CHAPTER 1

INTRODUCTION

Data cleaning is an emerging domain among the Knowledge Discovery in Databases process (Davison 2003, Fitzgibbon & Zisserman 1998, Ishiguro et al 2004, Jogan & Leonardis 2003) which attempts to improve the data quality by identifying and removing data artifacts (Levenshtein 1966, Lourakis & Argyros 2004, Micusik & Pajdla 2006, Mollineda et al 2002, Jeffrey & Gong 2002, Pollefeys et al 2004). The errors, discrepancies, redundancies, ambiguities, and incompleteness that are present in these data artifacts hinder the effectiveness of analysis or data mining (Lourakis & Argyros 2004, Micusik & Pajdla 2006). The automated process namely, data cleaning also scrubbing that analyzes the data, identifies missing and erroneous values and corrects them (Mollineda et al 2002). Its objective is to improve the overall data consistency (Jeffrey & Gong 2002, Pollefeys et al 2004, Schimke et al 2004), by concentrating on removing variations in data contents and decreasing data redundancy. Data cleaning involves decomposing and re-assembling and occasionally, “semantic enrichment,” e.g., gathering further information from external source(s) to settle the conflicts (Schimke & Vielhauer 2007, Li Yujian & Liu Bo 2007, Rousseeuw 1984). Finger Printing and Edit Distance is a technique Based on data cleansing process is used to find dirty records. After that, it cleans the dirty records using the following steps (i) Reducing them to get a unified whole free of omitted parts, (ii) uniting them with a single individual identity and (iii) Keeping only one copy of records that are absolute duplicates (Kukich 1992).
1.1 MOTIVATION

Knowledge discovery in database uses the methods and techniques that are derived from the areas of statistical and data analysis, decision support and machine learning. It differs from the traditional analytical methods since it overcomes some of the obstacles posed by large datasets. In traditional data analysis knowledge extraction is performed by multiple analysts who are very familiar with a given dataset. Through the use of statistical techniques these analysts manually probe the data searching for useful and interesting facts. However, as the dataset increases in size and dimensional complexity the amount of effort required by these traditional methods exceeds human capabilities. It should be noted that KDD is not a replacement for traditional methods, but is an augmentation to them (Fitzgibbon & Zisserman 1998). The KDD is an iterative process. Once the discovered knowledge is presented to the user, the evaluation measures can be enhanced, the mining can be further refined, new data can be selected or further transformed, or new data sources can be integrated, in order to get different, more appropriate results. Data cleaning also known as data cleansing, it is a phase in which noise data and irrelevant data are removed from the collection. For instance, data cleaning and data integration can be performed together as a pre-processing phase to generate a data warehouse (Angell et al 1983).

Data cleaning is an emerging domain that aims at improving data quality. It is a very large field that encompasses a number of research areas within database. Data cleaning, also called data cleansing or scrubbing, deals with detecting and removing errors and inconsistencies from data in order to improve the quality of data. Data quality problems are present in single data collections, such as files and databases, e.g., due to misspellings during data entry, missing information or other invalid data. When multiple data sources
need to be integrated, e.g., in data warehouses, federated database systems or global web-based information systems, the need for data cleaning increases significantly (Levenshtein 1966, Bledsoe 1966). The Duplicate Detection component extracts the duplicate candidates from user specified entities. Candidate entities are compared to identify duplicate clusters. The result of this process is a list of detected duplicate clusters that the user may choose to eliminate from the output XML document (Micsik & Pajdla 2006). Data cleaning removes duplicates by determining whether two or more records represented differently refer to the same real world entity, and then, it cleans the dirty records (Kukich 1992). Data cleaning determines first whether two or more records represented differently are referring to the same real-world entity, and secondly it performs any one (or combination) of the following actions (if the records represent the same object): (1) combining them to get a consolidated complete record, (2) unifying them with a single entity identifier, and (3) retaining only one copy of them (Rousseeuw 1984).

Duplicate elimination is hard because it is caused by several types of errors like typographical errors, and equivalence errors different (non-unique and nonstandard) representations of the same logical value (Sagues et al 2006). Eliminating fuzzy duplicates is applicable in any database but is critical in data-integration and analytical-processing domains, which involve data warehouses, data mining applications, and decision support systems (Levenshtein 1996). Fingerprint and Levenshtein algorithms uses irreducible polynomial to “fingerprint” files to that any unsanctioned change will be detected with a very high probability. Furthermore, updating the fingerprint when the file is locally modified in an unsanctioned change, is very simple (Kukich 1992). Fingerprints are short tags for larger objects. They have the property that if two fingerprints are different then the corresponding objects are certainly different and there is only a small probability that two different objects have the same fingerprint (Savary 2000). Near duplicate data bear
high similarity to each other, yet they are not bitwise identical. There are many causes for the existence of near duplicate data: typographical errors, versioned, mirrored, or plagiarized documents, multiple representations of the same physical object, spam, emails generated from the same template, etc (Mitton 2009).

1.2 THESIS OUTLINE

Chapter 1 Introduces the significance and intention to do the proposed work on data cleaning.

Chapter 2 discussed the literature review on the area which includes Fingerprint functions, learning user’s interest by building data cleaning method to eliminate the duplicate data.

Chapter 3 elaborates the process of knowledge discovery in databases and the related issues on redundant and illegitimate data occurrences.

Chapter 4 discussed the background work of Finger print and Edit Distance for duplicate data elimination.

In Chapter 5 focuses methodologies used on detecting duplicate data and eliminating and then identify the near-duplicate records presented in the database by utilizing threshold value. Similarly, the misspelling errors are identified and the correction can be done based on the frequency value. The records that have the lowest frequency are considered as misspelling and are corrected with the index value having the highest frequency. Finally, Illegal value errors have been recognized and corrected using Rule Based methods.

Chapter 6 describes the analysis of the results produced by the methods used to de-duplicate and remove the illegitimate values.
Chapter 7 concludes the work with the help of results obtained.

1.3 MAIN CONTRIBUTIONS

The main contributions of this thesis can be summarized as follows:

- The FPED (Finger Printing and Edit Distance) technique is proposed and also successfully implemented and tested with the data after the detailed study has been made. The results show that it provides accurate data records by eliminating the errors such as, duplicate records, near duplicate records, misspelling errors and illegal value errors.

- Data cleansing is the process of assuring correctness of data. This is performed by identifying erroneous data and eliminates them. Rabin's fingerprinting is based on the prototype of the class. It is fast and easy to implement, allows compounding, and comes with a mathematically precise analysis of the probability of collision. Namely, the probability of two strings \( r \) and \( s \) yielding the same \( w \)-bit fingerprint does not exceed max \( (|r|, |s|)/2^w-1 \), where \( |r| \) denotes the length of \( r \) in bits. The algorithm requires the previous choice of a \( w \)-bit internal "key", and this guarantee holds as long as the strings \( r \) and \( s \) are chosen without knowledge of the key.

- The Levenshtein distance is a string metric for measuring the difference between two sequences. Informally, the Levenshtein distance between two words is the minimum number of single-character edits (insertion, deletion, substitution) required to change one word into the other. The phrase edit distance is often used to refer specifically to Levenshtein distance.

- Rule-based methods, rule discovery or rule extraction from data, are data mining techniques aimed at understanding data structures, providing comprehensible description instead of only black-box prediction. Rule
based systems should expose in a comprehensible way knowledge hidden in data, providing logical justification for drawing conclusions, showing possible inconsistencies and avoiding unpredictable conclusions that black box predictors may generate in untypical situations. Sets of rules are useful if rules are not too numerous, comprehensible, and have sufficiently high accuracy. Rules are used to support decision making in classification (Classification, Machine Learning), regression (Regression, Statistics) and association tasks. Various forms of rules that allow expression of different types of knowledge are used: classical prepositional logic (C-rules), association rules (A-rules), fuzzy logic (F-rules), M-of-N or threshold rules (T-rules), similarity or prototype-based rules (P-rules). Algorithms for extraction of rules from data have been advanced in Statistics, Machine Learning, Computational Intelligence and Artificial Intelligence field data sets. This particular dataset does not contain more information related to recurrence/non recurrence question. However, it may contain interesting correlations between features, correlations that characterize a cluster of interrelated attribute values.

A typical rule-based system has four basic components

i) A list of rules or rule base, which is a specific type of knowledge base.

ii) An inference engine or semantic reasoner, which infers information or takes action based on the interaction of input and the rule base. The interpreter executes a production system program by performing the following match-resolve-act cycle.
- **Match**: In this first phase, the left-hand sides of all productions are matched against the contents of working memory. As a result a conflict set is obtained, which consists of instantiations of all satisfied productions. An instantiation of a production is an ordered list of working memory elements that satisfies the left-hand side of the production.

- **Conflict-Resolution**: In this second phase, one of the production instantiations in the conflict set is chosen for execution. If no productions are satisfied, the interpreter halts.

- **Act**: In this third phase, the actions of the production selected in the conflict-resolution phase are executed. These actions may change the contents of working memory. At the end of this phase, execution returns to the first phase.
  
  iii) Temporary working memory.
  iv) A **user interface** or other connection to the outside world through which input and output signals are received and sent.

The method induces solutions from samples in the form of ordered Disjunctive Normal Form (DNF) decision rules. A central objective of the method and representation is the induction of compact, easily interpretable solutions. This rule-based decision model can be extended to search efficiently for similar cases prior to approximating function values. Experimental results on real-world data demonstrate that the new techniques are competitive with existing machine learning and statistical methods and can sometimes yield superior regression performance.