 Individuality
"Mining Association Rules Between Sets of Items in Large Databases." [20]	Fulfilled	Not Fulfilled	Not Fulfilled	Fulfilled	Not Fulfilled	Not Fulfilled

Table 3.20: Req. Analysis of "Mining Association Rules Between Sets of Items in Large Databases."

Chapter 4

SYSTEM DESIGN

The term System Design defines the plan, modules, interfaces and data for a system to gratify the design requirements. The main focus of this chapter is on the project requirements and methodology of the proposed model in detail.

To achieve accurate results from a data warehouse, it is necessary to enter correct data. Detection of dirty data is an important issue. Several techniques are available for data cleansing. The proposed model is going to focus on algorithms and their implementation to detect the errors in the dirty datasets of an already existing data warehouse. In order to devise an algorithm, an analytical procedure is assumed. The literature survey helps in understanding the problems and on the basis of this survey a findings table is created. The actual objective of this effort is to upkeep the proposed model by keeping in mind the survey findings and to improve the outcomes in terms of competence and accurateness as compared to the previous work done in the domain of duplicity detection.

The chapter is organized as: The design requirements are discussed in section 4.1 and the section 4.2 provides the details of the proposed algorithms.

4.1 Design Requirements

The design requirements obtained after requirement analysis are explained below to understand the proposed model.

4.1.1 Precision

The planned model must provide correct outcomes for diverse data types as the precise data is an important prerequisite for a commendable information system.

4.1.2 Competence

The planned model must be competent enough to create the preferred outputs within the source restrictions.

4.1.3 Uniformity

The planned model must change the data from one consistent state to another consistent state. Through the detection and removal of redundancy, the hazard of the presence of defective data can be reduced. If a single occurrence of a tuple is presented in a database, then an alteration will make the new value instantaneously open to all the users. If several instances

are considered, then the system must guarantee that all the copies of the record are kept constant.

4.1.4 Integrity

The constraints are applied to ensure reliability. The uniformity rules are not violated by the databases. These rules are applied on data items inside a single record or between different records. The suggested model should not violate the integrity of data.

4.1.5 Comprehensiveness

Data comprehensiveness is the dimension to find whether all the required data meets the current and upcoming difficulties in the data warehouse or not.

4.1.6 Domain Independence

Domain independency should be achieved. It has been attained by using the concept of integer domain which saves the memory as well.

4.1.7 Mechanization

To avoid human intervention, the basic requirement is to create the data cleansing tool completely automatic so that the manual participation is eliminated resulting in high degree of computerization and accurateness.

4.2 Proposed Model

The research proposes the mathematical representation and implementation of different algorithms for data cleansing of a data warehouse.

4.2.1 Data Alliance Rules

Data Alliance Rules [1] are constructed on the basis of mathematical association rules and are made clear as follows:

Suppose $F = \{f_1, f_2, f_3 \ldots\ldots fn\}$ is taken as a set of fields, where every field is a subgroup of a data mart. There are m DM's.

$DM = \{DM_1, DM_2 \ldots\ldots\ldots\ldots DM_m\}$

Let S = {Score(1), Score(2).....Score(n)} be a set of scores, where each Score(n)⊆ F.

Also the score set has integer as the numerical domain D (S∈ D) owing relationships defined in

D= {= equal, ≠ not equal, ≥ greater or equal}

Then

Score(1), Score(2) =>Score(1) op Score(2) where op ⊆ {≠, =, ≥}, is an alliance rule if,

1. Score(1) and Score(2) happen in at least p% of the n records, where p is the support of the rule.
2. In c% of the records, B =>Score(1) op Score(2) op T where Score(1), Score(2) are reference table scores and Data Warehouse scores correspondingly. (B implies the match type between two scores K1 and K2).
3. Assurance of threshold value is 50%. If 50% letters are not matched the word is guaranteed to be an outlier value.
4. On behalf of q-grams matching [1], rule 2 is used. The q-grams are the substrings of length q of a given string.

4.2.2 Recommended Algorithms

Algorithms are proposed for finding duplicity in different fields of a data warehouse. The base paper eliminates the duplicity in the name field only (no implementation is provided). It is to be applied on different data types. The mathematical model and implementation of the same algorithm (presented in the base paper) is provided for the name and address fields with slight modifications and new algorithms are devised and implemented for the rest of the fields like E-mail, Date of Birth and Passport _No.

The algorithm for the detection of errors in the string and alpha-numeric type data types of different data marts consists of the following three phases:

- Preprocessing
- Application of Alliance Rules
- Detection of Errors

- **Name Attribute**

In pre-processing, strings are transformed into integer values which are kept in a file for processing. Alliance Rules are used in the application stage. The duplicity is identified and stated in the third stage.

A. Preprocessing

Two different data marts are considered such that a record from one data mart say DM1 has to be checked and matched for duplicity with the records in another data mart say DM2. The strings and numbers are considered. These strings and numbers are first converted into numerical integer values. These converted integer values are kept in a file named Score_File and termed as Scores of the names.

The transformation is completed by the following formula:

$$| (\text{radix})^{\text{place value}} * \text{Face Value} | \bmod m$$

Where

Radix = A set of 67 characters (26 letters + 10 numbers+ 4 arithmetic operators (+,-,*, /) , 3 Relational Operators (=, <, >,), 24 Special Characters (".", ",", "#","@", "$", "%", "^", "&", "_","?", ";", ":", "~", "\", "|", " ' ", "(", ")", "{", "}", "[","]", "!"," " ",))

(Note: Letters are case in-sensitive)

Place value = The place value of letters is marked from right to left starting with 0.

m = Any large prime number.

Face value = The face value of each character is marked by the sequence number with which they arrive in the alphabetical order starting with 0-A-------66-" " ".

Character	Face Value	Character	Face Value
A	0	8	34
B	1	9	35
C	2	+	36
D	3	-	37
E	4	*	38
F	5	/	39
G	6	=	40
H	7	<	41
I	8	>	42
J	9	.	43
K	10	,	44
L	11	#	45
M	12	@	46
N	13	$	47
O	14	%	48
P	15	^	49
Q	16	&	50
R	17	_	51
S	18	?	52
T	19	;	53
U	20	:	54
V	21	~	55
W	22	\	56
X	23	|	57
Y	24	‘	58
Z	25	(	59
0	26	)	60
1	27	{	61
2	28	}	62
3	29	[	63
4	30	]	64
5	31	!	65
6	32	“	66
7	33		

Table 4.1 characters with Face Values

With the help of the given table, face values can be taken into account. The calculated values for each dataset are kept in tabular form in the score_file.

B. Alliance Rules Application

The algorithm for detecting duplicity in "Name Field" of a data warehouse is as follows:

1. Take a person's name from DM1.
2. Find out the count of words in the person's name and represent it by N.
3. Evaluate N+1 scores for the person's name each related to a word present in name. The $(N+1)^{th}$ score is the score of the initials of the person's name.
4. In Data Mart 2 (DM2), group the names which have the same value of N.
5. Calculate N+1 scores for all the names present in this group of DM2.
6. Match the last name scores Score(n) of the person name in DM1 with Score(n) of all the names in DM2 group and cluster all those names in DM2 that have similar score value for Score(n) and further reduce the size of the group and store it in a file called scoring file 1.
7. Now match Score(1) of the person name from DM1 & Score(1) of the person names from new DM2 cluster and store the further reduced group in scoring file2.

C. Duplicity Detection

Now there exist three cases for Score(1) matching .This score matching finally helps in detecting the duplicates.

Case 1: Perfect Match

a) **Single Entry Match**: No duplicity is resulted in case of single entry so no error is noticed.

b) **Multiple Entry Match:** For this case, match other scores of the name which are Score(2), Score(3) up to Score(n-1). If a single entry is resulted out, then there is no error, but if multiple entries are skimmed out, datasets are infected with duplicate records.

Case 2: No Match

If Score(1) does not match then match the scores of initials i. e. Score $_{(N+1)}$ now this can result in two conditions.

a) **Same person**

b) **Different person with same initials**

In this case check the values of (DOB+ Address). If the value matches then it is the same person then error is identified due to duplicity of values. But if it is not the same person then no error exists.

Case 3: None of the score matches from Score(N) to Score(N+1)

a) **That entry does not exist.**

b) **Entry exists with some errors in name.**

For entry with some errors in name, the concept of q-grams will be applied. Most appropriate length is 3.

(Note: Space in name is represented by_).

The complexity of the algorithm is $O(n^2)$

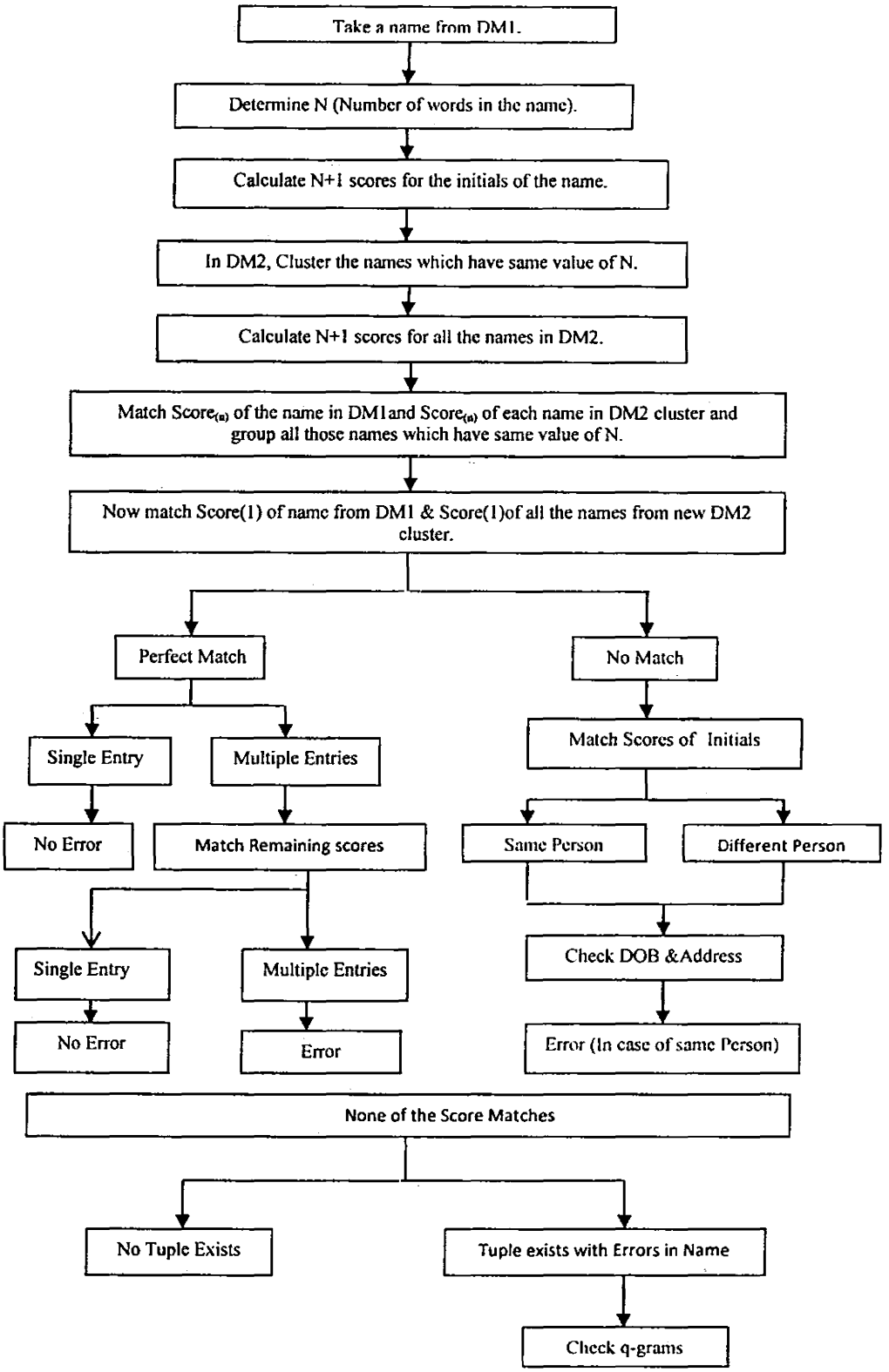


Figure 4.1Flow Chart of Algorithm for Name Attribute.

(Note: Slight modification in the above algorithm can make it more efficient.)

Modification

The above mentioned algorithm works efficiently with Western Names, but to make this algorithm more efficient for Local Names, some changes in the algorithm are to be made. In step 7, instead of matching Score(1), algorithm will match Score(2) first. Similarly there will be a minor change in the multiple entry match (Case 2) of the duplicity detection phase. The other scores of the name which are to be matched are Score(1), Score(3) up to Score(n-1).

- **Address Attribute**

In pre-processing strings are transformed into integer values which are kept in a file for processing. Alliance Rules are used in the application stage. The duplicity is identified and stated in the third stage.

A. Pre Processing

The strings are converted into numerical values by using the following relation and stored in a file for ready reference.

$$|(\text{radix})^{\text{place value}} * \text{Face Value}| \bmod m$$

B. Alliance Rules Application

The algorithm for detecting duplicity in address field of a data warehouse is as follows:

1. Take an address from DM1.
2. Calculate the word count in the address and represent it by N.
3. Evaluate N+1 scores for the address each related to a word present in the address. The $(N+1)^{th}$ score is the score of the initials of the address.
4. In Data Mart 2 (DM2), group the addresses which have same value of N.
5. Calculate N+1 scores for all the addresses in this group of DM2.
6. Match the last word's scores Score(n) of the address in DM1 & Score(n) of each address in DM2 group and cluster all those addresses in DM2 that have same score value for Score(n) and further reduce the size of the group. Store it in a file called scoring file 1.

7. Then match Score(2) of address from DM1 & Score(2) of addresses from the new DM2 cluster and store the further reduced group in scoring file2.

C. Duplicity Detection

Now there exist three cases for Score(2) score matching .This score matching finally helps in detecting the duplicates.

Case 1: Perfect Match

a) **Single Entry Match**: No duplicity is found in case of single entry so no error is noticed.

b) **Multiple Entry Match:** For this case, match other scores of the address which are Score(1), Score(3) up to Score(n-1). If a single entry is resulted out, then there is no error. But if multiple entries are skimmed out, datasets are infected with duplicate records.

Case 2: No Match

If Score(2) does not match then match the scores of initials i. e. Score $_{(N+1)}$, now this can result in two conditions.

a) **Same address**

b) **Different address with same initials**

In this case check the values of (CNIC & DOB).If the value matches then it is the same address then error is identified due to duplicity of values. But if it is not the same person then no error exists.

Case 3: None of the score matches from Score(n) to Score(n+1)

a) **That entry does not exist.**
b) **Entry exists with some errors in address.**

For entry with some errors in address, the concept of q-grams will be applied. Most appropriate length is 3.

(Note: Space in address is represented by_).
The complexity of the algorithm is $O(n^2)$.

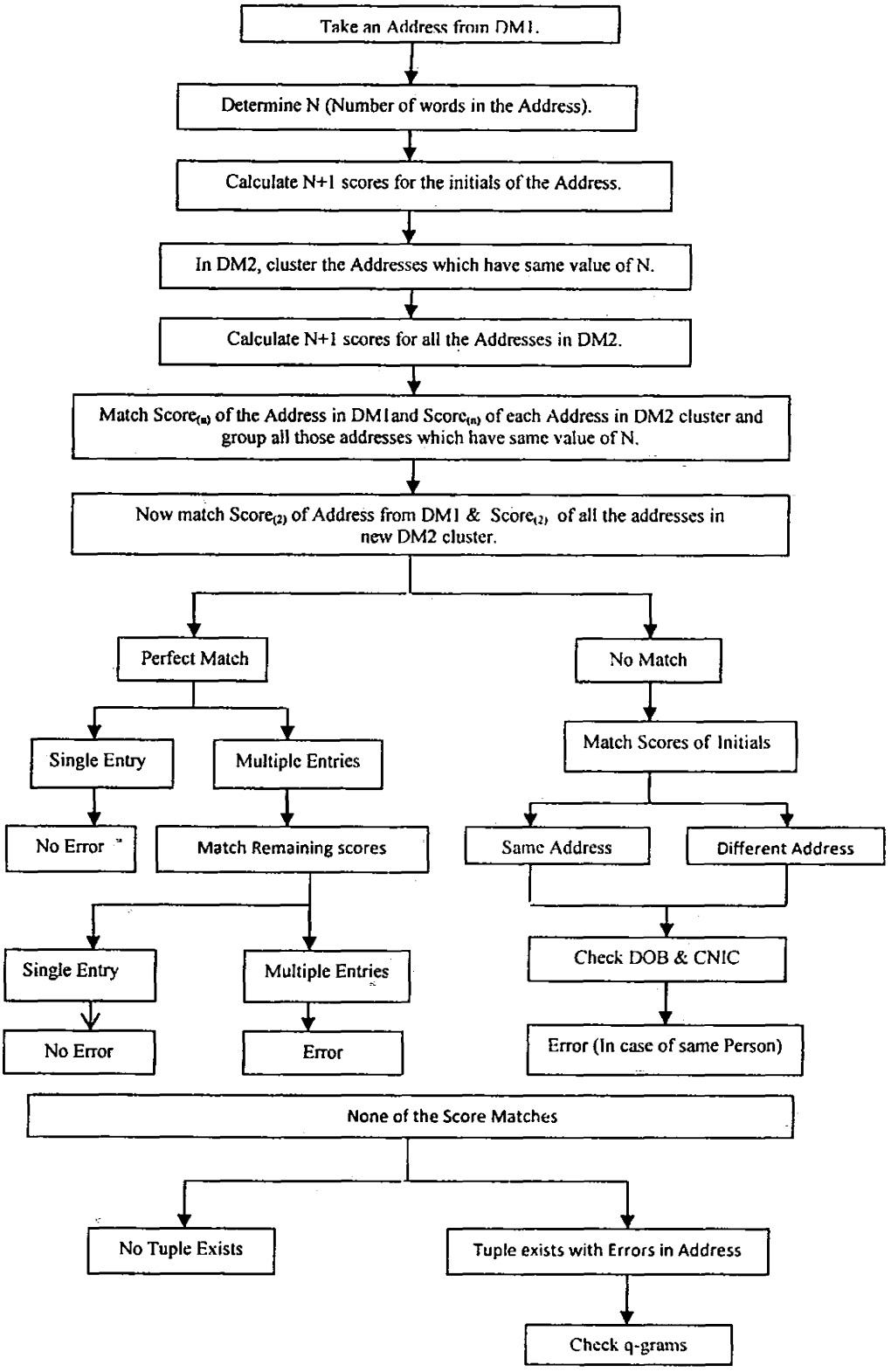


Figure 4.2Flow chart of Algorithm for Address Attribute.

- **E-mail Attribute**

In pre-processing, strings are transformed into integer values which are kept in a file for processing.

A. Pre Processing

The strings are converted into numerical values by using the following relation and stored in a file for ready reference.

$$|(radix)^{place\ value} * Face\ Value| \bmod m$$

B. Alliance Rules Application

The algorithm for detecting duplicity in E-Mail address field of a data warehouse is as follows:

1. Take an e-mail address from DM1.
2. Find out the number of words in the e-mail address. Let it be represented by N.
3. In Data Mart 2 (DM2), group the e-mail addresses which have same value of N.
4. Now divide the e-mail address in two parts i.e. e-mail_id and domain. The score of e-mail_id is denoted by $Score_{(id)}$ and score of domain is denoted by $Score_{(d)}$.
5. First of all match $Score_{(id)}1$ of e-mail address from DM1 with $Score_{(id)}1$ of all the e-mail addresses from DM2. Cluster the e-mail addresses in DM2 which have the same score value of $Score_{(id)}1$.
6. Now match the $Score_{(d)}1$ of DM1 with $Score_{(d)}1$ of all the e-mails present in DM2 cluster and cluster all those e-mail addresses in DM2 that have the same score value for $Score_{(d)}1$ and further reduce the size of the cluster. Store new cluster in scoring file1.
7. Now match $Score_{(d)}2$ to $Score_{(d)}n$ of e-mail address from DM1 and of all the e-mail addresses present in DM2 cluster and store the further reduced cluster in file called scoring file 2.

C. Duplicity Detection

Case 1: Perfect Match

a) **Single Entry Match**: There is no duplicity, in case of single entry, hence no error detected.

b) **Multiple Entry Matches**: For this case, match other scores which are $Score_{(id)}2$ to $Score_{(id)}n-1$. If a single entry is resulted out by matching all the scores then there is no

error but if multiple entries are skimmed out, the dataset is infected with duplicate records.

The complexity of the algorithm is $O(n^2)$.

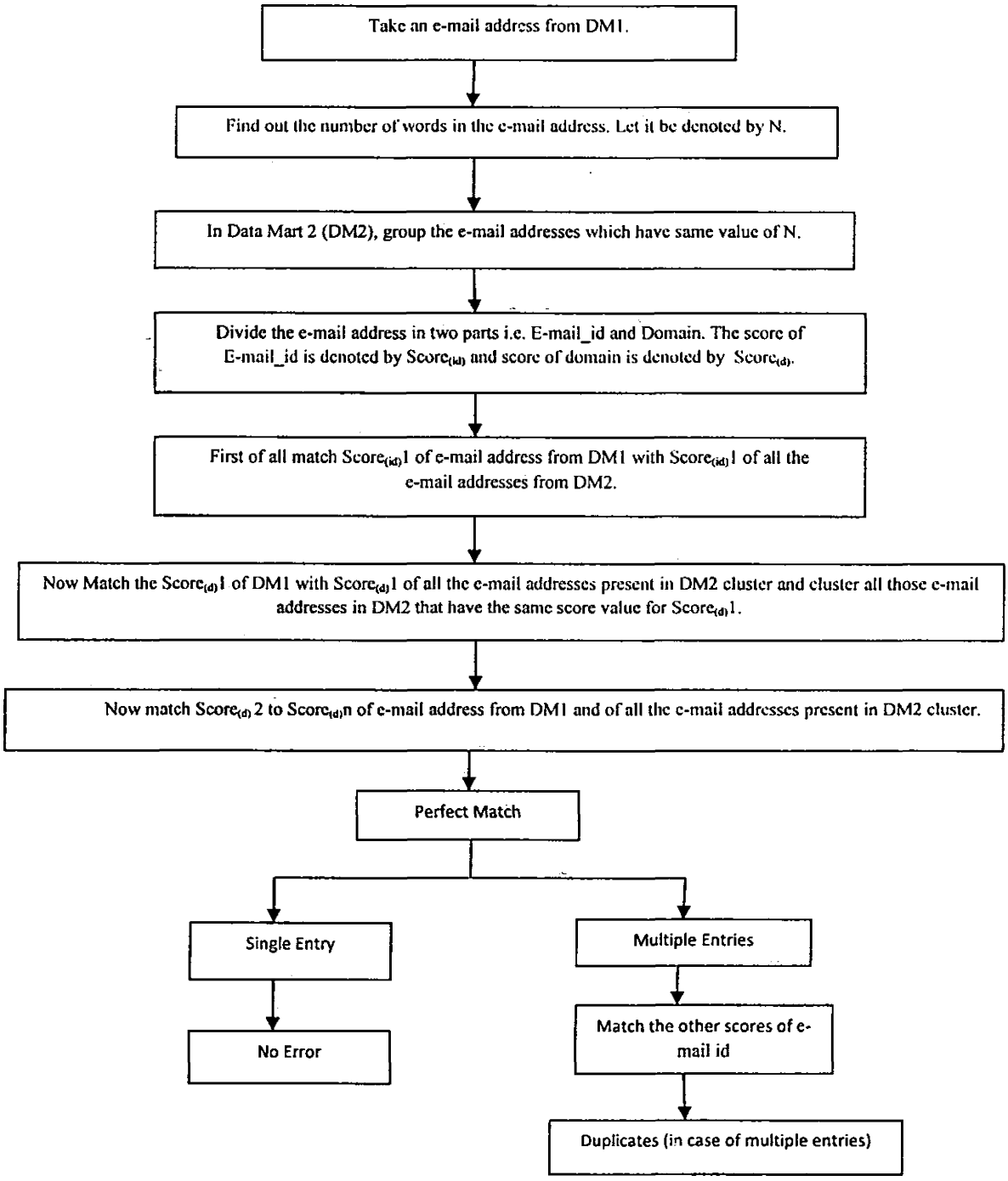


Figure 4.3: Flow Chart of Algorithm for E-mail Address Attribute

- **Date Attribute**

The dates are converted into integer values in the preprocessing step.

A. Pre processing

Dates are converted into numerical values by using the following relation and stored in a file for ready reference.

$$|(\text{radix})^{\text{place value}} * \text{Face Value}| \bmod m$$

B. Alliance Rules Application

1. Take a date from DM1.
2. A date consists of 3 parts, (Month, Day and Year,). Let us denote a Month as D1, Day as D2, and Year as D3.
3. First of all match the score of D3 (year) of date from DM1 with score of D3 of all the dates present in DM2 and cluster all those dates in DM2 that have the same result for D3 and form a cluster. Store this new cluster in a file called FIRST-MATCH.
4. Now match score of D1 (Month) of the date from DM1 with score of D1 of all the dates present in DM2 cluster and store the further reduced cluster in a file called SECOND-MATCH.
5. Now match the score of D2 (Day) of the date from DM1 with the scores of D2 of all the dates present in the new cluster of DM2. This matching will finally help in detecting the duplicates.

B. Duplicity Detection

For perfect match, consider the following two cases.

Case 1: In case of single entry, no duplicity exists so no error is noticed.
Case 2: If the result obtained is in the form of a single entry by matching all the values then no error is generated, but if multiple entries are skimmed out, then in this case check the values of (CNIC + Name). If the values match then it's the same person. Hence the error is identified as duplicity. If the values do not match then no error is found.
The complexity of the algorithm is $O(n^2)$.

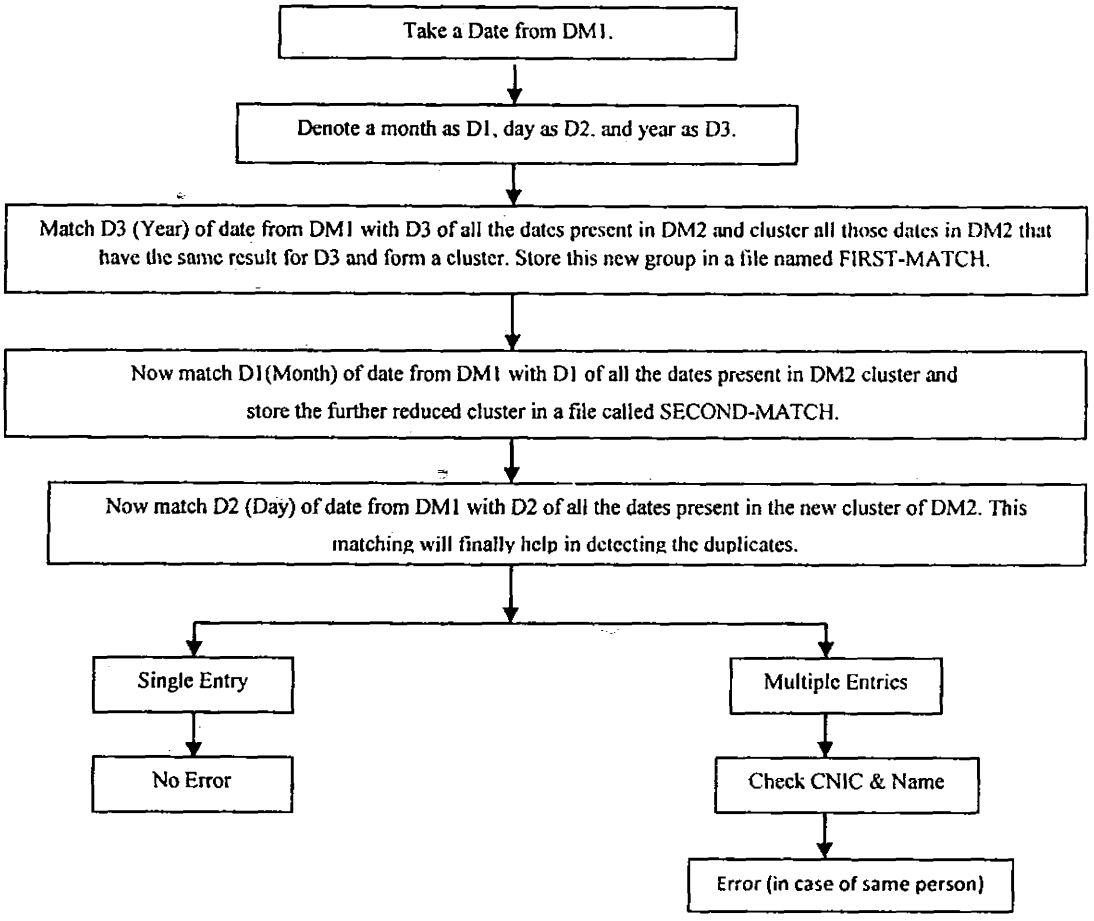


Figure 4.4 Flow Chart of Algorithm for Date Attribute.

- **Passport Number**

The Passport number is an alphanumeric value so it consists of two parts, alpha part and numeric part.

A. Preprocessing

The passport numbers are converted into numerical value by using the following relation and stored in a file for ready reference.

$$|(\text{radix})^{\text{place value}} * \textbf{Face Value}| \bmod m$$

B. Alliance Rules Application

The algorithm for detecting duplicity in passport field of a data warehouse is as follows:

1. Take a passport number from DM1.
2. A passport number consists of two parts. Denote the score of alpha part as Score(1) and the numerical part as Score(2).
3. Now match Score(2) of passport number from DM1 with Score(2) of all the passport numbers from DM2 and group the passport numbers with same values of Score(2) and store it in scoring file 1.
4. Now match Score(1) of passport number from DM1 with Score(1) of all the passport numbers from new DM2 cluster and further reduce the size of the cluster and store it in scoring file 2.

   Now there exist two cases for Score(1) matching as discussed below. This score matching will fully help in detecting the duplicates.

C. Duplicity Detection

Case 1: Perfect Match

a) **Single Entry Match**: In case of single entry there is no duplicity and hence no error is detected.

b) **Multiple Entry Match**: If multiple entries are skimmed out, dataset is infected with duplicate records.

Case 2: Partial Match

If Score(1) does not match, then check (DOB & Address). If the value matches then it is the same person. Hence error is identified as duplicity due to wrong data entry.

The complexity of the algorithm is $O(n^2)$

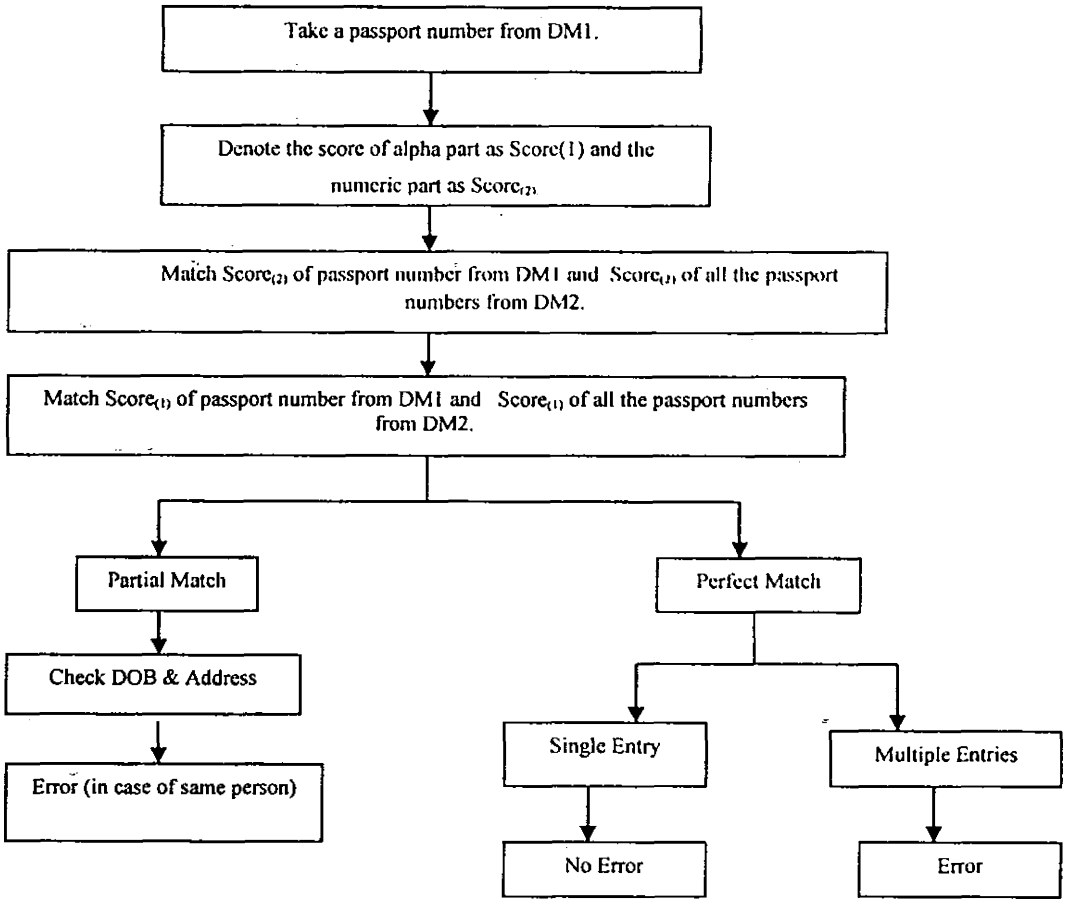


Figure 4.5Flow Chart of Algorithm for Date Attribute.

4.2.3 Numeric Attributes (CNIC, Emp_id, Phone_No)

The numeric attributes like CNIC, Emp_id and Phone _No can be directly matched. There is no need to devise an algorithm for these fields based on the concept of Alliance Rules.

Chapter 5

IMPLEMENTATION

The term implementation means the preparation or application of a plan, a technique or any strategy for undertaking something. In the field of IT , deployment means, implementing new software or hardware in its environment including installation, configuration, running and testing.

This chapter will provide the implementation details of the software being deployed. The chapter is organized as: The section 5.1 and 5.2 describe the detail of the tools and technologies used for application development and documentation; section 5.3 displays the workflow diagram of the system; section 5.4 provides the pseudo-code of the software.

5.1 Tools and Technologies used for Application Development

The following tools and technologies are used to build the application:

5.1.1 Visual Studio .Net 2010

Microsoft Visual Studio by Microsoft provides IDE used to develop GUI based applications and web sites. It supports different programming languages by means of language services. It is packed with a lot of new functionality and features for the software developers.

5.1.2 Visual Basic .Net

When object oriented programming languages are considered, Visual Basic .NET comes into mind that can be perceived as an advancement of the classic Visual Basic (VB). It provides the easiest, most creative language and tool for building Windows and Web applications. It provides improved visual designers, increased application performance and a powerful integrated environment. Fast and effective coding is provided by visual basic .NET.

5.1.3 SQL Server 2005

SQL Server 2005 is a complete; unified end to end data clarification with a safe, dependable and prolific platform for enterprise data and commercial intelligence applications. It reduces the difficulty of generating, installing, supervision and consuming enterprise data and investigative submissions on platforms ranging from mobile devices to enterprise data system. SQL Server 2005 provides enhanced features of developing programs and error resistant SQL code. It provides faster query execution capability with a complete data solution for enterprises of all sizes.

5.2 Tools Used for Documentation

Following tools are used for documentation.

5.2.1 Microsoft Word

Microsoft Word is a classy word processing software, particularly aimed for carrying out various word processing jobs, like typing, editing and printing text based information. It permits adequate control over the formation and demonstration of theoretical work. Microsoft Word provides the services for basic Graphic design, statistical report of a document, spelling and grammar check etc. It provides many other features and services that make it exceptional in performance as compared to the rest of the word processing softwares.

5.2.2 Microsoft Excel

Excel is no doubt the most important computer software program used in offices today. Microsoft Excel is a marketable spreadsheet application designed and circulated by Microsoft for Microsoft Windows operating system. It has the elementary qualities of all the spreadsheet softwares having cells at the intersection of each row and column. It can be used as a calculator, a data converter and for making spreadsheets for information interpretation. It allows the users to apply a wide variety of formulas like mathematical, statistical and accounting methods. With the help of excel, charts and graphs can be prepared. Excel can be customized to perform a variety of functions.

5.3 System Flow Diagram

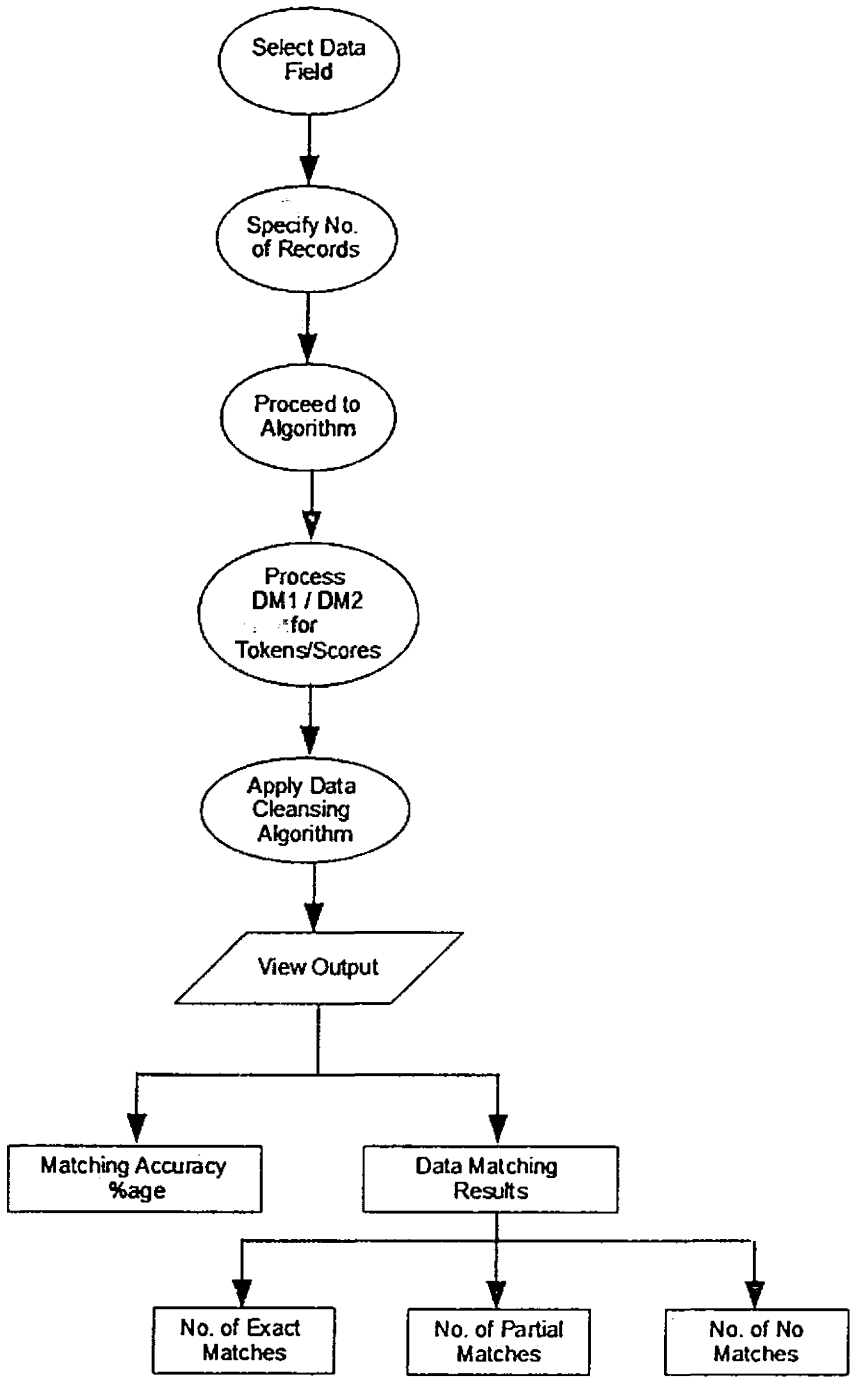


Figure 5.1: System Flow diagram

5.4 Pseudo-code for the Algorithms

The pseudo-code for different algorithms is provided below:

5.4.1 Pseudo-code for Western Names

```
Sub CleanseStringData
    Load DM1
    for Each Row in DM1 Rows
        Create Tokens of Name
        Calculate Scores for Each Token
        Create Initials for Name and Calculate Score
        Count Number of Token (NCount) of Name including Initials
        Append Tokens and Scores to Row
    Next

    Load DM2
    for Each Row in DM2 Rows
        Create Tokens of Name
        Calculate Scores for Each Token
        Create Initials for Name and Calculate Score
        Count Number of Token (N Count) of Name including Initials
        Append Tokens and Scores to Row
    Next

    For Each Row in DM2
        Match Rows from DM1 having Same N Count in DM1
        If Count of Filtered Rows > 0 Then
            Match Filtered Rows of DM1 with Same Score_n as of DM2 Row
            If Count of Filtered Rows > 0 Then
                It is Partial Match
                Match Filtered Rows of DM1 with Same Score_f as of DM2 Row
                If Count of Filtered Rows = 0 then
                    No Match
                Else
                    For Each Remaining Scores in DM2
```

```
                                        Match Filtered Rows of DM1 with Similar Score of
DM2
                                        If Count of Filtered Rows = 1 then
                                            Single Entry Match (Perfect Match)
                                            Exit For
                                        End if
                                    Next
                                End if

                            End If
                        End if
                        If Perfect Match then
                            Store PID of Matching DM1 Rows in DM2
                        End if
                        If No Match then
                            Increase No Match Count
                        Else If Partial Match then
                            Increase Partial Match Count
                        Else if Perfect Match then
                            Increase Perfect Match Count
                        End if
    Next
End Sub
```

5.4.2 Pseudo-code for Local Names/Addresses

```
Sub CleanseStringData
    Load DM1
    for Each Row in DM1 Rows
        Create Tokens of Name/Address
        Calculate Scores for Each Token
        Create Initials for Name/Address and Calculate Score
        Count Number of Token (NCount) of Name/Address including Initials
        Append Tokens and Scores to Row
```

```
Next

Load DM2
for Each Row in DM2 Rows
        Create Tokens of Name/Address
        Calculate Scores for Each Token
        Create Initials for Name/Address and Calculate Score
        Count Number of Token (NCount) of Name/Address including Initials
        Append Tokens and Scores to Row
Next

For Each Row in DM2
        Match Rows from DM1 having Same NCount in DM1
        If Count of Filtered Rows > 0 Then
                Match Filtered Rows of DM1 with Same Score2 as of DM2 Row
                If Count of Filtered Rows> 0 Then
                        It is Partial Match
                        Match Filtered Rows of DM1 with Same Score(n+1) as of DM2 Row
                        If Count of Filtered Rows = 0 then
                                No Match
                        Else
                                For Each Remaining Scores in DM2
                                        Match Filtered Rows of DM1 with Similar Score of
DM2
                                        If Count of Filtered Rows = 1 then
                                                Single Entry Match (Perfect Match)
                                                Exit For
                                        End if
                                Next
                        End if

                End If
        End if
        If Perfect Match then
```

```
                Store PID of Matching DM1 Rows in DM2
            End if
            If No Match then
                Increase No Match Count
            Else If Partial Match then
                Increase Partial Match Count
            Else if Perfect Match then
                Increase Perfect Match Count
            End if
        Next
End Sub
```

5.4.3 Pseudo-code for E-mail Address

```
Sub CleanseEmailData
    Load DM1
    for Each Row in DM1 Rows
        Create Tokens of Email (Email ID as SId_n and Email Domain SD_n)
        Calculate Score for Each Token
        Append NCount, DomainCount, Tokens and Scores to Row
    Next

    Load DM2
    for Each Row in DM2 Rows
        Create Tokens of Email (Email ID as SId_n and Email Domain SD_n)
        Calculate Score for Each Token
        Append NCount, DomainCount, Tokens and Scores to Row
    Next

    For Each Row in DM2
        Match Rows from DM1 having Same NCount and Domain Count in DM1
        If Count of Filtered Rows > 0 Then
            Match Filtered Rows of DM1 with Same SId_n as of DM2 Row
            If Count of Filtered Rows > 0 Then
```

```
                    Partial Match
                    Match Filtered Rows of DM1 with Same SD1 as of DM2 Row
                    If Count of Filtered Rows > 0 Then
                        For Each Remaining Scores in Email Domain Scores
                            Match Filtered Rows of DM1 with Similar SD of
DM2
                                If Count of Filtered Rows = 0 Then
                                No Match
                            Else
                                For Each Remaining Scores in Email ID
Scores
                                    Match Filtered Rows of DM1 with
                                    Similar SI of DM2
                                Next
                                If Count of Filtered Rows = 1 Then
                                    Perfect Match
                                End if
                            End if
                        Next
                    End if
                End if
            End if
            If Perfect Match then
                Store PID of Matching DM1 Rows in DM2
            End if
            If No Match then
                Increase No Match Count
            Else If Partial Match then
                Increase Partial Match Count
            Else if Perfect Match then
                Increase Perfect Match Count
            End if
        Next
End Sub
```

5.4.4 Pseudo-code for Date

```
Sub CleanseDateData
    Load DM1
    for Each Row in DM1 Rows
        Create Tokens of DOB (Month as Token1, Day as Token2, Year as Token3)
        Calculate Scores for Each Token
        Append NCount, Tokens and Scores to Row
    Next

    Load DM2
    for Each Row in DM2 Rows
        Create Tokens of DOB (Month as Token1, Day as Token2, Year as Token3)
        Calculate Scores for Each Token
        Append NCount, Tokens and Scores to Row
    Next

    For Each Row in DM2
        Match Rows from DM1 having Same Score_n in DM1
        If Count of Filtered Rows > 0 Then
            Match Filtered Rows of DM1 with Same Score_1 as of DM2 Row
            If Count of Filtered Rows > 0 Then
                It is Partial Match
                Match Filtered Rows of DM1 with Same Score_2 as of DM2 Row
                If Count of Filtered Rows = 0 Then
                    Partial Match
                Else
                    Perfect Match
                End if
            End If
        End if
        If Perfect Match then
            Store PID of Matching DM1 Rows in DM2
        End if
        If No Match then
```

```
                    Increase No Match Count
                Else If Partial Match then
                    Increase Partial Match Count
                Else if Perfect Match then
                    Increase Perfect Match Count
                End if
        Next
End Sub
```

5.4.5 Pseudo-code for Passport Number

```
Sub CleanseAlphaNumericData
        Load DM1
        for Each Row in DM1 Rows
            Create Tokens of Passport Number (Alpha Value as Token1, Numeric Value as
Token2)
                Calculate Scores for Each Token
                Append NCount, Tokens and Scores to Row
        Next

        Load DM2
        for Each Row in DM2 Rows
                Create Tokens of Passport Number (Alpha Value as Token1, Numeric Value as
Token2)
                Calculate Seores for Each Token
                Append NCount, Tokens and Scores to Row
        Next

        For Each Row in DM2
                Match Rows from DM1 having Same Score_n in DM1
                If Count of Filtered Rows > 0 Then
                    Match Filtered Rows of DM1 with Same Score_1 as of DM2 Row
                    If Count of Filtered Rows = 0 Then
                        Partial Match
```

```
                    If Count of Filtered Rows = 1 Then
                        Perfect Match
                    End If
                End if
                If Perfect Match then
                    Store PID of Matching DM1 Rows in DM2
                End if
                If No Match then
                    Increase No Match Count
                Else If Partial Match then
                    Increase Partial Match Count
                Else if Perfect Match then
                    Increase Perfect Match Count
                End if
        Next
End Sub
```

Chapter 6

TESTING AND PERFORMANCE EVALUATION

In this chapter the details of testing and performance evaluation of the proposed model will be provided. The significance of software testing and its influence on software cannot be underrated. Testing of a software is a basic factor for software quality assurance and signifies an evaluation of requirement, scheme and coding. The better prominence of software systems and the cost related with software disaster are encouraging aspects for development, through testing.

The Alliance Rules based on the principal of Data Mining Association Rules provide a solution for detecting errors in the datasets. The errors are detected automatically. This chapter will provide the results of the mathematical model after applying the algorithms on different data types present in a data warehouse. The attributes having maximum probability of being primary or composite key are considered.

6.1 Name Field

The application of the proposed algorithm for checking and removing duplicity in the name field is given below:

A. Preprocessing

The strings are transformed into integers using the following formula:

$$|(\text{radix})^{\text{place value}} * \text{Face Value}| \bmod m$$

Supposing the "Name Attribute" as a primary key in DM1, then consider any occurrence of name to calculate the value of the attribute by means of the above stated formula. Take "m" as a large prime number. The formula considers the values as case in-sensitive.

For example:

Name = M. Asif Javaid

Calculation of scores of all the words present in this name is given below.

Let m = 104729.

First Name= M.

$= M^{1.0}$ (Apply place values)

$= [(67)^1 * 12 + (67)^0 * 43] \bmod m$ (Apply face values)

= [(67) * 12 + (1)* 43] mod 104729 (m is any large prime number)

= [805 + 43] mod 104729

= [847] mod 104729

= [847] (The final integer value of M.)

Middle Name= ASIF

$= A^3S^2I^1 F^0$

$= [(67)^3 * 0 + (67)^2 * 18 + (67)^1 *8 + (67)^0 *5] \text{ mod } 104729$

= [(300763 * 0 + (4489) *18 + (67) * 8 + (1) *5] mod 104729

= [0 + 80802 + 536+ 5] mod 104729

= [81343] mod 104729

= 81343

Last Name= JAVAID

$= J^5 A^4 V^3A^2 I^1D^0$

$= [(67)^5 * 9 + (67)^4 * 0 + (67)^3* 21 + (67)^2 * 0+ (67)^1 * 8 + (67)^0 * 3)] \text{ mod } 104729$

= [(1350125107) *9 + (20151121) *0 + (300763) * 21 + (4489)* 0 + (67) * 8 + (1) * 3] mod 104729

= [12151125963 + 0 + 6316023 + 0 + 536 +3) mod 104729

= [12157442525] mod 104729

=81289

(Note: Similarly the integer values of all the names, addresses, E-mail addresses, dates and passport numbers is calculated and stored in a file for ready reference.)

B. Alliance Rules Application for De-duplicity

1. Take a name from DM1. **(M. Asif Javaid)**
2. Calculate the count of words in the name and represent it by N. **(N= 3)**
3. Calculate N+1 scores for the name each corresponding to a word present in name. The (N+1) th score is the score of the initials of the name. **(MAJ = 53877)**
4. In Data Mart 2 (DM2), cluster the names which have the same value of N.

DM2 Cluster 1	
NAME	**Value Of N**
Ch. Rizwan Ahmad	3
M. Arif Javaid	3
M. Amir Raza	3
M. Asif Javaid	3
M. Atif Javaid	3
M. Ahsan Raza	3
M. Asif Javaid	3
M. Kashif Javaid	3

Table 6.1: DM2 Cluster 1(Name Attribute)

5. Calculate N+1 scores for all names in this cluster of DM2.

DM2 Cluster 2		
Name	**Value of N**	**Score of N+1**
Ch. Rizwan Ahmad	3	(CRA)= 10117
M. Arif Javaid	3	(MAJ)= 53877
M. Amir Raza	3	(MAR)= 53885
M. Asif Javaid	3	(MAJ)= 53877
M. Atif Javaid	3	(MAJ)= 53877
M. Ahsan Raza	3	(MAR)= 53885
M. Asif Javaid	3	(MAJ) =53877
M. Kashif Javaid	3	(MKJ) = 54547

Table 6.2: DM2 Cluster 2(Name Attribute)

6. Match the last name scores Score(n) of the name in DM1 & Score(n) of each name in DM2 cluster and group all those names in DM2 that have same score value for Score(n) and further decrease the size of the cluster. Store it in a file called scoring file 1.

The Last name score Score(n) of the name in DM1 (M. Asif Javaid) is = 81289. The matching records in DM2 cluster are stored in a file called scoring file 1 as under.

Scoring File 1	
Name	**Score(n)**
M. Asif Javaid	81289
M. Arif Javaid	81289
M. Atif Javaid	81289
M. Asif Javaid	81289
M. Kashif Javaid	81289

Table 6.3: Scoring File 1(Name Attribute)

7. Now match Score(1) of name from DM1 & Score(1) of all the names from new DM2 cluster and store the further reduced cluster in file called scoring file2.

Score(1) for the first name from DM1 (M. Asif Javaid) is = 847. The Score(1) of the name taken from DM1 is matched with Score(1) of all the four records present in the new cluster of DM2 (Score file 2) as under.

Scoring File 2	
Name	**Score(1)**
M. Asif Javaid	847
M. Arif Javaid	847
M. Atif Javaid	847
M. Asif Javaid	847
M. Kashif Javaid	847

Table 6.4: Scoring File 2(Name Attribute)

C. Duplicity Detection

Now there exist three cases for Score(1) matching .This score matching finally helps in detecting the duplicates.

Case 1: Perfect Match

a) **Single Entry Match**: In case of single entry there is no duplicity hence no error is detected. **(Single entry is not resulted.)**

b) **Multiple Entry Match:** In this case, match other scores of the name which are Score(2), Score(3) up to Score(n-1). If a single entry is resulted out, then there is no error. But if multiple entries are skimmed out, dataset is infected with duplicate records.

(Multiple entries are skimmed out.)

(DM2)				
Name	**Score(1)**	**Score(2)**	**Score(3)**	**Score(N+1)**
M. Asif Javaid	847	81343	81289	53877
M. Asif Javaid	847	81343	81289	53877

Table 6.5: Final DM2 Cluster (Name Attribute)

Now in the case of current example case1: (b) is applicable as multiple entries are resulted so the value of Score(2) will be checked. Thus Score(2) of name from DM1 matches with two records of Score(2) of the names present in DM2 cluster.

Case 2: No Match

If Score(1) score does not match then match the scores of initials i. e $S_{(N+1)}$ now this can result in two conditions.

a) **Same person**

b) **Different person with same initials**

In this case check the values of (DOB + Address). If the value matches then it is the same person then error is identified due to duplicity of values. But if it is not the same person then no error exists.

Case 3: None of the score matches from Score(n) to Score(n+1)

a) **That entry does not exist.**

b) **Entry exists with some errors in name.**

For entry with some errors in name, apply the concept of q-grams. These are the substrings of a given string of length q. Most appropriate length is 3. The q-grams of M. Asif Javaid are given below:

(1. ##M), (2. #M.), (3.M._), (4. ._A), (5. _AS), (6. ASI), (7. SIF), (8. IF_), (9. F_J), (10. _JA), (11. JAV), (12.AVA), (13.VAI), (14.AID), (15.ID#), (16.D##)

(Space in name is represented by_).

(If modification is applied in the example specified i.e. if Score(2) of the name in DM1 is matched with Score(2) of all the names in DM2 cluster in Step. 7, instead of matching Score(1) then a reduced cluster will be resulted and the further procedure of duplicity detection will become quite efficient because in local names the first name is usually the same as M., Muhammad., Ch., and Chaudhary etc. but the second name is different so Score(2) should be matched first instead of Score(1).

6.2 Address Attribute

The application of the proposed algorithm for checking and removing duplicity in the address field is given below:

A. Preprocessing:

The strings are converted into numbers using the following relation:

$$|(\text{radix})^{\text{place value}} * \text{Face Value}| \bmod m$$

B. Alliance Rules Application

1. Take an address from DM1. **(H. #34, Lane.02, WahCantt.)**
2. Calculate the sum of words in the address and represent it by N. **(N= 3)**
3. Calculate N+1 scores for the address each corresponding to a word present in the address. The $(N+1)^{th}$ score is the score of the initials of the address. **(HLW= 32182)**
4. In Data Mart 2 (DM2), cluster the addresses which have the same value of N.

DM2 Cluster 1	
Address	**Value Of N**
H.F-450, F-6/2, Islamabad.	3
H. #34, Lane.02, WahCantt.	3
H. #34, Lane.02, WahCantt.	3
H. #34, Lane.02, WahCantt.	3
H. #34, Lane.02, WahCantt.	3
H. #34, Lane.02, WahCantt.	3

Table 6.6: DM2 Cluster 1 (Address Attribute)

5. Calculate N+1 scores for all the addresses in this cluster of DM2.

DM2 Cluster 2		
Address	**Value Of N**	**Score of N+1**
H.F-450, F-6/2, Islamabad.	3	(HFI) = 31766
HouseNo.34, Lane.02, WahCantt.	3	(HLW) = 32182
HouseNo.34, Lane.02, WahCantt.	3	(HLW) = 32182
HouseNo.34, Lane.02, WahCantt.	3	(HLW) = 32182
HouseNo.34, Lane.02, WahCantt.	3	(HLW) = 32182
HouseNo.34, Lane.02, WahCantt.	3	(HLW) = 32182

Table 6.7 DM2 Cluster 2 (Address Attribute)

6. Match the last word's score Score(n) of the address in DM1 & Score(n) of each address present in DM2 cluster and cluster all those addresses in DM2 that have same score value for Score(n) and further decrease the size of the cluster. Store it in a file called scoring file 1.

The Last word's score Score(n) of the address in DM1 (WahCantt.) is = 908.The matching records in DM2 cluster are stored in scoring file 1 as under.

Scoring File 1	
Address	**Score(n)**
HouseNo.34, Lane.02, WahCantt.	908
HouseNo.34, Lane.02, WahCantt.	908
HouseNo.34, Lane.02, WahCantt.	908
HouseNo.34, Lane.02, WahCantt.	908
HouseNo.34, Lane.02, WahCantt.	908

Table 6.8: Scoring File 1 (Address Attribute)

7. Now match Score(2) of address from DM1 & Score(2) of all the addresses from new DM2 cluster and store the further reduced cluster in file called scoring file2.

The Score(2) for the address from DM1 (Lane.02) is = 99299 so the Score(2) of the address taken from DM1 matches with Score(2) of all the five records present in the new cluster of DM2 Scoring file 2 as under.

Scoring File 2	
Address	**Score(2)**
HouseNo.34, Lane.02, WahCantt.	99299
HouseNo.34, Lane.02, WahCantt.	99299
HouseNo.34, Lane.02, WahCantt.	99299
HouseNo.34, Lane.02, WahCantt.	99299
HouseNo.34, Lane.02, WahCantt.	99299

Table 6.9: Scoring File 2 (Address Attribute)

C. Duplicity Detection

Now there exist three cases for Score(2) matching .This score matching will finally help in detecting the duplicates.

Case 1: Perfect Match

a) **Single Entry Match**: In case of single entry there is no duplicity hence no error is detected. **(Single entry is not resulted.)**

b) **Multiple Entry Match:** For this Case, match other scores of the address which are Score(1), Score(3) up to Score(n-1). If a single entry is resulted out, then there is no error. But if multiple entries are skimmed out, datasets are infected with duplicate records.

(Multiple entries are skimmed out.)

(DM2)				
Address	**Score(1)**	**Score(2)**	**Score(3)**	**Score(N+1)**
HouseNo.34, Lane.02, WahCantt.	51697	99299	908	32182
HouseNo.34, Lane.02, WahCantt.	51697	99299	908	32182
HouseNo.34, Lane.02, WahCantt.	51697	99299	908	32182
HouseNo.34, Lane.02, WahCantt.	51697	99299	908	32182
HouseNo.34, Lane.02, WahCantt.	51697	99299	908	32182

Table 6.10: DM2 Final Cluster (Address Attribute)

Now in the current example case 1: (b) is applicable as multiple entries are resulted so check the value of Score(2). Thus Score(2) of address from DM1 matches with five records of Score(2) of addresses present in DM2 cluster.

Case 2: No Match

If Score(2) does not match then match the scores of initials i. e $S_{(N+1)}$ now this can result in two conditions.

a) **Same address**

b) **Different address with same initials**

In this case check the values of (Name + CNIC). If the value matches then it is the same person then error is identified due to duplicity of values. But if it is not the same person then no error exists.

Case 3: None of the score matches from Score(n) to Score(N+1)

a) That entry does not exist.

b) Entry exists with some errors in address.

For entry with some errors in address, apply the concept of q-grams. These are the substrings of a given string of length q. Most appropriate length is 3.

6.3 E-Mail Address Attribute

The application of the proposed algorithm for checking and removing duplicity in the E-Mail Address field is given below:

A. Preprocessing

The strings are converted into numbers using the following relation:

$$|(\text{radix})^{\text{place value}} * \text{Face Value}| \bmod m$$

B. Application of Alliance Rules

1. Take an e-mail address from DM1. **(asifjavaid@yahoo.com)**
2. Find out the count of words in the e-mail address and represent it by by N. **(N=3)**
3. In Data Mart 2 (DM2), group the e-mail addresses which have same value of N.

DM2 cluster	
E-Mail Address	**Value Of N**
rizwanahmad@yahoo.com	3
ayeshakhan@yahoo.com	3
asifjavaid@yahoo.com	3
sana@gmail.com	3
atif@gmail.com	3
asifjavaid@yahoo.com	3

Table 6.11: DM2 cluster 1 (E-Mail Address)

4. Now divide the e-mail address in two parts i.e. E-mail_ID and E-Mail_Domain. The score of E-mail_ID is denoted by $Score_{(id)}$ and score of E-Mail_Domainis denoted by $Score_{(d)}$.

DM1				
E-Mail Address	**Parts of E-Mail Address**		**Notation of Scores**	
	E-Mail_ID	**E-Mail_Domain**	**$Score_{(id)}$**	**$Score_{(d)}$**
asifjavaid@yahoo.com	asifjavaid	yahoo.com	asifjavaid: 103665	yahoo: 20757 com: 9928

Table 6.12: DM1 (E-Mail Address)

5. First of all match $Score_{(id)}1$ of e-mail address from DM1 with $Score_{(id)}1$of all the e-mail addresses from DM2 cluster. Cluster the e-mail addresses in DM2 which have the same score of $Score_{(id)}1$.Store the new cluster in scoring file 1.

The $Score_{(id)}1$ of E-Mail Address from DM1(asifjavaid@yahoo.com)is 103665. Two records are resulted.

Scoring File 1	
E- Mail Address	**$Score_{(id)}1$**
asifjavaid@yahoo.com	103665
asifjavaid@yahoo.com	103665

Table 6.13: Scoring File 1(E-Mail Address)

6. Now match the $Score_{(d)}1$ of DM1 with $Score_{(d)}1$ of all the e-mails present in DM2 cluster and cluster all those e-mail addresses in DM2 cluster that have the same score value for $Score_{(d)}1$ and further reduce the size of the cluster. Store the new cluster in scoring file 2.

Scoring File 2	
E- Mail Address	**$Score_{(d)}1$**
asifjavaid@yahoo.com	20757
asifjavaid@yahoo.com	20757

Table 6.14: Scoring File 2 (E-Mail Address)

7. Then match $Score_{(d)}2$ to $Score_{(d)}n$ of e-mail address from DM1 and of all the e-mail addresses present in DM2 cluster and store the further reduced cluster in file called scoring file 3.

Scoring File 3		
E-Mail Address	**$Score_{(d)}1$**	**$Score_{(d)}2$**
asifjavaid@yahoo.com	20757	9928
asifjavaid@yahoo.com	20757	9928

Table 6.15: Scoring File 3 (E-Mail Address)

C. Duplicity Detection

Case 1: Perfect Match

a) Single Entry Match

In case of single entry, there is no duplicity, hence no error detected.

(Single Entry is not resulted.)

b) Multiple Entry Matches

In this case, match other scores which are $Score_{(id)}2$ to $Score_{(id)}n-1$. If a single entry is resulted out by matching all the scores then there is no error but if multiple entries are skimmed out, the dataset is infected with duplicate records.

(Multiple entries are present.)

6.4 Date Attribute

The application of the proposed algorithm for checking and removing duplicity in the date field is given below:

A. Pre processing

The date is converted into numbers using the following relation:

$$|(\text{radix})^{\text{place value}} * \text{Face Value}| \bmod m$$

B. Alliance Rules Application

1. Take a date from DM1. **(02/12/1978)**

2. A date consists of 3 parts, (Month, Day and Year,). Let us denote a Month as D1, Day as D2, and Year as D3. **(12(D1), 02(D2), 1978(D3)**

3. First of all match the scores of D3 (year) of date from DM1 with scores of D3 of all the dates present in DM2 and cluster all those dates in DM2 that have the same result for D3 and form a group. Store this new group in file FIRST-MATCH.

FIRST-MATCH
08/11/78
02/12/78
07/12/78
02/11/78
02/12/78

Table 6.16: DM2 cluster1

4. Now match the scores of D1 (Month) of date from DM1 with the scores of D1 of all the dates present in cluster1 of DM2 and store the further reduced group in a file SECOND-MATCH.

SECOND-MATCH
02/12/78
07/12/78
02/12/78

Table 6.17: DM2 cluster 2

5. Now match the scores of D2 (Day) of date from DM1 with the scores of D2 of all the dates present in the new cluster of DM2. This matching will finally help in detecting the duplicates.

FINAL-MATCH
02/12/78
02/12/78

Table 6.18: DM2 Final Cluster

B. Duplicity Detection

For perfect match, consider the following two cases.

Case 1: In case of single entry, duplicity is not detected so no error is found.

(Single Entry is not resulted.)

Case 2: If the result shows a single entry by matching all the values then there is no duplicity, but if multiple entries are skimmed out, then in this case check the values of (CNIC + Address). If the values match then it is the same person. Hence the error is identified as duplicity. If the values do not match then no error is found.

(Multiple entries are skimmed out.)

6.5 Passport Number

The Passport number is an alphanumeric value so it consists of two parts, alpha part and numeric part.

A. Preprocessing

The passport number is converted into numerical value by using the following relation and stored in a file for ready reference.

$$|(\text{radix})^{\text{place value}} * \text{Face Value}| \bmod m$$

B. Alliance Rules Application

The algorithm for detecting duplicity in passport field of a data warehouse is as follows:

1. Take a passport number from DM1. **(ISB2356467)**
2. A passport number consists of two parts. Denote the score of alpha part as Score(1) and the score of numeric part as Score(2).

Passport Number	
α-Part Score(1)	Numeric Part Score(2)
ISB	2356467

Table 6.19: Parts of Passport Number

3. Now match Score(2) of passport number from DM1 with Score(2) of all the passport numbers from DM2 and group the passport numbers with same values of Score(2) and store it in scoring file 1.

Passport Number (DM2 cluster)	
α- Part	Numeric Part Score(2)
ISB	91069
ISB	91069

Table 6.20: Scoring File 1 (Passport Number)

4. Now match Score(1) of passport number from DM1 and Score(1) of all the passport numbers from new DM2 cluster and further reduce the size of the cluster and store it in scoring file 2.

Passport Number (DM2 cluster)		
α- Part	α-Part Score(1)	Numeric Part Score(2)
ISB	37119	91069
ISB	37119	91069

Table 6.21: Scoring File 2 (Passport Number)

C. Duplicity Detection

Now there exist two cases for Score(1) matching as discussed below. This score matching will fully help in detecting the duplicates.

Case 1: Perfect Match

a) **Single Entry Match**: In case of single entry there is no duplicity and hence no error is detected. **(Single entry is not resulted.)**

b) **Multiple Entry Match**: If multiple entries are skimmed out, dataset is infected with duplicate records. **(Multiple entries are resulted.)**

Case 2: Partial Match

If Score(1) does not match, then check the value of (DOB & Address). If the value matches then it is the same person. Hence error is identified as duplicity due to wrong data entry.

Chapter 7

CONCLUSION AND FUTURE RECOMMENDATIONS

This chapter will provide the concluding remarks, achievements and future recommendations. The Alliance Rules based on the principal of Data Mining Association Rules provide a way out for detecting errors in the datasets. The errors are detected automatically. The high degree of computerization can be achieved only by decreasing the manual interference.

7.1 Achievements

- All through the current research a useful deep scrutiny of prevailing cleansing methods was brought about. Evaluation table is created that provides an overview of the current traditional data cleansing approaches.
- In the next stage, based on the evaluation table, the requirement analysis is carried out. This analysis provided information about the requirements which are fulfilled and which are pending.
- The mathematical model and implementation of different algorithms based on Alliance Rules is provided to detect the errors in different fields of a data warehouse. These algorithms are fully supported for satisfying the requirements of data cleansing as likened to the prevailing approaches or methods for data cleansing process.
- The main aim of developing these algorithms is to attain high precision by taking into account all the characteristics of defective data.
- The proposed algorithms cover different data types present in a data warehouse, which is lacking in the previous research work.
- For detecting and removing the faulty data, different automated and generalized algorithms are presented. The process of de-duplication is applied on different attributes of the data warehouse for cleansing.

7.2 Future Recommendations and Improvements

- The future work can be scoped out to implement all these algorithms on different working data marts.
- The implementation of q-grams with maximum precision is the next step of this research.

REFERENCE AND BIBLIOGRAPHY

Reference and Bibliography

[1] Rajiv Arora, PayalPahwa, Shubha Bansal, "Alliance Rules for Data Warehouse Cleansing", International Conference on Signal Processing Systems, pp. 743-747, Department of IT GPMCE, Delhi, India, 16 July, 2009.

[2] G.N. Wikramanayake, J.S. Goonetillake, "Managing Very Large Databases and Data Warehousing", Sri Lankan Journal of Librarianship and Information Management, vol. 2, no.1, pp 22-29, University of Colombo School of Computing, Colombo, Sri Lanka, 2009.

[3] Mariam Rehman, Vatcharapon, Esichaikul. "Duplicate Record Detection for Database Cleansing", Second International Conference on Machine Vision, IEEE Computer Society, PathumThani, Thailand, 2009.

[4] J. Jebamalar Tamilselvi, Dr. V. Saravannan, "A Unified Framework and Sequential Data Cleansing Approach for a Data Warehouse", IJCSNS International Journal of Computer Science and Network Security, vol.8 no.5, pp. 117-121,Ideas Group ***Publishers,***Tamilnadu, India, 20 May, 2008.

[5] Joseph M. Hellerstein, "Quantitative Data Cleaning for Large Databases", ACM SIGKDD, International Conference on Knowledge Discovery and Data Mining, vol.1, no.1, pp 1-42, United Nations Economic Commission for Europe (UNECE), 27 February, 2008.

[6] Ahmed K. Elmagarmid, Panagiotis G. Vassilios S. Verykios, "Duplicate Record Detection: A Survey." In proceedings of IEEE Transactions on Knowledge and Data Engineering, vol.19, no.1, pp.1 – 16, IEEE educational Activities Department, New York, USA, January, 2007.

[7] Peter Christen. Karl Goiser, "Quality and Complexity Measures for Data Linkage and Deduplication",Conference on Knowledge Discovery and Data Mining(KDD), Springer Studies in Computational Intelligence, vol. 43, pp.127 – 151, Heidelberg, Springer Link, 2007.

[8] Wing Ning Li, Johnson Zhang, Roopa Bheemavaram, "Efficient Algorithms for Grouping Data to Improve Data Quality", International Conference on Data Mining, vol.26, no.12, 23 May, 2006.

[9] Johan Karlsteen, "Automation of Metadata Updates in a Time Critical Environment", Proceedings of the 27th International Conference on Very Large Databases, Master's Thesis in Computer Science, Umea University, Department of Computing Science SE-901-87, Umea, Sweden, August, 5, 2006.

[10] Heiko Muller, Johann Christoph Freytag, "Problems, Methods and Challenges in Comprehensive Data Cleansing", Proceedings of the 10th International ***Conference*** on Database and Expert systems, ***Journal*** of Molecular Biology, vol. 147, pp. 195-197. Humboldt University, Berlin, 10099 Berlin, Germany, 2003.

[11] Timothy E. Ohanekwu, C.I. Ezeife, "A Token-Based Data Cleaning Technique for Data Warehouse Systems", IEEE Workshop on Data Quality in Cooperative Information Systems, Siena, Italy, January, 2003.

[12] Jonathon I. Maletic, Andrian Marcus, "Data Cleansing Beyond Integrity Analysis", In Proceedings of the Conference on Information Quality, pp.200-209, Boston Press, Toronto, Boston, October 2000.

[13] Jonathon I. Maletic, Andrian Marcus, "Automated Identification of Errors in Datasets", TR-CS-00-02, University of Memphis, 2000.

[14] Andrian Marcus, Jonathon I. Maletic, "Utilizing Association Rules for the Identification of Errors in Data", The University of Memphis, Division of Computer Science, Memphis, Technical Report TR-14-2000, June 23, 2000.

[15] Erhard Rahm, Hong Hai Do, "Data Cleansing: Problems and Current Approaches", Bulletin of the Technical Committee on Data Engineering, vol.23, no.4, pp. 03-13, IEEE Computer society, University of Leipzig, Germany, December 2000.

[16] Alvaro E. Monge, "Matching Algorithms within a Duplicate Detection System", Bulletin of the Technical Committee on Data Engineering, vol.23, no.4, pp. 14-20, IEEE Computer society, Chiba, Japan, December 2000.

[17] PanosVassiliadis, Zografoula, SpirosSkiadopoulos, Nikos , TimosSellis, "ARKTOS: A Tool for Data Cleansing and Transformation in Data Warehouse Environments", Bulletin of the Technical Committee on Data Engineering, vol.23, no.4, pp. 42-47, IEEE Computer society, National Technical University of Athens, Greece, December 2000.

[18] Hernandez, S. Stolfo, "Real World Data is Dirty: Data cleansing and the Merge/ Purge Problem", Journal of Data Mining and Knowledge Discovery, vol.2, no. 1, pp. 9-37, Kluwer Academic Publishers, Boston, Netherland, January 1998.

[19] Thomas C. Redman, "The Impact of Poor Data Quality on the Typical Enterprise", Communications of the ACM, vol. 41, no. 2, pp. 79-82, February, 1998.

[20] Rakesh Agrawal, Tomasz Imielinski, Arun Swami, "Mining Association Rules Between Sets of Items in Large Databases", In Proceedings of ACM SIGMOD, International Conference on Management of Data, pp 207-216, Washington DC, USA, May 1993.

Web Links:

[21] http://en.wikipedia.org/wiki/Data_Cleansing

[22]http://www.tdan.com/view_articles/4881

ACRONYMS

Acronyms

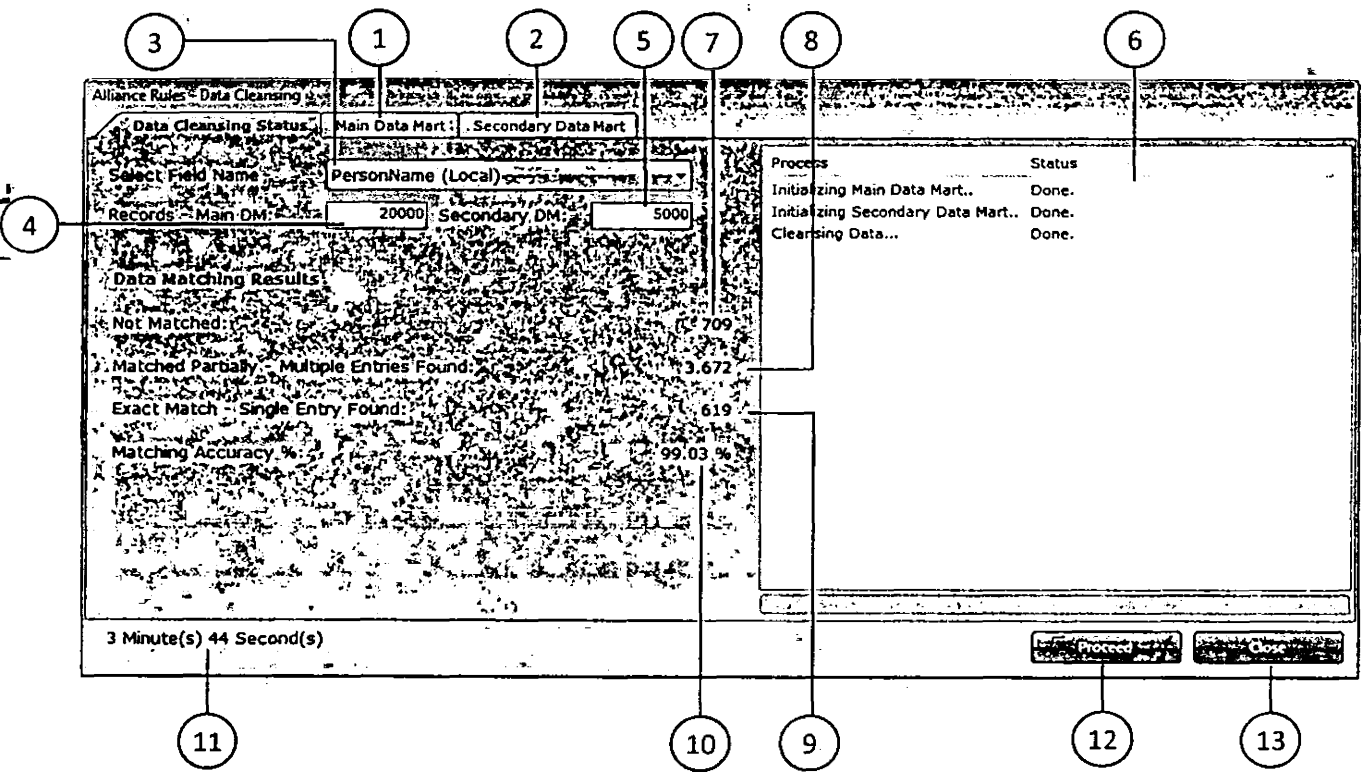
API's	Application Programming Interface
CRM	Customer Relationship Management
DBMS	Database Management System
DM	Data Mart
DSA	Data Staging Area
DW	Data Warehouse
ERP	Enterprise Resource Planning
ETL	Extract, Transform and Load
HMM	Hidden Markov Model
HTML	Hyper Text Markup Language
KDD	Knowledge Data Discovery
RDBMS	Relational Database Management System
SQL	Structured Query Language

APPENDIX

APPENDIX A

Appendix A – User Manual

Snapshot of Data Cleansing Algorithm Implementation Screen



1. Tab showing the data of the main data mart. Main data mart contains the data mart fields and fields relating to the cleansing algorithm.
2. Tab showing the data of the secondary data mart. Secondary data mart contains the data mart fields and fields relating to the cleansing algorithm.
3. Dropdown list to select the field on which the data cleansing algorithm will be applied.
4. Number of records which will be processed from main data mart during the processing of cleansing algorithm. This option helps to run the algorithm on the desired number of the rows to avoid long processing time.
5. Number of records which will be processed from secondary data mart during the processing of cleansing algorithm.
6. List showing the activities and their status during the processing of the algorithm.

7. Count of rows of the secondary which did not match with any of the main data mart record.
8. Count of rows of the secondary having partial match with any of the main data mart record.
9. Count of rows of the secondary which match perfectly with any of the main data mart record.
10. Percentage of Precision of the algorithm, calculated only for those records where perfect match was found.
11. Total time taken for the execution of the data cleansing algorithm, depending upon the selection of field and number of records.
12. After choosing the field on which the data cleansing algorithm implementation is desired and the specifying the number of record of main and secondary data marts, when the Proceed button is clicked, the system will start the processing of the algorithm.
13. Clicking Close button will close this screen.

APPENDIX B

Appendix B – Data Marts

Data Mart 1 (DM1)

E_ID	E_Name	DOB	CNIC	Address	E-Mail Address	Passport_No	Phone_No
148	M. Asif Javaid	02/12/78	3740615060790	H.#34, Lane.02, WahCantt.	asifjavaid@yahoo.com	ISB2356467	3456789
155	Nabeel Mehmood	03/13/80	3986548764903	23 A, New Chorangi, Malir, Lahore.	Nabeel_123@iiui.edu.pk	FbA1452976	72892541
180	Ch. Wajid Ali	02/11/76	9834527895609	MohallahGulshan Abad, QadarPurRaan.	wajid_ali@hotmail.com	LHA3456782	97645398
130	Naseem Awan	01/08/01	2654893767889	House No. 51B, Gulberg, Lahore.	Naseemawan@gmail.com	MLN5347862	34567887
147	Ch. Waheed Rehman	03/30/65	9843256789134	House No. 119, GaliQasim Ali, ShaheedPura, Multan.	Waheed_ch@hotmail.com	MLA4356883	94376568
178	M. Rizwan Malik	07/23/75	3740678456987	H.no.23, St.12, F-6/2, Islamabad.	rizwanmalik@yahoo.com	ISB5643789	9325648
147	Sana Rehman	03/14/74	3756489308764	H. No.34, St.15, G-10/2, Islamabad.	sana_rehman@yahoo.com	KHI7654387	9203240
931	Ch. Nabeel Mehmood	08/12/80	4182210099093	24 A, Model Town Rawalpindi.	nabeel_ch@iiui.edu.pk	ISB8976987	9207865
220	Ayesha Khan	05/30/77	4228473486193	House. No. F-230, St.10, F-7/2, Islamabad.	ayeshakhan@yahoo.com	LHA9870664	9208976
783	M. Amir Raza	04/19/67	4251605179743	House. No. 33, St. 11, Model Town, Rawalpindi.	amir_raza@gmail.com	MRD8769087	4536785

Data Mart 2 (DM2)

E_ID	E_Name	DOB	CNIC	Address	E_mail	Passport_No	Phone_No
123	Ch. Rizwan Ahmad	08/11/78	984536782998	House#.F-230, St.10, F-7/2, Islamabad.	rizwanahmad@yahoo.com	LHA546738	6543789
220	Ayesha Khan	05/30/77	4228473486193	H.F-450, F-6/2, Islamabad.	ayeshakhan@yahoo.com	LHT9870664	9208976
112	M. Arif Javaid	06/02/74	3740615437896	HouseNo.34, Lane.02, WahCantt.	arif.javaid@iiui.edu.pk	ISB6754908	3456789
149	M.Amir Raza	02/12/67	3740615060790	H.# 321, St.10, F-10/2, Islamabad.	amir.raza@yahoo.com	KHA23564	21268574
148	M. Asif Javaid	02/12/78	3740615060790	HouseNo.34, Lane.02, WahCantt.	asifjavaid@yahoo.com	ISB2356467	3456789
142	Sana Rehman	07/12/78	9876654789546	H.No.23,St 45, G-10/2, Islamabad.	sana@gmail.com	FBA7654890	7865438
256	M. Atif Javaid	23/04/73	3740689678564	HouseNo.34, Lane.02, WahCantt.	atif@gmail.com	ISB9087654	3456789
145	M. Ahsan Raza	02/11/78	3740615067690	H.# 321, St.10, F-10/2, Islamabad.	ahsan_raza@gmail.com	KHA235649	21268574
148	M. Asif Javaid	02/12/78	3740615060790	HouseNo.34, Lane.02, WahCantt.	asifjavaid@yahoo.com	ISB2356467	3456789
234	M. Kashif Javaid	07/23/86	3740634568907	HouseNo.34, Lane.02, WahCantt.	kashif_javaid@gmail.com	ISB6549073	3456789