# Data Profiling to Reveal Meaningful Structures for Standardization

Nyero Walter

Master's Thesis Completed as Part of the Requirements
for the Degree of Master of Science in Informatics, Department of
Informatics – Faculty of Mathematics and Natural Sciences
University of Bergen - Norway

November, 2009.

# Foreword

I would wish to express my gratitude to the Almighty Lord for enabling me to complete my master programs at this department. Several people have contributed in many different forms towards this research and my study. First, I would wish to thank my supervisors Professor Marc Bezem and Hans Tau Hatlestad (IntelCom AS) for their inspiration, commitment, accessibility, good cooperation, and diligent critics accorded to me in the course of this research.

I would also wish to thank Professor Bjørnar Tessem for his valuable guidance and approaches to research design and implementation.

I extend my appreciation to Ana Veronica Cordova and Brit Olaug Kalsnes (centre for information and Student Services), Leikny Lie Lavik (SIB), Ida Holen (Department of Informatics), and Professor Khalid Azim Mughal, who have contributed greatly in supporting my stay and study at this University.

Thank you also to all my friends, especially Oryem Samuel, Okello Edgar, Tom Nande, Onyango Dennis, Ayaa Leah, Sanya Patrick, Buga George, Syayo Mike, Angunda Richard, Orach Fisher, Otim Geoffery, Ibtihaj Abdelbagi, Bruno Machado, and Andrew Tiwangye for his time and constructive input in this research.

Lastly, I wish to thank my family for their invaluable support, encouragement, and love. God Bless you Mum - Ana Okello and Dad - Capt Joshua Wandeba.

Walter Nyero, Friday 20th November, 2009.

# Table of Contents

# List of Tables

# List of Figures

# List of Abbreviations

| | |
|---|---|
| ETL | Extract, Transform and Load |
| FD | Functional Dependency |
| RFD | Relative Functional Dependency |
| SME | Subject Matter Expert |
| UNC | Unified Coarse |
| UNF | Unified Fine |
| NLP | Natural Language Processing |
| 1NF | First Normal Form |
| 2NF | Second Normal Form |
| 3NF | Third Normal Form |
| AI | Artificial Intelligence |
| GRFD | Group Relative Functional Dependency |
| KDD | Knowledge Discovery in Databases |
| NP | Noun Phrase |
| ODBC | Open Database Connectivity |
| DBMS | Database Management System |
| POS | Part-of-Speech |
| IE | Information Extraction |

# Abstract

Today many organisations and enterprises are using data from several sources either for strategic decision making or other business goals such as data integration. Data quality problems are always a hindrance to effective and efficient utilization of such data. Tools have been built to clean and standardize data, however, there is a need to pre-process this data by applying techniques and processes from statistical semantics, NLP, and lexical analysis. Data profiling employed these techniques to discover, reveal commonalties and differences in the inherent data structures, present ideas for creation of unified data model, and provide metrics for data standardization and verification. The IBM WebSphere tool was used to pre-process dataset/records by design and implementation of rule sets which were developed in QualityStage and tasks which were created in DataStage. Data profiling process generated set of statistics (frequencies), token/phrase relationships (RFDs, GRFDs), and other findings in the dataset that provided an overall view of the data source's inherent properties and structures. The examination of data ( identifying violations of the normal forms and other data commonalities) from a dataset and collecting the desired information provided useful statistics for data standardization and verification by enable disambiguation and classification  of data.

# Chapter 1

This chapter discusses the overview of the research by highlighting the following aspects: background, aims, objectives, significance/justification, and scope.

## 1.0 Introduction

## 1.1 The Layout/Structure of the Thesis

Chapter 2 deals with the review of literature on this topic (data profiling) and it is divided into background and related work. Chapter 3 describes the research methodology and an overview of the IBM WebSphere tool as the technology used in this thesis' work.

Chapter 4 is devoted entirely to the discussion of the basic metrics of data profiling. Discussion of the results and Evaluation of the process is presented in Chapter 5. Chapter 6 concludes the thesis with a summary, the research outcome, a recommendation, and the way forward in terms of future work.

## 1.2 Background

### 1.2.1 The Amount of Data Quality Problems

Management and storage of data are common problems to many organizations, businesses, and institutions in the 21st century. These data and information are quite valuable for strategic decisions, customer care management, and other uses to the various entities holding them.

Globalization, businesses merger, the increased speed and flow of data interchange, data distribution; with the Internet as one of the data sources and channels have doubled data quality problems and increased the need for data standardization.

Several data from a single domain or different domains may have to be integrated and their quality is of utmost importance to the entities utilizing them.

These quality problems present the need to have reliable data sources, storage medium, and standardized data so as to meet the enterprise business goals. Thus the question: - “How can we organize data, transform it, and easily extract meaningful structures from a given dataset or corpus?” The meaningful structures extracted from the dataset are input data and information for the ETL developers performing data standardization and verification.

### 1.2.2 The Effect of Poor Data Quality

The effect of poor data quality in organization and enterprises is hard to measure. Several studies have estimated such effect on the performance and operation of these organization and enterprises. In terms of revenue costs, Eckerson (2002)[12] and Redman (1998)[34] estimated that data quality problems cost U.S. businesses more than $600 billion annually, and Redman (1996)[33] also estimated that an industrial data quality error rate of 1-5% can constitute a 10% revenue loss.

A study in an enterprise by Wang et al. (2000)[40] found that 70% of all orders had errors. Data quality problems are not only limited to revenue losses but also on human life as shown by the Institute of Medicine (2000)[17].

It should also be noted that data reliability is quite important to business leaders as compared to the other data quality problems. The datasets can contain errors but when the underlying structures are complicated then their analysis may not reveal meaningful information. It is therefore necessary that data should be put into its normal forms with a view to fulfilling some of its properties like referential integrity.

When the reliability of the data is attained, its deviation from ontologically correct representation in the individual fields (name) and records (consistent set of attributes) are considered in solving the quality problems.

It is quite important to discriminate information from noise; detecting those data that are useful or interesting owing to the reality that enterprises, governments, and individuals are turning to the Web and electronic communication for disseminating and accessing information. These require close attention to data quality particularly by addressing the following problems:

i. How to come up with basic rules for organizing data into a relational database;

ii. How to eliminate duplicative elements or values from the same table from a relational database;

iii. How to split natural elements into different fields without prior knowledge about its contents;

iv. How to attain a single data view/representation;

v. How to understand the data structures.

Items i-iii implicitly define a First Normal Form (1NF) in a relational database and items iv and v are more concerned with data records or flat files.

These data quality problems present enormous challenges for the study and understanding of the dataset or corpus with a view to analysing, extracting, and discovering knowledge and the underlying data structures from the domain.

Data profiling as one of the solution to data quality problems, should be used to generate significant understanding of the corpus and provide statistical evidence for the translation of records into a relational database so as to improve the following aspects of the system:

i. Metadata-wise: improving the definition by finding more accurate terms and definitions.

ii. Data-wise: Having precise, consistent, complete, and accurate data because of data standardization and enrichment.

iii. Structure-wise: Finding an enterprise wise model which reflects the real collection of data and relationship to improve its definition.

## 1.2.3 Data Profiling

Investigating and evaluating hypotheses and claims about human languages, similarities (with computer languages), and human interactions (knowledge representation) with computers involve the use of Natural Language Processing (NLP) technology and methods. NLP technologies and methods are helpful in data profiling, data standardization, and understanding the violation and verification of the normal forms (1NF, 2NF, 3NF, etc.).

NLP has a historical relationship with Artificial Intelligence (AI): the study of cognitive function by computational processes, with an emphasis on the role of knowledge representation, and also machine learning: the design and development of algorithms to allow computer to learn based on some set data.

These two computer science disciplines are used for the formulation and generation of some needed facts, statistics, measures, etc., which are required in solving the data quality problems extracted by NLP technology and methods such as data profiling.

### 1.2.3.1 Definitions of Data Profiling

Data profiling can be defined and explained in different ways. The following are some of the definitions and explanations:

i. Coming up with commonalities between individual records in unstructured data. These could be: the record patterns, their relationships, and frequencies of occurrences. In some instances, data may be stored in well structured data model and the commonalities are not implicitly documented hence getting data profiles are useful in finding data structures.

ii. Determining relational database imperfection such as 1NF, 2NF, 3NF violation and non-conformance with the ontology (lack of standardization). Profiling tools go a long way in revealing these violations, though most current tools do not adequately cover 1NF violation.

iii. Revealing differences when combining well-structured databases and how to create a unified model which reflects all data sources in the database.

iv. Collection of statistics that can reveal information about the data source or part of it to help in data integration and data cleansing.

Data profiling is generally defined as the process of revealing structures, patterns in the contents of data and any other information helpful for Extract, Transform, and Load (ETL) developer(s) to make the right modelling decisions and precautions in processing the data so that the results can be reliable.

### 1.2.3.2 Micro Level Profiling

The development of profiling technologies should be seen against the background of data quality problems. These technologies are thought to efficiently collect and analyse data so as to find or test knowledge in the form of statistical patterns between data. The current profiling tools have system that can perform the following tasks:

i. Column analysis:

- to reveal data types of text fields: date, integer, real, etc.
- to reveal distribution on distinct values.
- to generate frequencies of different values or tokens.

ii. Table analysis to reveal relative functional dependency and other relationships,

iii. Analysis on different tables (cross table analysis) to reveal further relationship between tables,

iv. Suggesting a data model which covers the union of all data sources.

The above tasks show that the current profiling tools are able to handle significant amount of data quality problems at the macro level however, they show many gaps in complying with 1NF and handling of data contents in records.

Examples can be on addresses that may be stored in a dataset and they have to be put in a standard form where all fields are clearly identified and duplication removed (Agichtein 2003)[11].

Consider an arbitrary field of some database record called "ITEM", the contents "HEXAGON SCREW FOR PLATE MOUNTING" is not in 1NF because it contains both the item name (HEXAGON SCREW) and its purpose (PLATE MOUNTING) which are two different kinds of information. But this is not easy to see at first. The field needs to be split, but how do we make the rule to be able to perform the field splitting?

To be able to split the fields in the records, and extract meaningful structures and information, profilers should not only limit data profiling at the macro level (examining the data, and collecting statistics and information across different table) but also perform data profiling at a micro level (examining the data, and collecting statistics and information in greater depth within a field).

The focus of this research was therefore on data profiling at a micro level where statistical evidences and the dependencies relationships are analysed into details.

### 1.2.3.3 Basic Metrics from Data Profiling

Most of the data profiling tools support the tasks mentioned in Section 1.2.3.2, but may not be able to support  tasks like putting contents into a relational database in 1NF. Data profiling at micro level therefore aids in revealing violation of the normal forms and non-conformance with an ontology; facts and metrics that are used later in data standardization and verification.

The basic metrics were: phrase frequency, relative functional dependency, and group relative functional dependency.

Statistical confidence level estimates were used in this research to show that a chosen token or group of tokens have some meaning or significance in the dataset. Tokens/phrase relationships and other terminologies such as membership and group confidence were introduced so as to provide more metrics for the ETL developers.

The tool for the project was the IBM WebSphere which is divided into QualityStage (used for creating rule sets ) and DataStage (used for creating jobs/tasks).

## 1.3 Justification/Significance

### 1.3.1 Data Quality Challenges

When there is a need for integrating several data sources into one system, for example, data warehouses, database systems, or web-based information systems, the need for data cleansing increases considerably. The considerable increase is due to the fact that data sources often contain many data quality problems in different representation.

To understand data quality challenges, the two perspectives considered were: data quality challenges in general combined with specific reflections and metadata quality challenges.

### 1.3.1.1 General and Specific Reflections

**a) The Legacy System**

The information society is dealing with the increasing challenges of data overload as a result of digitalization of all sorts of contents, and the improvement and drop in cost of recording technologies. The large amounts of available data are increasing and growing exponentially in today's competitive environments.

The legacy system, still present in  some enterprises/institutions have millions of data and records which were collected by structured and unstructured techniques or methods.

The enterprises/institutions are presented with the challenges of discovering meaningful data structures and information from such an enormous and changing environment so as to continue operating competitively. The data quality problems that have to be solved here could include, but are not limited to: data accuracy, data completeness, timeliness, data reliability, and information quality which often tend to conflict with the set goals of the entities.

**b) Data at the Enterprise Level**

There exists many systems and subsystems at the legacy level in enterprises. The data collected and stored at the individual system or subsystem level could be well defined and structured.

Data inconsistencies, reliability, etc., can arise in situation where by the entire scope of the enterprise data is considered for integration or creation of a unified model. Apart from the generation of data quality problems during data integration or creation of a unified model, other data quality problems that existed at the individual legacy system or subsystem level are also inherited in the overall enterprise system.

Data quality is also lost in the data migration processes thereby making the data quality quite questionable.

Cleansing the data warehouse can be one of the new tasks in addressing these anomalies.

### c) Data Storage (Distributed and Integrated Data)

Data stored in a distributed system have quality problems such as: different data definition, different data practices, and data granularity when the data is being integrated.

Data quality problems are further worsened by the competing nature of businesses and their need for market dominance. In achieving those goals, they are faced with the challenges of having timely and accurate data on their customers. Kyeong Kim et al.(2005)[18] proposed in their paper a methodology for mining the change in customer behaviour before and after a certain point in the contexts of decision tree classification.

These challenges and competitions are manifested in questions like "how can we achieve competitive advantages over our competitors?", hence they need to identify trend and pattern of customer information and interaction for future prediction.

As Kyeong Kim et al. stated above, a lot (in millions) of unstructured data are thereafter collected and stored about customers; examples can be from financial institutions like banks and insurance companies, chain stores, telephone companies, and universities.

These data may be collected by varying techniques and stored in different location hence the enterprises/institutions are presented with new challenges of integrating such data/information into their databases and ontologies. Bhide et al.(2007)[21] developed a tool called LIPTUS that associates customer interactions with the customer and their account profiles thus advancing the need to structure such data/information into a single and standardized relational database.

Customer retention and acquisition is key to business growth and survival, hence understanding their behaviour through simple interactions like phone conversation is important. Jansche and Abney (2002)[22] discussed in their paper extraction of customer mood from voice-mails messages.

The advancement of technology coupled with poor or unstructured data storage has affected insurance companies by way of individuals or companies making fraudulent claims or try to abuse the set systems by falsification/alteration of their claims. Popowich (2005)[14] discussed a health care application which processes both structured and unstructured information associated with medical insurance claims.

**d) Data Storage (Data Representation)**

Many Enterprise Resource Planning (ERP) systems today tend to store important facts and data about objects that the company considers sufficient and relevant to their system and business functionalities.

This idea is very cost effective but quite disadvantageous in that most attributes of the data are omitted in mitigating the developmental cost of such systems; data structures and formats, terminologies, and data representation may vary across different systems.

Since ERP comprises of many subsystems in the enterprise, these subsystems store data in their respective formats hence there exists a lot of inconsistencies in the way data is stored, collected, and used in the enterprise.

The data formats and definitions in the enterprise may also not be standardized. Halevy (2001)[2] discussed the problem of answering queries in data management, query optimization, and data integration system. He further outlined the need to standardized work in solving the problems.

### 1.3.1.2 Metadata Quality Challenges

The National Information Standards Organization (NISO) (2004)[26] defined metadata as a "structured information that describes, explains, locates, or otherwise makes it easier to retrieve, use, or manage an information resource. Metadata is often called data about data or information about information".

With the above definition of metadata, it is thus important to understand how data and meta-data tend to lose their quality along the path of their description. Consider the figure below:

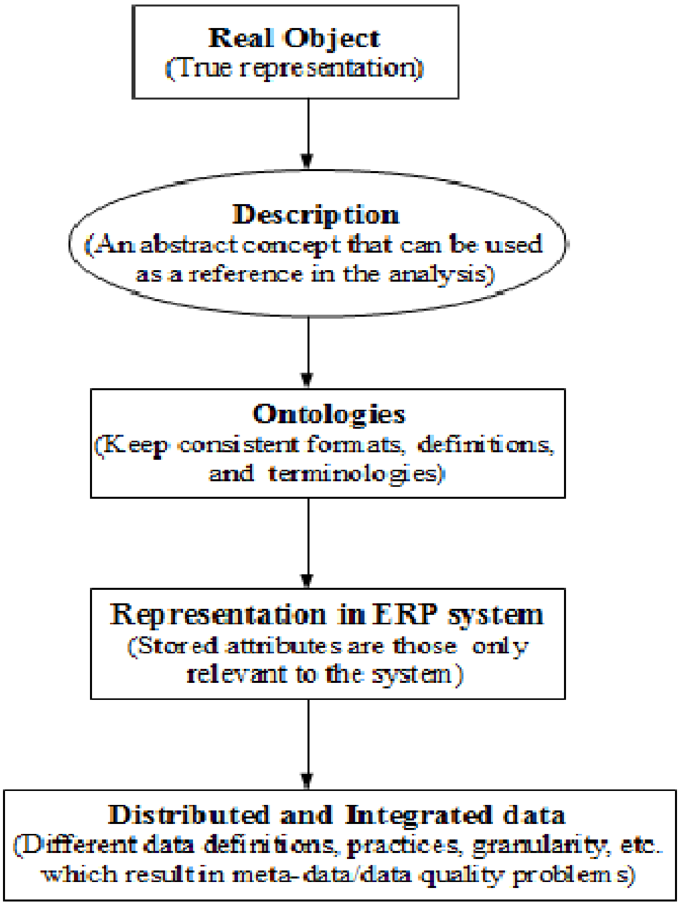

Real Object
(True representation)
Description
(An abstract concept that can be used as a reference in the analysis)
Ontologies
(Keep consistent formats, definitions, and terminologies)
Representation in ERP system
(Stored attributes are those only relevant to the system)
Distributed and Integrated data
(Different data definitions, practices, granularity, etc. which result in meta-data/data quality problems)

Figure1:Level of meta-data/data description.

Figure 1 above shows a level description of meta-data/data. At the top (Real object) is the item that desired attributes are to be collected from.

- Description

Any aspect of the object can be described and new description are always found and added to the existing ones so as to improve the object's description, hence it is not possible to make full object description. Consider the following aspects of a screw below:

- Head:
  - Shape: Hexagon
  - Drive style: Hexagon
  - Width: 10mm
  - Height: 6mm
- Point: Cone
- Diameter: 4mm
- Shaft length: 30mm
- Thread length: 15mm
- Thread angle:
- Coarse/Fine: Fine

This level (Description) is an abstract concept that can be used as a reference in the analysis since many aspects of the object could be described.

- Ontologies

The main purposes of ontologies are to keep consistent: formats, definitions, terminology, and give full descriptions of the object under investigation however, different ontologies describe different aspects of the same object or class.

- Representation in ERP system.

Data stored in most ERP systems are those relevant and sufficient to the functionalities of such systems. The draw back here is that many attributes of an object are left out of the system.

- Distributed and integrated data.

This is the last level of metadata/data representation. It is the organization's formalized meta-data.

### a) Data Sources

These data quality problems present themselves at the bottom level, i.e., the distribution level in reference to Figure 1. At this level, there are many data sources like data warehouse and other similar data construction whose purposes are to have a unified picture of the organization's data.

Dushay and Hillmann (2003)[25] classified some four categories of metadata quality problems associated with the National Science Digital Library (NSDL), these were:

- Inaccurate data contents (metadata values do not conform to standard element use).
- Incomplete or missing data attributes.
- Confusing data – multiple values crammed into a single metadata element, embedded html tags, etc.
- Insufficient data – e.g., no indication of controlled vocabularies used.

Other quality problems could also come as a result of merging two or more databases together; these among others include:

- Data accuracy (correct values are recorded as it was reflected).
- Data consistency (two or more data items do not conflict with each other).
- Data currency (how recent is the information).
- Data completeness (availability of data to meet current and future information demands in a

data collection).

- Structural problems such as violations of the normal forms.

Umar et al.(1999)[3] cited some additional important data quality attributes such as data definition (data must be clearly and unambiguously defined), data access (the ease in which the users can access the data), and data presentation (a reflection of the style with which the data is presented).

In order to address the above identified data quality problems, activities such as consolidation of different data representation and elimination of duplicate information among others become necessary.

### b) The Role of Ontology

In reference to Figure 1, most legacy systems have meta-data/data organized in their respective ontologies. The data quality are quite good when restricted to a specific ontology.

The anomalies like data inconsistencies, come up when the entire data scope of the enterprise is considered for integration, i.e., different systems at the legacy level being integrated.

Examples of these anomalies could be presented in cases like news tracking (automatic creation of multimedia news by integrating video and pictures of entities and events annotated in news articles, and hyper linking news articles to background information on people, location , and company.), disease outbreak tracking as discussed by Grishman et al.(2002)[30], and possibly terrorist events from news sources extraction (Grishman 1997)[29].

These examples illustrate the usage of data from different sources and also show the need for quick and timely processing of data so as to provide the wanted information. The information extraction tasks can not be performed effectively when there are many data anomalies at the data sources.

## 1.4 Statement of the Problem

How can structured/unstructured text or data records be pre-process and categorize with the aim of collecting statistics and other relevant metrics or measures needed to add knowledge to database ETL developers to improve on data standardization and verification?, i.e., finding inherent data structures and translating structured/unstructured text or data records into a relational database.

### 1.4.1 Importance of the Research

The purpose of data profiling at the micro level is not only to add knowledge to the ETL developers but also improve the overall views of enterprise data and records by way of suggesting better ontologies.

Since data profiling is an iterative process, it can be more time consuming and less cost effective for enterprise SMEs to be presented with large volume of data and queries upon which their critical decisions and input are needed.

Shorter volume of data and concise queries allow enterprise SMEs more time for redeployment to other production area hence in the long run, data profiling can improve the enterprise productiveness in this competitive environment.

## 1.5 The Research Aim/Purpose

The aim of this study was to generate metrics for data standardization and verification by applying techniques and processes from NLP, statistical semantics (how to figure out what words mean, simply by recognizing patterns of words in huge collections of text), and parsing (analysing a text made of a sequence of words or tokens).

Data profiling employed these techniques to discover, reveal commonalties and differences in the inherent data structures, present ideas for creation of unified data model, and provide metrics for data standardization and verification.

The idea was to identify violations of the normal forms using patterns/contents combinations that enable disambiguation and classification of these data in a better way than currently done.

### 1.5.1 The Research Objective

The examination of data from a dataset and collecting the desired information provided useful statistics for data standardization. The following were the objectives:

i. Finding likely terms or families of terms.

ii. Identifying relationships that can reveal meaningful structures in the dataset (dependencies between tokens/phrases and groups).

iii. Showing the importance of large volumes of data in the profiling (for the statistics to work).

iv. Using and relating known facts to the results of profiling.

## 1.6 The Research Scope

The research was carried out in collaboration with Intelligent Communication (IntelCom AS)- Bergen branch in Norway.

Data for the research was based on the mechanical domain of fastener (screws, nuts, and bolts) and more particularly on the various types and nomenclatures of screws available in the shipping industry.

The specific issue that was looked at in this research was data profiling with a focus at a micro level on how the profiling process could reveal meaningful structures, tokens/phrases, and interrelationships (dependencies) between the tokens/phrases; by uncovering data anomalies such as data inconsistencies, data redundancies when analysing the data contents, their structures, and the relationships.

Thus data profiling at the microlevel can be described as the study of inherent dependencies and linguistic practices in the corpus.

# Chapter 2

This chapter looks at the related literature in greater detail.

## 2.0 Literature Review

This section is structured into two parts: The background literature and Related work section.

### 2.1 Background Literature

The following articles were reviewed so as to get a general understanding of the tasks involved in the research. The articles showed the need and importance of identifying meaningful structures from both structured databases and unstructured text records/dataset, and applying those extracted knowledge to the real-world practical applications.

Mansuri and Sarawagi (2006)[16] designed a data integration system for information extraction to exploit useful information in both structured data and labelled unstructured data in spite of their format, structure, and size variations.

Fayyad et al.(1996)[39] discussed the historical context of Knowledge Discovery in Databases (KDD) and data mining, and its intersection with other related fields. They provided a brief summary of recent KDD real-world applications. Definitions of KDD and data mining were provided, and the general multistep KDD process was outlined.

The multistep process had the application of data-mining algorithms as one particular step in the process. Finally, the article outlined a discussion of the data-mining step in the context of specific data-mining algorithms and their application.

McCallum (2005)[1] described information extraction as the process of filling the fields and records of a database from unstructured or loosely formatted text. He showed that IE and data mining are intertwined processes; where by IE populates a database from unstructured or loosely structured text and data mining then discovers patterns in that database.

McCallum further listed the five major IE subtask as: Segmentation (finding the starting and ending boundaries of the text snippets that will fill a database field), Classification (determining which database field is the correct destination for each text segment), Association/relation extraction (determining which fields belong together in the same record), Normalization (putting information in a standard format in which it can be reliably compared), and De-duplication (collapsing redundant information so you don’t get duplicate records in your database).

Sarawagi (2008)[37] stated that the field of information extraction had its genesis in the natural language processing community; where the primary impetus came from competitions centered around the recognition of named entities (people names and organization) from news articles. As society became more data oriented with easy on-line access to both structured and unstructured data, new applications of structure extraction came around.

In his review of a survey of information extraction research, Sarawagi also created a taxonomy of the field along various dimensions derived from the nature of the extraction task, the techniques used for extraction, the input resources exploited, and the type of output produced. Elaboration on rule-based and statistical methods for entity and relationship extraction was discussed .

Ananthanarayanan et al. (2008)[35], showed in their paper that existing domain knowledge, encoded as rules, can be used effectively to address the synonym-problem to a considerable extent. They argued that this makes the disambiguation task simpler without the need for much training data.

Their focus was on a subset of application scenarios in named entity extraction, categorize the possible variations in entity names, and define rules for each category. The created rules generated synonyms for the canonical list and match these synonyms to the actual occurrence in the data sets. In particular, they described the rule categories that they developed for several named entities and reported the results of applying their techniques (extracting named entities by generating synonyms) for two different domains.

When categorizing words or groups of words, their meaning or the exact sense of the token, a group of tokens, or phrases is of paramount important in understanding a given dataset or a corpus. Pantel and Lin (2002)[27] developed a clustering algorithm, called Clustering By Committee (CBC) that automatically discovers word senses from text and Jurafsky et al. (2000)[19] in their book, presented many approaches to word sense disambiguation.

According to Jurafsky et al, the approaches included selectional restriction-based disambiguation whose main focus is on correct senses, which is achieved by eliminating flawed representation from incorrect sense; robust word sense disambiguation such as supervised and unsupervised machine learning approaches (systems are trained to perform that tasks of word sense disambiguation); bootstrapping approaches which are similar to the machine learning approaches but are able to create larger training set from a small set of seeds.

## 2.2 Related Work

Data profiling process might start as an afterthought in a data integration project in most organizations. Research in data profiling are closely associated with data cleansing.

A lot of research has been carried out in mining/retrieving data and information from noisy or unstructured text, for example, Michelson and Knoblock (2008)[23], Dey and Haque (2008)[10], Mooney and Bunescu (2005)[32], Ananthanarayanan et al. (2008)[35], Fayyad et al.(1996)[39]; data profiling is considered as an activity in most of these work.

Erhard and Hong (2000)[13] considered data profiling and data mining as the two related approaches for data analysis, and that the focus of data profiling was on the instance analysis (the data type, length, value range, discrete values and their frequency, variance, uniqueness, occurrence of null values, typical string pattern) of individual attributes.

To be able to mine or retrieve the data and information, there is a need to collect and pre-process these noisy or unstructured text. “Information Extraction starts with a collection of texts, then transforms them into information that is more readily digested and analysed. It isolates relevant text fragments, extracts relevant information from the fragments, and then pieces together the targeted information in a coherent framework", (Cowie and Lehnert, 1996)[8].

The various tasks of preprocessing text such as in Dey and Haque (2008)[10] involved decomposing and reassembling of data; mainly to remove errors, duplicating values, unwanted characters, symbols, or white spaces in the text.

In data profiling, the unwanted characters, symbols, or white spaces in the text can be removed but errors are not easily determined or identified. To determine or identify errors in the text, data profiling need to encompass the various data processing techniques from NLP such as data mining, text mining, information extraction/retrieval, and data analysis among other.

One of the most common techniques of preprocessing text in NLP is the use of part-of-speech tagging (POS) as shown in Ghani et al. (2006)[28], Rajman and Besancon (1997)[31] where morpho-syntactic categories (noun, verb, adjectives, preposition, etc.) are assigned to words in context.

The results of profiling are subjected to a number of text mining techniques to extract and discover the hidden information from the underlying dataset. Categorization is one of those traditional text mining techniques that is often performed on the dataset in order to extract meaningful data structures.

According to Shehata et al. (2007)[36], “categorization is supervised learning paradigm where categorization methods try to assign a document to one or more categories, based on the document content”. In their paper, they further say that classifiers are trained from examples to conduct the category assignment automatically and that involves presenting each category as a binary classification problem.

Categorization techniques are based on word or phrase analysis of the text and statistical analysis of a phrase frequency to capture the importance of the term within a document.

Extracting information is opening up new ways/methods for querying, organizing, and analysing data by drawing upon the clean semantics of structured databases and the abundance of unstructured data.

Mooney and Bunescu observed that many information extraction systems treat text as a sequence of tokens. They used this observation in discussing one of their approaches in the construction of information extraction system that treats the extraction task as a sequence of labelling task (words/tokens are assigned to a label from a fixed set of alternatives).

Extraction of tokens, phrases, and terms are part of the data profiling process; the interests here are to identify those tokens/phrases whose occurrences or co-occurrences are relevant to the understanding of the underlying structures and useful clues to other meaningful information in the corpus.

Term extraction is a very vital task in NLP; Daille (1994)[9] showed that this task, i.e., term extraction can be narrowed down to the extraction of term candidates on the basis of structural linguistic information, and filtering of the term candidates on the basis of some statistical relevance scoring schemes.

When data profiling becomes an afterthought and is considered as an activity not as a process; less attention is then given to its results (profiles) since the basic reason for performing the activity is to get simple views of the attributes.

For in-depth views of data, it is therefore important to consider data profiling not as a simple activity but as a process so as to better understand the hidden/lock knowledge, inherent data structures, and tokens/phrases relationships in the dataset in the process of profiles generation.

## 2.2.1 Extract, Transform, and Load (ETL) Concept

It is important to understand this concept of data manipulation before discussing the various tools that make up this ETL process.

### 2.2.1.1 ETL Process

This is a central process in database manipulation and data warehousing. It involves the processes of extraction, transformation, and loading of data. The processes are explained below:

Extract: In this phase, data is extracted from operational data sources using flat file or DBMS entry such as ODBC. Within the Extract phase, parsing the extracted data is a sub-process that analyses the data for conformity with the expected pattern or structure.

Transform: Many business rules are applied to the extracted data so as to derive data that will be loaded into the destination target. Some data sources will require little or no manipulation when converting their  formats into the desired destination.

Load: This phase loads the data into the destination target, which in most cases is the organisation data warehouse.

The ETL tools are central in discussing data profiling since the profiles are inputs for the developers performing data standardisation and verification.

### 2.2.1.2 ETL Tools

A large number of commercial and open-source software tools are able to support the ETL processes for data warehouses. Examples of these tools include: IBM's QualityStage and DataStage, InformationBuilders, WarehouseAdministrator, TrilliumSoftware, Informatica Data Explorer, dataFlux, dataCleaner, QASSystems, and Oracle Warehouse Builder.

These tools use a repository built on a DBMS to manage all the metadata about the data sources, targets, mappings, script programs (proprietary languages), etc., in a standardized approach.

Their basic functionalities are: data profiling (presentation of the overall views of the data sources), data cleansing (correction of data quality problems discovered), data parsing and standardisation (splitting text/data into single or atomic units and converting the data into the desired formats), and data matching (putting together similar records and identifying relationships).

Most of these ETL tools handling data quality problems are domain-specific, i.e., supporting name and address validation data or elimination of some duplicate values from the data.

### 2.2.1.3 Gaps in the Tools

While these tools are quite advanced in their technology and operation; they cover only part of the problems of data profiling at a micro level and some substantial manual effort or programming are still necessary to handle a complete data profiling process.

Data profiling is not only limited to unstructured dataset/records but is also extended to data stored in well-structured data models. The structures of these models may not be documented, hence there is a need to perform data profiling to identify these structures and discover some imperfections in the data models.

Creating a unified data model that reflects all the data sources necessitates combination of several databases; profilers need to restructure these databases and identify/determine their commonalities and differences.

Profilers still need to identify violations of the normal forms using patterns/contents combinations to enable disambiguation and classification of data in greater depth.

There is still a need to identify terms in the dataset; suggest groups for tokens/phrases having some commonalities (relationships); obtain contextual views; and determine confidence levels of the identified terms.

# Chapter 3

## 3.0 The Research Methodology

## 3.1 Introduction

This chapter provides an overview of the approaches and technologies chosen for the project tasks and the focus was primarily on understanding the activities and processes in data profiling such as preprocessing of data, i.e., how token reports (token,  groups of tokens, or phrases), their patterns, relationships, etc., are generated.

These activities were achieved by design and implementation of rule sets (script programming language with sets of logic for parsing, classifying, and processing of data) developed in QualityStage and creation of jobs/tasks in DataStage.

The rule sets were continuously redesigned and refined to perfect and generate meaningful reports about the corpus.

Hevner et al., (2004)[15] observed that design is inherently an iterative and incremental activity. Evaluation of the design process provides essential feedback to the construction phase so as to improve the quality of the process and the product under development.

## 3.2 Design Science

Design Science Research was used to develop general knowledge used in designing solutions to some specific problems. March and Smith (1995)[24] in their paper, described design science as a scientific approach to scientific information research.

They compared natural science with design science; natural science tries to understand reality and design science attempts to create things that serve human purposes, and that design science is technology-oriented. Its products are assessed against criteria of value or utility - does it work? Is it an improvement?

March and Smith further identified two design processes and four design artefacts produced by design-science research in Information System. The two processes are build and evaluate. “Building is the process of constructing an artefact for a specific purpose; evaluation is the process of determining how well the artefact performs”.

The artefacts are: constructs, models, methods, and instantiations. Construct is a basic language of concept used to characterize the phenomena; models are used to describe the tasks or artefacts; methods are ways of performing goal-directed activities.

In data profiling, ontologies contain common names and other vocabularies; often referred to when identifying and confirming registered names within a domain. The designed rule sets, from QualityStage are methods that define the data profiling processes, i.e., guidelines for the solutions of the problems and how to handle the profiles.

The profiles can be considered as models describing the relationships between the dataset under investigation and the ontologies, and they provide an overview of the corpus and suggestion of possible solutions.

### 3.2.1 Data Profiling Process

With the above approaches and the tools used in this research outlined below, it is important to examine how data profiling at a micro level can lead to revealing meaningful structures in a specified dataset for the standardization activities.

Data profiling is a section of the preliminary subtasks in text mining that integrates ideas from information extraction and retrieval, data mining, data quality and integration, and text analysis techniques. Its goals are: to discover, filter, and examine structured data or knowledge from a large volume of unstructured text or records.

The goals were achieved by collecting and analysing statistics, and discovering new or previously hidden data representation by applying techniques and methods from Natural Language Processing (NLP) to the text or data records.

The question is "how can we pre-process and categorize unstructured text or data records with the aim of collecting statistics and other relevant information so as to standardize the data".
The following processes and activities were involved:

**Processes**:

i. Preliminary assessment- the dataset domain is specified and the interest of the analyses are identified.

ii. Data collection – the dataset or database of interest for the analysis is selected based on the current domain knowledge and data understanding.

iii. Data preparation – data is processed to remove noise and stop-words or delimiters that have no significance in the results.

iv. Data analysis – data is analysed with the focus on discovering new features and structures in the data.

v. Result interpretation – the tokens and data patterns are analysed and evaluated on their relevancy and also they are validated by SMEs.

vi. Rule refinements – new knowledge are discovered and rules refined and the process continuous.

**Activities**:

i. Lexical analysis (converting a sequence of characters into a sequence of tokens),

ii. Parsing/syntactic analysis (analysing a sequence of tokens to determine their grammatical structure with respect to a given formal grammar),

iii. Frequency analysis.

Other activities performed on the results of the profiling process are text and functional dependencies analysis.

Data profiling is thus an iterative process. The reports generated are analysed, questions are presented to the SMEs for their interpretation and the business rule sets redesigned. The iteration procedure helped in refining and reviewing the rule sets, and acquiring more knowledge on the domain.

The processes and activities, as illustrated in Figure 2 would then begin from lexical analysis and going through the other steps again in tuning the dataset to the desired output.

The overall overview and structure of the data profiling processes and activities are shown below.

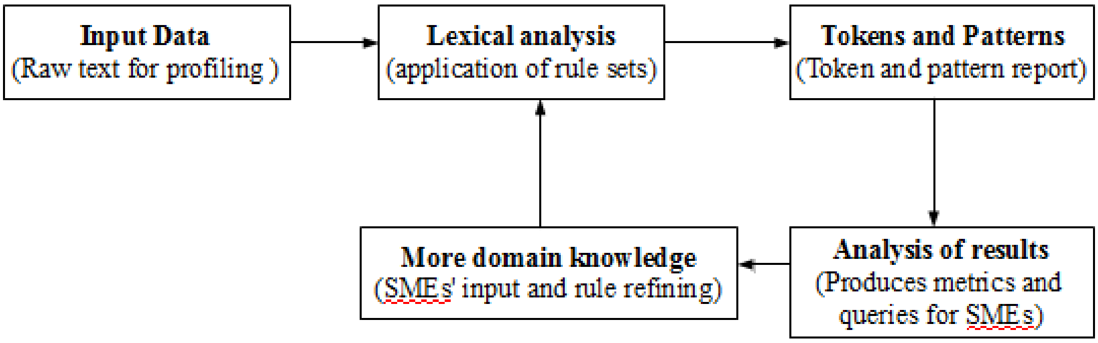

Input Data
(Raw text for profiling )
Lexical analysis
(application of rule sets)
Tokens and Patterns
(Token and pattern report)
More domain knowledge
(SMEs' input and rule refining)
Analysis of results
(Produces metrics and queries for SMEs)

Figure 2: Data profiling structure.

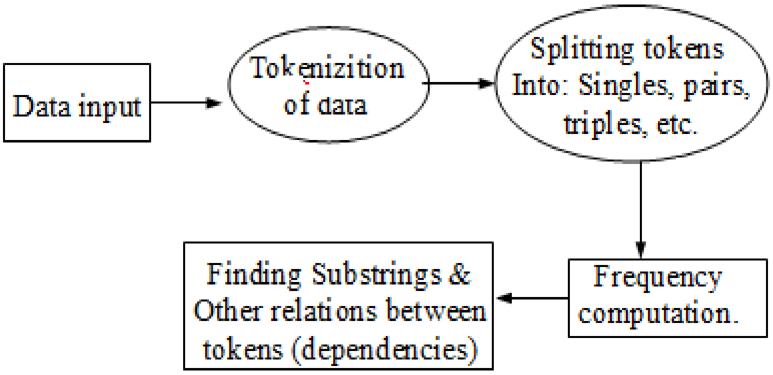

Data input
Tokenizition of data
Splitting tokens Into: Singles, pairs, triples, etc.
Frequency computation.
Finding Substrings & Other relations between tokens (dependencies)

Figure 3: Data profiling steps.

Figure 2 illustrates the general arrangement of data profiling processes and activities while Figure 3 shows the steps used to obtain the profiles.

### 3.2.2 Data Profiling Steps

The following constituted the major steps in data profiling at the micro level using the IBM WebSphere tool and techniques used to achieve the objectives of the research.

The following data profiling steps were illustrated in Figure 3 above.

i. **Data input**

- The dataset for profiling is selected and specific area of interest noted.
- Acquire some domain knowledge by having a brief discussion with SMEs.

ii. **Tokenization and parsing of the data**

- The dataset (input data in Figure 2) is split into individual substrings called tokens.

iii. **Transformation and aggregation of data**

- Word delimiters such as white spaces, prepositions, punctuation, etc., are removed.
- The dataset is further split into pairs, triples, quadruples, quintuples, etc., to cover all the N-tuples in the string or a record.
- Sort the split data.
- The occurrence and co-occurrences (frequencies) of the tokens and tuple combination in the dataset are counted.

iv. **Analysis of results**

- Identifying likely terms or families of terms.
- Identifying relationships between the terms or families of terms.

v. The process is further repeated from ii to iv to identify more terms and relationships.

## 3.3 Definitions and Explanation of Terminologies

The following are some of the known and developed terminologies, and metrics used in the research:

i. A token is a single unit of numeric, alphabetic, or alphanumeric characters group together.

ii. A phrase is a group of tokens working as a single unit to give some meaning.

iii. A term is a sequence of tokens or phrases

iv. Phrase frequency: the number of time a token/phrase or a group of tokens/phrases are occurring within the corpus.

v. Relative functional dependency (RFD): a relationship between the individual terms in a given data set.

RFD can classified as Asymmetric (direct dependency) or Symmetric (bidirectional dependency).

It is asymmetric when a token/phrase or groups of tokens/phrases are very dependent on the other tokens/phrases or groups while the reverse dependencies are not true.

The dependency is symmetric when there exits a mutual dependency between the token/phrase or groups of tokens/phrases in the dataset under investigation.

vi. Group relative functional dependency (GRFD) occurs when a group of tokens/phrases have some partial or total dependency on a token/phrase or group of tokens/phrases.

vii. Group confidence: the likelihood that the group forms a valid group.

viii. Membership confidence**:** the confidence that a phrase is a valid member of a given group.

ix. Prepositions like: for, in, above, below, etc., help in identifying terms and their attributes.

x. Substring Divider. A token/phrase or a preposition whose dependency on the prefix or suffix is considered negligible and insignificant to the term meaning.

xi. Substring Connector. A phrase whose dependency on the prefix or suffix phrase is considered to be symmetric.

xii. Substring Identifier. A phrase that is able to identify another phrase based on its 1:1 dependency relationship.

xiii. Substring Descriptors. A phrase that is well distributed within the dataset, has a distinct RFD, and is able to describe an object.

## 3.4 IBM WebSphere Tool

The IBM WebSphere tool was central to the research. This tool refers to a brand of software products which are designed to set up, operate, and integrate electronic businesses applications across multiple computing platforms using Java-based Web technologies.

It includes both the run-time components and the tools to develop applications that runs on WebSphere Application Server (WAS).

The basic purposes of this tool are: data integration, and data cleansing (data matching and standardization), i.e., Extracting, Transforming, and Loading data. ETL tools extract data from specified source(s), transform it into new formats according to business rules, and then load it into target data structure(s).

The focus and interest in the tool for this research was on the IBM InfoSphere DataStage and WebSphere QualityStage.

### 3.4.1 IBM InfoSphere DataStage

The IBM InfoSphere DataStage tool has stages such as: general (general purpose stages), file (file manipulation stages), databases (database manipulation stages), and processing (transforming and filtering tasks in the stages) that were used in designing jobs (tasks).

The file stages were used both for the specification of the input files and the output files.

The processing stages were used in the file transformation and filtering the expected data output into a desired data structure and format.

### 3.4.2 WebSphere QualityStage

The WebSphere QualityStage is a subset of the InfoSphere DataStage.

The central point here was the creation of rule sets that provide the logic required to achieve data standardization. When developing the rule sets for data standardization and matching; Pattern Action file (.PAT), Dictionary files (.DCT), Classification table (.CLS), and Rule set Description file(.PRC) were used.

The two tools, i.e., (QualityStage and DataStage) complement each other. QualityStage provides the development environment for building data-cleansing tasks while DataStage provides the graphical notation for building the tasks.

## 3.5 Design Process

The processes of data profiling were developed and a structured design of how they were achieved has been outlined.

The designed process involved the use of QualityStage and DataStage, IBM tools that were central in the design and implementation of the rule sets and tasks.

The rule sets  were created in QualityStage to perform activities such as splitting each input string into single or different tuple combination of tokens; the different stages in DataStage filter, transformed, and aggregated the input strings into the desired formats or reports.

The result of data profiling (token and pattern reports) helped to reveal and discover hidden relationships and functional dependencies between the tokens in the dataset. In this process (data profiling), the frequencies of occurrences of different combination such as: individual tokens, pairs, triples, etc., were measured.

The grouping together of related tokens helped in identifying relationships among tokens and also helped to show which tokens derived their meaning from other token hence the term functional dependency.

### 3.5.1 Evaluation Strategy

In using design science as a methodology for this research, it was therefore necessary that the design process was evaluated to identify weaknesses so as to refine and reassess the process (Hevner et al. 2004)[15]. The following aspects of the design were evaluated.

- **The designed process**

The process was evaluated on a large set of data from the shipping industry in the domain of fasteners. The statistics collected were on screws and these statistical figures were used to show among others the level of confidence by which related tokens can form meaningful groups and also identify other relationships exhibited by the tokens.

- **Developed Metrics**

The metrics provided from this research were input for data standardization and verification process. These metrics were to aid in improving the identification/determination of the data qualities problems by adding knowledge to the ETL developers designing and implementing data standardization, hence the whole strategy for developing the metrics need to be evaluated.

The evaluation was performed after the design process so as to determine the reliability and consistency of the metrics.

- **Design evaluation methods**

Hevner et al. further suggested a number of design evaluation methods; testing and descriptive methods of evaluation were used in this research.

In the testing method, functional testing was executed on design process to discover design flaws in the process and refined them. These involved the domain SMEs and the ETL developers so as to show the following: correctness, completeness, strengths, and weaknesses of the developed metrics and also to show the validity of the design process when standardizing items or data values from a domain.

In the descriptive method, informed arguments were used to show the usefulness of the metrics to the standardization process; which involved the used of relevant literatures and domain knowledge from the SMEs.

# Chapter 4

## 4.0 The Basic Metrics

## 4.1 Introduction

The analysis presented below focuses mainly on: phrase frequency, relative functional dependency (RFD) and group relative functional dependency (GRFD) in the dataset.

This was particularly important in understanding the hidden relationships and other tacit information in the datasets. The exception to this research focus was on how data profiling can help in providing metrics that could be used by ETL developers to standardize data contents and also transform data records into a relational database in 1NF.

## 4.2 Presentation and Interpretation of the Basic Metrics

In understanding the underlying data structure from the dataset, profilers need to perform deep data profiling scans at the micro level on the selected dataset. The deep scans can be quite resource consuming depending on the type of profiling being done and also the amount of records being scanned.

It is therefore necessary to have preliminary assessments of the dataset to be able to decide on how the data profiling process would be performed. The data profiling processes and activities are further examined in detail below.

### 4.2.1 Data Assessment

Data assessment refers to the art of collecting, reviewing, and acquiring knowledge and some contextual information about the domain of fastener; in particular the screws under investigation.

This assessment comprises of preliminary assessment of data and data collection processes and these two processes, (preliminary assessment and data collection) complement each other.

In this context, preliminary data assessment refers to making decisions and selecting the part of records for the profiling task while data collection is concerned with looking for and gathering preliminary knowledge about the dataset.

Examples could include the languages used for naming domain elements; what are the elements under considering; the interests of the clients; necessary metrics to produce, etc.

The two processes can be illustrated by the sample input strings in Table 1 below, taken from an assumed record file.

| Sample string |
|---|
| HEXAGON SCREW 12.9 DIN933 |
| 6K.SKRU M20X 90 ELF |
| 6K.SKRU M12X 35 A4-80 BORET |
| 6K.SKRU M12X 35 A4-SIMPLEX |
| HEXAGON SCREW FOR PLATE MOUNTING |
| ADAPTERPLATE NEDRE TT3300 |

Table 1: Sample input file.

The above table can have thousands or millions of records. It is thus important to acquire some domain knowledge and contextual information by either having discussion with the domain expert or obtaining some background knowledge of how fasteners (screws, nuts, and bolts) are described in an ontology.

The ontological description could include: fastener types, units of measurements (metric in millimeters), standards, sizes, shapes, and thread coarseness, for example, Unified Coarse (UNC).

When performing data profiling, there is a need to have some simple visual scans/inspections of the dataset. The visual scans/inspections can reveal clues in the dataset like: languages used, possible descriptions and meanings of strings, possible presence of abbreviations and synonyms.

The visual scans/inspections are very cost effective and time saving to profilers. They (profilers) can focus more effort and resources in identifying other underlying structures and relationships in the dataset.

In Table 1 above, it can be noticed that some strings are mentioned in English while others are in Norwegian languages; abbreviations can also be noticed. The domain knowledge showed that 6K was an abbreviation for the token SEKSKANT, and that this token, i.e., SEKSKANT is a Norwegian word for HEXAGON and also that the token SKRU is the word SCREW in English language.

The goal of the two processes is to give profilers general overview of the dataset and tasks at hand, and their possible results thus positioning themselves for the tasks and activities ahead of them.

### 4.2.2 Data Preparation

This is the main process in data profiling, the following are some of the activities performed in this process.

#### 4.2.2.1 Tokenization of Records

Tokenization is a subtask in data preparation process whose results are central to data profiling. Texts or strings in the records are split into individual strings or substrings called tokens by lexical analysis so as to obtain useful statistics. These statistics are the main focus of data profiling.

Rule sets developed in QualityStage are able to divide strings into a sequence of tokens. When dividing the strings, delimiters like: punctuation and other string delimiters are removed from the strings. This was achieved using two QualityStage commands called STRIPLIST and SEPLIST.

Another activity that takes place in the tokenization process is the creation of the combination of pairs, triples, etc., for the N-tuple occurrences, this was done by:

- Creating a list of all the tuples,
- Counting their occurrences.

The counting of the occurrences of the tuples covers all the individual tokens and their combination. This activity was performed by DataStage processing stages such as: transformer, filter, etc.

Tokenization therefore combines the processes of data parsing, transformation, filtering, and aggregation so as to generate a report.

With reference to the process in Section 4.2.1, 6K was replaced by SEKSKANT since it was known from the SME that 6K was an abbreviation of the token SEKSKANT in Norwegian language.

The following two tables below demonstrate the output of the tokenization process which is either a token report as illustrated by Table 2; or pattern report as illustrated by Table 3 of the dataset or records under investigation.

The example in the Table 2 below, illustrates a sample result of the tokenization process of 6K.SKRU M20X 90 ELF as an input string.

| Combination of tokens | Token(s) | Frequency |
|---|---|---|
| | SKRU | 1067 |
| | SEKSKANT | 633 |
| | 90 | 295 |
| | M20X | 95 |
| | ELF | 63 |
| Pairs | SEKSKANT SKRU | 450 |
| | SKRU M20X | 33 |
| | M20X 90 | 5 |
| | 90 ELF | 3 |
| Triples | SEKSKANT SKRU M20X | 33 |
| | SKRU M20X 90 | 5 |
| | M20X 90 ELF | 1 |
| Quadruples | SEKSKANT SKRU M20X 90 | 3 |
| | SKRU M20X 90 ELF | 1 |
| Quintuples | SEKSKANT SKRU M20X 90 ELF | 1 |

Table 2: A sample of the token report from the tokenization process.

| Pattern | Frequency |
|---|---|
| SEKSKANT SKRU @ | 295 |
| SEKSKANT SKRU @ ^ ^ | 219 |
| SEKSKANT SKRU M > | 90 |
| SEKSKANT SKRU UNC @ | 31 |
| SEKSKANT MUTTER M ^ ^ | 5 |
| SEKSKANT PASSKRU @ ^ | 5 |
| SEKSKANT SKRU UNF @ | 4 |
| SEKSKANT MUTTER M @ ^ | 2 |
| SEKSKANT SKRU @ ^ < | 1 |

Table 3: A sample of the pattern report from the tokenization process.

Once the tokenization process has been completed as illustrated in tables above, the output are analysed so as to gain meaningful information from the dataset/record.

In Table 3, the @ sign means a complex mix alphanumeric characters, ^ sign means a numeric, > sign means a leading numeric character, < sign means a leading alphabetic character, etc.

More WebSphere QualityStage patterns classes and their meanings are found in Appendix A.

### 4.2.3 Data Analysis

It was from the results of profiling that the goal for data profiling at a micro level was achieved and some new metrics were developed. According to Cardie (1997)[4], information extraction systems effectively skim a text to find relevant sections and focus only on these section in the sebsequent preprocessing.

In this research, the focus on data analysis was to discover new phrases, grouping the phrases, and identifying the inherent relationships between them. Token reports, pattern reports, and contextual information formed the basis of these tasks.

These were achieved by studying and analysing statistical evidences such as token/phrase frequencies so as to identify/determine token/phrase patterns; dependencies and other relationships in the dataset/records.

It was therefore important that the data profiling results were examined in greater depth. The examination (studying, reviewing, and understanding the results) were on the basic metrics, i.e., phrase frequency, relative functional dependency, and group relative functional dependency together with other developed metrics.

These conclusions and the useful information (frequencies, RFDs, GRFDs, and other information) are the new metrics to be used by the ETL developers for data standardization and verification of the data correctness.

#### 4.2.3.1 Phrase Frequency

This is the summation of the number of occurrences or co-occurrences of distinct data patterns/values in the dataset/records under investigation. The interpretation of this summation (token/pattern frequencies) was central to data analysis because the frequently occurring data patterns/values could provide/capture some meaningful information about those tokens/phrases.

In the tokens reports, tokens/phrases with higher frequencies were identified and considered for further analysis so as to identify/determine their contribution and importance to the dataset. These findings give profilers the clues and motivations for further analysis.

The interest and the goal of this process (data analysis) is understanding and classifying tokens or families of tokens in the dataset; the focus was more on tokens/phrases with higher frequencies.

The phrase frequency analysis narrows down the focus of the investigation to tokens/phrases having higher occurrences in the dataset. Profilers can align the tokens/phrases in order of their decreasing frequency values and use Zipf law as discussed by Wentian (1992)[41] so as to condense the analysis to fewer tokens/phrases with higher occurrences in the dataset by selecting them based on their ranks.

When some of these substrings with higher frequencies are identified, the SMEs are consulted to determine whether these tokens/phrases are names, identifiers for items, and what their significance are in the dataset. For example, the following tokens/phrases from Table 2 would be considered for further analysis: SKRU occurs 1067 times, SEKSKANT occurs 633, SEKSKANT SKRU occurs 450, and the triple SEKSKANT SKRU M20X has 33 occurrences in the dataset.

Other tokens/phrases with smaller occurrences such as ELF, M20X, 90 ELF; using the token report of Table 2, could be ignored on the assumption that the dataset is large enough (millions of records) hence the contribution of such tokens/phrases to meaningful information and the development of metrics for ETL developers were insignificant.

Tokens/phrases with smaller frequencies could also be ignored due to lack of data storage facility, and the need to have a shorter and constructive discussion with the SMEs.

Profilers can also derive more meaningful information and knowledge by considering the occurrences of patterns/contents combination in the dataset, for example, a scan of the pattern report in Table 3 revealed the following occurrences of patterns in the sample dataset: SEKSKANT SKRU @ occurs 295 times, SEKSKANT SKRU @ ^ ^ occurs 219 times, SEKSKANT SKRU M > occurs 90 times, SEKSKANT SKRU UNC @ occurs 31 times, etc.

Since the number of distinct occurrences of the patterns in the dataset were summed up, again those with higher occurrences are considered for further analysis.

The phrase SEKSKANT SKRU can be seen to have higher occurrences of 450 times (token report as seen in Table 2) in the dataset and in the pattern report, it has many occurrences with other patterns. Phrases/tokens such as SEKSKANT SKRU may have other dependency relationships on other phrases/tokens or patterns and can help in identifying item names in the dataset.

The phrase frequency and the string or substring patterns are indication that present some interests and ideas into the relevance of these tokens/phrases in the dataset; profilers can identify such tokens/phrases or patterns with higher occurrences and consider them for further investigation to reveal their meanings and importance to the dataset; or completely reject them.

The frequencies of the tuples are compared with each other. These comparisons help in predicting or identifying possible relationships amongst the tokens.

Consider the table below of some substrings (in pairs) from a given dataset and their frequencies.

| Phrase | Phrase Frequency |
|---|---|
| SEKSKANT SKRU | 450 |
| SEKSKANT MUTTER | 92 |
| SEKSKANT ST | 26 |
| SEKSKANT CUZN39PB3 | 23 |
| SEKSKANT STÅL | 15 |
| SEKSKANT PASSKRU | 5 |
| SEKSKANT HODE | 1 |

Table 4: Sample results of profiling for pairs of substrings.

From Table 2, it was seen that the token SEKSKANT has a high occurrences of 633 times in the dataset and when groups of pairs were scanned, the token SEKSKANT was dominant in all the substrings as illustrated in Table 4. Without knowing the meaning of SEKSKANT, profilers could infer that such tokens have some meaning  in the dataset and they could also have stronger dependency relationships with other tokens.

Such observation and inferences could be applied to several other tokens/phrases in the dataset that exhibit such findings. They (tokens/phrases) are then isolated/grouped for further investigation.

The phrase frequency and string or substring patterns also show how tokens/phrases such as SEKSKANT or SEKSKANT SKRU are closely coupled with other in the record; these are suggestion for dependencies relationships in the record/dataset.

This analysis leads to determining membership of a group which is discussed in Section 4.2.3.4.

### 4.2.3.2 Relative Functional Dependency (RFD)

It is important to consider functional relationships when attempting to group tokens into relations. Functional relationships among tokens in a dataset can be considered as a concept of functional dependency (FD). The FD concept has much application to data base systems.

FD is defined as a relation on the attributes of the database hence it is concerned with a particular semantic relationship between the attributes of a table in a database.

For example, suppose that we have a functional dependency between column A and column B in an assumed table, which may be written as A —> B. This implies that the value of column A determines the value of column B, i.e., B is functionally dependent on A or A determines B.

In this research, relative functional dependency (RFD) specific to substrings was considered so as to come up with figures and substrings relationships that are helpful to ETL developers performing data standardization. RFD was further used to isolate tokens and identify abbreviations and synonyms in the record/dataset.

RFD can be defined as the phrase frequency of two tokens, for example, (SEKSKANT MUTTER), divided by the phrase frequency of the dependent; in this case it is SEKSKANT.

This is the definition of the RFD of SEKSKANT on MUTTER.

### a) RFD on Substrings

RFD can be expressed as a percentage by multiplying the quotient by 100% or it could also be expressed as a fraction. Consider the sample result of Table 2, RFD of SEKSKANT on SKRU is calculated as the phrase frequency of SEKSKANT SKRU divided by the phrase frequency of SEKSKANT. Thus the RFD of SEKSKANT on SKRU is 450/633 = 0.711 or 71.1% and that of SKRU on SEKSKANT is 450/1067 = 0.422 or 42.2%.

The computations of RFDs of tokens were used for the identification of tokens and their relationships in the dataset. These computations make it possible to determine whether a token or a phrase has a symmetric or asymmetry dependency with either their prefixes or suffixes.

From the computation of the RFD of SEKSKANT on SKRU and the reverse case, we can deduce that SEKSKANT is asymmetrically depended on SKRU and that SEKSKANT can also be a candidate for an element that is able to describe other tokens.

The above RFD is specific to substrings relationships; where token1 (SEKSKANT) is revealing some meaning of part of token2 (SKRU).

Based of a large volume of dataset/records evidence from the domain knowledge, contextual information, phrase frequency aggregation of distinct tokens/phrases, patterns combination, and the RFDs interpretation can be combined to identify tokens/phrases names and those tokens/phrases that should be considered together to form a term.

For example, it was deduce that SEKSKANT is a token describing part of the head shape of a SKRU hence SEKSKANT could be designated as a token or phrase descriptor.

Finding other tokens or phrase descriptors of a screw like: thread type, point type, shaft type, helps in the identification of screw names and grouping the screws based on such relationships.

### b) General RFDs on Substrings

We also have general RFDs where tokens are not necessarily following the item name or values directly in the dataset; as illustrated in the example of an assumed strings input below.

HEXAGON HEAD CAP SCREW M20X30 12.9 DIN933

After the profiling process of the above string, i.e., listing all the various tokens such as HEXAGON, HEAD, CAP, M20X30, 12.9, DIN933, etc., counting their occurrences, and analysing the figures; the outcome revealed that HEXAGON HEAD CAP SCREW has higher occurrence with DIN933.

This shows that the phrase HEXAGON HEAD CAP SCREW and DIN93 have some relationship and their combination could form some meaning like describing/naming an object in the dataset.

The following tasks need to be performed so as to determine the inherent meanings general to phrases:

i. Determination of token or group of tokens/phrases that give meaning to the string by describing it and those tokens/phrases which have relationships between the substrings.

ii. Suggestion and derivation of patterns from the list of tokens, strings, or substrings.

iii. Identify groups of related tokens/phrases either based on their prefixes or suffixes such as those in Table 4, where SEKSKANT is the prefix.

iv. Based on domain knowledge and the input of the SMEs, appropriate nomenclatures and purposes of some of the tokens/phrases can be identified. Examples from the fastener domain would be knowing which token or groups of tokens are: diameters, material type, metric measures, or standards for the fasteners.

v. Perform activities of Section 4.2.2.1. by listing strings or substrings patterns with their respective frequencies instead of listing the tokens as shown in Table 3.

By having some domain knowledge, profilers can derive some patterns specific to the domain under investigation.

The goal here was to further condense the input strings by replacing some of its substrings by patterns. It was easier, time saving, and cost effective for both profilers and SMEs to derive meaningful structures and more knowledge from condensed dataset. This is quite advantageous in the generation of better and concise metrics for data standardization.

Table 5 and 6 below illustrate the usages of patterns from a tool specific such as WebSphere QualityStage and those generic patterns derived from understanding the domain and using contextual information of the dataset/record under consideration.

IBM's WebSphere QualityStage tool specific patterns were used in Table 5 below.

| Sample text input | Pattern using QualityStage |
|---|---|
| SEKSKANT SKRU M20X | SEKSKANT SKRU @ |
| SEKSKANT SKRU M10X 35 | SEKSKANT SKRU @ ^ |
| SEKSKANT SKRU M10X 40 A4-80 | SEKSKANT SKRU @ ^ < |
| SEKSKANT SKRU M10X 55 ELF | SEKSKANT SKRU @ ^ & |

Table 5: Sample of the pattern report using QualityStage.

| Sample text input | Pattern using domain knowledge |
|---|---|
| SEKSKANT SKRU M20X | SEKSKANT SKRU MNX |
| SEKSKANT SKRU M10X 35 | SEKSKANT SKRU MNX N |
| SEKSKANT SKRU M10X 40 A4-80 | SEKSKANT SKRU MNX N AN-N |
| SEKSKANT SKRU M10X 40 A4-80 | SEKSKANT SKRU MNX N ELF |

Table 6: Sample of the pattern report using domain knowledge.

In Table 5, alphanumeric characters were translated to @ pattern (meaning a complex mix), numeric characters were translated to ^ pattern, while a single token and a leading alphabetic characters were translated to & and < patterns respectively. These pattern types are specific to IBM QualityStage.

In Table 6, all integers were translated to letter N. Other letters that could have been used would be D for diameter, L for length, STD for standard etc.

When the input strings are condensed, profilers can analyse the string/substring patterns to identify and determine their inherent meanings and dependency relationships. This leads to the discussion on identifications of synonyms in a dataset.

### c) Identification of Synonyms in Datasets

Synonyms can be defined as different tokens or phrases having identical or very similar meanings. The meanings can be based on some aspect of the token such as the object property. Identifying the true name of an object in a dataset is part of the profiling process.

Datasets can have different nomenclatures (names and abbreviations) from different languages to identify an object. An item can have a different name and abbreviation in different languages. The following listing refers to the same item/object in English, Norwegian, and Swedish: HEXAGON SCREW, SCREW, HEX SCREW, SEKSKANT SKRU, 6K SKRUE, SKRUE, SKRUV.

When identifying synonyms for an object, the object properties among other aspects are considered for detail analysis. In the fasteners domain, the domain knowledge of screws showed that the screw sizes and standards are basic properties that can be used to identify the true screw name.

The derivation of the string patterns below is not based on proved criterion but on basic knowledge of the domain. For example, it was known that screw standards can have different naming formats like DINXXX, ISOXXX, ANSI BXX; where the XXX represent numerics. Screw sizes are known to be in different units of measurement such as millimeters or inches, i.e., UNC, UNF, M, etc.

With this knowledge, the profilers can choose which patterns or groups of patterns are more representative and simpler in achieving their objectives.

In a situation where domain knowledge is not sufficient in identification of synonyms, the frequencies of the tokens and the respective patterns frequencies can be compared.

The assumed input strings in Table 7 below are used to illustrate how to identify synonyms using domain knowledge.

| Sample text input | Pattern |
|---|---|
| 1.HEXAGON HEAD CAP SCREW M10X20 12.9 DIN933 | 1.HEXAGON HEAD CAP SCREW MNXN N.N STDX |
| 2. HEXAGON HEAD BOLT M30X20 12.9 DIN930 | 2. HEXAGON HEAD BOLT MNXN N.N STDX |
| 3. SLOTTED HEAD BOLT M20X40 8.8 DIN930 | 3. SLOTTED HEAD BOLT MNXN N.N STDX |
| 4. SEKSKANT SKRU M10X90 5.9 DIN933 | 4. SEKSKANT SKRU MNXN N.N STDX |
| 5. SOCKET HEAD BOLT M20X2 4.8 DIN930 | 5. SOCKET HEAD BOLT MNXN N.N STDX |

Table 7: Illustration of synonyms identification using patterns.

It can be seen from Table 7 above that all the strings have a common patterns, i.e., MNXN N.N STDX. The phrase, HEAD BOLT is common to strings number 2, 3, and 5.

These observations can be made on records with few elements but on a large dataset (thousands or millions of records), it may not be easily seen coupled with the fact that the dataset could have many different patterns. To solve these problems, there is need to consider the individual phrase frequencies of the tokens when performing string translations into patterns.

Suppose that the frequency analysis of the tokens revealed that DINXXX, which is the standard of the screw occurs in 20 different formats compared to the screw size, i.e., MNXN which has over 1000 different formats in the dataset. The idea here is that the translations of the strings to patterns may be restricted to those tokens with many different formats since the intention is to condense the dataset so as to quicken and make the analysis easier.

Since we have some knowledge of the domain under investigation, we conclude that standard as a screw property has more describing power than screw size; the different screw standards would not be translated while the different screw sizes, and the steel type would be translated into some patterns such as MNXN and N.N. The resulting generic patterns of the strings from Table 7 would be as illustrated in Table 8 below.

| Patterns |
|---|
| 1.HEXAGON HEAD CAP SCREW MNXN N.N DIN933 |
| 2. HEXAGON HEAD BOLT MNXN N.N DIN930 |
| 3. SLOTTED HEAD BOLT MNXN N.N DIN930 |
| 4. SEKSKANT SKRU MNXN N.N DIN933 |
| 5. SOCKET HEAD BOLT MNXN N.N DIN930 |

Table 8: Sample pattern report.

It was stated above that screw standards have more describing power, hence strings patterns with the same standards were identified and grouped together. When the grouped elements were compared against each other and then matched, some commonalities and dependency relationships were identified.

From Table 8, identifying the common strings patterns and grouping them together, showed that strings patterns 1 and 4 have the same standards of DIN933 and strings patterns 2, 3, and 5 have the same standards too of DIN930.

Further analysis, i.e., comparisons, use of domain knowledge, and matching of the strings patterns in Table 8 with the original sample input strings of Table 7, revealed that strings 1 and 4 are describing the same item but in different languages and strings 2, 3, and 5 are also describing the same item but with different head style, drive style, and different drive types.

By the definition of synonyms above, it can be concluded that such strings are indeed synonyms.

A question could then be asked about which preferred name(s) should be given to the synonyms? The answers to this question can vary. Since profiling is to present figures and statistics to ETL developers performing data standardization and verification; the answers could then be determined at that level. It would include among other: consultation with the SMEs and the data ontologies, etc.

### d) Formulation of Terminologies

In discussing substrings relationships in a dataset, the profiler must seek to understand different substrings or tokens relationships in a string since these relationships are able to give clues or meanings to the records in the domain under investigation.

The profiler needs to calculate the RFDs on the prefix and suffix of a give token to identify these relationships. RFDs were used to formulate some terminologies specific to this thesis.

These terminologies were: substring divider, substring connector, substring identifier, substring descriptor, prefix-wise dependency, etc. To illustrate these concepts, consider Table 9 below. The table shows the tokenization of the string SEKSKANT SKRU M20X 90 ELF as was performed in Table 2; with an addition of FDs column.

| Combination of tokens | Token(s) | Phrase frequency | FD on Suffix % | FD on Prefix % |
|---|---|---|---|---|
| | SKRU | 1067 | --- | --- |
| | SEKSKANT | 633 | --- | --- |
| | 90 | 295 | --- | --- |
| | M20X | 95 | --- | --- |
| | ELF | 63 | --- | --- |
| Pairs | SEKSKANT SKRU | 450 | 71.1 | 42.2 |
| | SKRU M20X | 33 | 3.1 | 34.7 |
| | M20X 90 | 5 | 5.3 | 1.7 |
| | 90 ELF | 3 | 1 | 4.8 |

Table 9: Identifying terminologies using RFDs.

In coming up with these terminologies, the goals are to identify two consecutive pairs of tokens in a string whose combination and relationships can give some clues or ideas that may lead to an object identification , naming, or categorizing them into a term.

i. **Substring connector**

From Table 9 above, it can be observed that the RFD of SEKSKANT on SKRU is 71.1% and that on SEKSKANT from SKRU is 42.2%. Both RFDs are relatively high hence we can deduce that the dependencies is in both direction, i.e., a symmetric dependency, which could mean that connecting the two tokens has a high likelihood of identifying an object.

The terms given to such tokens with high connecting power inferred by their symmetric RFDs is a substring connector.

ii. **Substring Descriptor**

When the next consecutive pairs of tokens such as SKRU M20X are considered, the analysis of the dependency relationship revealed that the RFD of SKRU on M20X is 3.1% and RFD of M20X on SKRU is 34.7%. Here, the RFD of the suffix (M20X) is distinct and much stronger on the prefix (SKRU).

From the domain knowledge, it was known that MNXN is a pattern of screw size where numeric are translated as letter N. The screw sizes are well distributed within the dataset and by knowing a screw size, a particular screw can be identified/described.

With the above analysis, it can be deduced that screw sizes such as the token M20X, are substring descriptors.

iii. **Substring Divider**

The next tokens pairs such as M20X 90 and 90 ELF, have relatively low RFDs on their respective prefixes and suffixes. The deduction here is that the token 90, which is joining the two pairs of tokens is considered to have negligible impact and is thus insignificant to the string meanings. Such tokens are termed as substring dividers.

iv. **Substring Identifiers**

These are tokens which have 1:1 dependencies with each other.

Further analysis of RFDs can reveal more useful relationships in the dataset. ETL developers need to know these relationships so as to improve data standardization and verification.

As seen from the above discussion on a phrase being a connector; the RFDs analysis of

SEKSKANT SKRU are on both the prefix and the suffix. In a dataset of large volume, such dependencies could be many; terminologies like prefix-wise and suffix-wise can be coined to describe such relationships in either directions, such dependency relationships were discussed below:

v. **Prefix-wise Dependency**

This type of dependency relationships occurs between tokens/phrases which are directly preceding each other and the RFD of the preceding tokens/phrase is significantly higher on the succeeding tokens/phrases. Example of prefix-wise dependency can be seen in Table 9 between the phrase SEKSKANT SKRU; SEKSKANT is the preceding token while SKRU is the succeeding token; and the RFD of SEKSKANT on SKRU is 71.1%.

vi. **Suffix-wise Dependency**

This dependency relationship is the opposite of the prefix-wise relationship. Examples of these types of dependencies can be seen in Table 9 between the phrase SEKSKANT SKRU, but it is more evident in the phrase SKRU M20X; M20X is succeeding SKRU in how they are co-written in the text; and the RFD of M20X on SKRU is 34.7%.

vii. **Attribute-wise Dependency**

In this dependency relationship, attributes of one field can identify/determine attributes of another field in the same record. It is common amongst substring identifiers and substring descriptors .

viii. **Record-wise Dependency**

Tokens such as the screw size or M20X can appear anywhere in the dataset, giving details as to which measure a particular screw is compliant with. The relationships are of the two phrases (M20X and SEKSKANT SKRU) co-occurrences in the dataset.

### 4.2.3.3 Relative Group Functional Dependency (RGFD)

A relation is in first normal form (1NF) if each domain contains simple values. This has two main advantages: it allows the database to be viewed as a collection of tables with simple and understandable structure and it permits the definition of a small class of primitive operators that are capable of manipulating relations to obtain all necessary logical connections among attributes (Codd, 1970)[5].

Codd, (1972)[6] also showed that by applying simple decomposition steps to a 1NF relations in which the FDs were known, the relation could be split up into a set of relations in 3NF that represents all of the FDs.

Codd's explanations on how to split up relations come in handy when grouping tokens. Since it was known that the dataset under investigation is not even in 1NF; grouping tokens or phrases depending on their dependencies relationships can aid in solving the inherent 1NF violations in the dataset.

After data profiling process, different tokens were classified based on their relationships, structural significance, or meanings to the dataset hence the term relative group functional dependency (RGFD).

A token may have a low RFD on another but when such tokens are grouped together basing on some relations, their combined group dependencies may increase to 100% which is one of the measures of increasing the likelihoods for a relationships between the given phrases and also prove that the tokens/phrases belong together.

Before discussing the relationships and dependencies in groups; the group members or tokens that constitute a group needs to be identified. Grouping of tokens are based on certain token relationships that are exhibited by performing an in depth analysis of the frequencies of the various tuples in the dataset and calculating their RFDs.

### 4.2.3.4 Group Membership

Single tokens/phrases with their frequencies can give clues/ideas about the dataset contents and structures. On some instances, they could suggest groups but such groups are rarely specific enough for accurate classification. Strzalkowski (1994)[38] suggested that instead of having single tokens/phrases accidentally forming a group; a better method would be to identify groups of tokens/phrases that create meaningful phrases, especially if these phrases denote important concepts in the database domain.

Identification of tokens to constitute a group can be a very resource consuming procedure however, using functional dependencies; ideas from Cormen et al. (1990)[7] can be used to greedily identify or suggest candidates for a group using the following steps:

i. List the tokens or phrases beginning with those having higher RFDs;

ii. Sort tokens or phrases having common prefixes and repeat the same procedure for those with the same suffixes;

iii. Group the sorted tokens or phrases according to their prefixes, suffixes, or some identified patterns. Zipf's law can also be used to identify tokens/phrases for grouping.

iv. With expert knowledge from the SME, set a lower RFD outliers below which tokens or phrases can be discarded.

It should be noted that to discard tokens or phrases from being part of a group, the SMEs must be consulted since some of these tokens or phrases with lower RFDs may have significant information regarding the over all view of the data structures.

Another idea to note is that phrases can either be group by their prefixes or suffixes. These groupings aid in identification of other types of relationships in the dataset.

Consider the table below taken from some sample profiles:

| Combination of tokens | Token(s) | Phrase frequency | FD on Suffix % | FD on Prefix % |
|---|---|---|---|---|
| Single | SKRU | 1067 | --- | --- |
| | SEKSKANT | 633 | --- | --- |
| | SLOTTED | 340 | --- | --- |
| | NUT | 329 | --- | --- |
| | PHILIPS | 174 | --- | --- |
| | HODE | 101 | --- | --- |
| | SOCKET | 90 | --- | --- |
| Pairs | SEKSKANT SKRU | 254 | 40.1 | 23.8 |
| | SEKSKANT NUT | 171 | 20.9 | 6.7 |
| | SLOTTED SKRU | 71 | 19 | 3.1 |
| | PHILIPS SKRU | 33 | 27 | 52 |
| | HODE SKRU | 15 | 15 | 1.4 |
| | SOCKET SKRU | 4 | 4 | 0.4 |

Table 10: Identifying group relationships in phrases.

Identification of string relationships are based on certain observed criterion decided upon by the profilers. These criterion could include string patterns, the use of Zipf's law (grouping related tokens once they have been ranked in the order of their distinct occurrences in the dataset), RFDs, etc.

In Table 10 above, the following patterns of grouping can be identified: left group (prefix) and right group (suffix) dependencies, and also Zipf's law in identifying phrases.

The phrases in Table 11 below have some dependencies on SKRU. With the acquired domain knowledge, it was deduced that the phrases in the column Phrase 1, are types of screw head shape.

Knowing that the various phrases in the column Phrase 1 are kind of or subtypes of head shape from the domain knowledge, it can be deduced that those phrases are hyponym of the head shape and head shape is a hypernym of phrases in Phrase 1 column.

This grouping of the phrases has enable the identification of hyponym and hypernym relationships in phrases. However, there is still need to identify phrases that constitute good candidates for a valid group.

The table below was extracted from Table 10 and used to illustrate the discussion of Group Membership identification.

| Phrase 1 | Phrase 1 on | RFD on prefix | RFD on suffix |
|---|---|---|---|
| SEKSKANT | SKRU | 40.1 | 23.8 |
| SLOTTED | | 20.9 | 6.7 |
| PHILIPS | | 19 | 3.1 |
| HODE | | 15 | 1.4 |
| SOCKET | | 4 | 0.4 |

Table 11: Identification of phrases dependencies on a single phrase.

Suppose that the profilers with the help of SMEs set their lower outliers of RFDs on prefixes at 15%. On close inspection of Table 11, the following phrases can be identified to form a meaningful group; meaningful in the sense that their combined RFDs can provide a general view of the dataset under investigation.

The identified members are : SEKSKANT, SLOTTED, PHILIPS, and HODE; with a combine RFDs of 95% on SKRU. The type of relationships identified for this grouping was a hyponym as a subtype of hypernym. The hypernym suggested was the head shape. This idea of grouping and suggesting relationships, aids in reducing the dependencies from many phrases to a group and a single token.

In respect to the above discussion, the dependency is reduced to head shape; representing all prefixes in the dataset and SKRU as a single token in the same dataset. The RFD of head shape on SKRU is 95% and RFD of SKRU on head shape is 35%.

It can also be deduce that the hypernym (head shape) has asymmetric dependencies with SKRU.

Applying the idea of grouping prefix phrases above to those of suffixes, the following phrases from Table 10 would be suggested for a suffix (right) group:

Members {SKRU and NUT}. This group has an RFD of 30.5% on SEKSKANT while the dependency of SEKSKANT on this group is 61%. It can further be observed that the occurrence of SEKSKANT is well distributed within the dataset.

From the discussion of Formulation of Terminologies in Section 4.2.3.2.d, it can be deduced that the dependencies is symmetric and that the token SEKSKANT is a substring descriptor and an identifier for tokens/phrases in the dataset. This deduction can also reveal that there is a likelihood of suffix dependencies revealing substring descriptors and identifiers in a dataset.

Many tokens/phrases could appear in several smaller groups; the solution is to form a more generic groups in such scenarios by comparing the token/phrase that is depending on the group (prefix dependency) or the token/phrase that the group depends on (suffix dependency) and matching them.

These activities (comparison and matching of tokens/phrases) are clearly shown in cases where the tokens/phrases are synonyms or abbreviations.

Group dependency increases the confidence of standardisation due to the fact that the dependencies of individual members have been combined into a group dependency.

Grouping related elements with certain kinds of relationships such as on their properties significantly eliminate different representation of the same element in the dataset; a critical contribution of data profiling in the standardisation and verification of the data.

# Chapter 5

## 5.0 Interpretation and Evaluation of the Results

## 5.1 Result Interpretation

When performing data profiling, the profilers' interest is to find the underlying structures and the hidden relationships in a dataset. These involve separating strings into different fields such that they are transformed into a relational database in 1NF (a focus in this research).

The separation of strings into several fields presents a lot of challenges since a string like HEXAGON SCREW M10X2 8.8 DIN933 FOR PLATE MOUNTING has several pieces of information contained in it. The question then would be: what kind of numerical information or knowledge is needed to perform such activities?.

Profilers should therefore be able to identify and categorize this information (prepositions, item names, substring connectors, dividers, descriptors, etc.) from other additional information with less significance to the string meaning. Below are some of the approaches used in the analysis.

### 5.1.1 Domain Knowledge

The basis of analysing such a string would be the acquisition of some domain knowledge and contextual information. These would include the type of domain under investigation, the selection of parts of the dataset for the tasks, what possible metrics to generate from the dataset, etc.

There is therefore a need to extend the analysis of strings to positions of tokens/phrases; the presence of some kind of tokens/phrases like prepositions in the strings; special attributes like screw standards ( attributes whose fields have a high descriptive power in the record), etc.

In the case of the screw domain, profilers would seek to know which groups of the substrings are valid formats for the size of the screws; valid standards of screws; groups of substrings that are common measures for steel quality, the different units of screw measurement, etc.

These are important properties of screws that help in identifying/suggesting item names, the existence of relationships types, and also grouping of tokens/phrases within a record/dataset.

### 5.1.2 Phrase Frequency

Phrase frequency of tokens/phrases are computed so as to investigate the importance of the tokens/phrases in the dataset (Shehata et al. 2007)[36]. This is done by organising distinct occurances or co-occurrences of the tokens/phrases in a desired order, for example, by ranking the tokens/phrases so as to use Zipf's law to extract meaningful information.

The investigation is also achieved by revealing the general string patterns and other attributes patterns within the dataset.

It is thus a common knowledge that those tokens/phrases with higher distinct occurrences or co-occurrences in the dataset are more important and contribute significant information to the string and overall dataset meanings. However, string delimiters and other stop words could also exhibit higher frequencies.

To further investigate the importance of the tokens/phrases in the dataset and identify the insignificant tokens/phrases, profilers need to consider other aspects of string relationships such as those discussed below.

### 5.1.3 RFDs and Dependency Relationships

The use of functional dependencies in this research had a lot advantages. It was easier to isolate tokens and phrases based on their RFDs, i.e., a token/phrase can confidently be isolated from the record of single or independent groups if profilers know that such a token/phrase has a suffix or a record-wise relationships.

RFDs enabled the identification of different types of relationships between phrases and tokens in the dataset thus improving the confidence of term identifications. These relationships included: prefix-wise (how a token is preceding another token in a string or substring), suffix-wise (opposite of prefix-wise), record-wise (how two phrases co-occur in the same record), and attribute-wise (how attributes from one field can determine attributes from another field in the same record).

The most common types of resources useful for relationship extraction reviewed by Sarawagi (2008)[37] are surface token (tokens around and in-between two entities); part of speech tags (marking up the words in a text as corresponding to a particular part of speech ); syntactic parse tree structure (words are grouped into prominent phrase types); and dependency graph (words are linked to those they depend on).

In this thesis, surface tokens are the substring dividers and connectors. Profilers can split strings and identify item names and relationships when they know the groups of tokens/phrases that are substring dividers  and substring connectors.

Tokens/phrases which were considered to be substring dividers can also be ignored or removed from strings when identifying relationships or ontologically correct names. When a token/phrase has a strong symmetric dependency, there is a high likelihood that such a term is an identifier.

Once profilers are able to classify some part of the input strings and categories them (grouping substrings descriptors, identifiers, and connectors for the strings), they do not only condense the data and identify the underlying structures but also obtain a lot of help from SMEs who would be dealing with precise and organised data.

Some of these relationships can be used together to identify terms and other relationships more accurately, for example, DIN930 is related to HEXAGON HEAD BOLT both as an attribute (informing us as to which standard the bolt is compliant with) and in the record.

If the profilers had already established that the token HEAD was a substring connector and that the dependency of HEAD on HEXAGON and BOLT is symmetric, then the deduction could be that the phrase HEXAGON HEAD BOLT is an item name with a standard of DIN930.

Kang and Lee (2005)[20] observed that words which have more relations with other words in a document are semantically more important. Examples of such words (tokens/phrases) could be SCREW/SKRUE, SEKSKANT SKRU, etc., that may have multiple relationships with other tokens/phrases; these mutilple relationships could provide evidence of their importance in the dataset/record.

Such tokens/phrases can have prefix-wise, record-wise, or suffix-wise relationships depending on how they are written or how they occur in the record.

### 5.1.4 Groups

Grouping tokens/phrases further aided in the identification of more relationships and reduction of storage resources since groups of tokens/phrases would be identified by a group name.

There are some tokens/phrases like SEKSKANT, which have asymmetric dependencies with other tokens/phrases in the dataset. Asymmetric dependencies could also have some relation to an object property, for example, SEKSKANT is a description of a shape that is applicable to many different objects.

With such a background knowledge and some supporting evidence from phrase frequency, RFDs; it can be deduced that tokens/phrases such as SEKSKANT are descriptive elements.

Identification of abbreviation in a dataset can also be obtained by comparing and merging together groups of tokens/phrases which have similar members. The merger leads to the formation of a more generic grouping of tokens/phrases. For example, several tokens/phrases of SCREW HEAD TYPES occurring in many other groups can be combined into a generic group of SCREW SHAPES.

The whole idea of grouping tokens/phrases together as observed by Strzalkowski, is to derive meaningful information from those tokens/phrases that have lower RFDs and could be considered insignificant individually to the general meaning of a string.

When profilers group tokens/phrases; their idea of grouping the tokens/phrases are not only based on strings or substring relationships but also on the patterns of the strings, how they are co-written, frequencies in the dataset. This significantly increases the probability of identifying an item name, abbreviation, synonyms, etc., with a much higher degree of confidence.

For example, when the profilers know that the phrase SEKSKANT SKRU occurs 450 times in a dataset; the token SEKSKANT is a substring descriptor; and that the dependency relationship shown by the phrase is symmetric (prefix and suffix-wise), they can provide these facts to be used for the correct extraction of phrases as terms by ETL developers in the standardization and verification tasks.

### 5.1.5 Patterns

The replacement of some parts of the input strings by patterns is quite effective in aiding the analysis as shown in Section 4.2.3.2.c (Identification of Synonyms in Datasets).

The most significant use of patterns is in identification of tokens/phrases that would otherwise be statistically ambiguous. This is achieved by revealing commonalities between individual tokens/phrases (what are the patterns of the elements), identifying relationships such as pattern dependency, and occurrences of distinct string patterns in the dataset/records.

For example, if the string HEXAGON SCREW M10X2 DIN933 was translated to some pattern of <generic type> SCREW <MDXL> DIN933; where <generic type> would be the different shapes such as screw head and <MDXL> would represent the various measures of screw sizes.

The profilers would have reduced the dependency analysis to tokens such as SCREW and DIN933 due to the commonalities identified in the string patterns of <generic type> and <MDXL>.

## 5.2 Result Evaluation

From the discussion of Evaluation Strategy in Chapter 3 Section 3.5.1; evaluation of data profiling process can be divided into: tokens/phrases confidence level and the probability of occurrences of the tokens/phrases, and evaluation using design methods, i.e., functional testing and descriptive methods.

### 5.2.1 Confidence level and probability of tokens/phrases

When tokenization (generation of tokens/phrases, tuple combinations, and their respective frequencies) is performed and RFDs computed; the confidence level of these figures needs to be determined.

RFDs and confidence level should help profilers determine the likelihood that a selected sequence of tokens/phrases have some meaning in the dataset. The level of confidence in this context is therefore the likelihood that the RFDs found in a sample of the input dataset are correct given some margin of error.

The likelihood for the tokens/phrases could be computed using confidence interval or discussed using the rule of thumb, presented in Section 5.2.2.2. below. Confidence interval provides an estimated range of values within which an identified limit may be included.

Confidence level is therefore the probability value linked with the provided range of values, i.e., confidence interval.

Since the suggested or identified groups were to be used in the data standardization and verification; determining the group confidence (likelihood that the group formed is valid) and membership confidence (the likelihood that a token/phrase is a valid member of a group) together provide the likelihood that a selected sequence of tokens/phrases have some meaning in the dataset.

It should also be noted that data profiling can suggest or identify a group, but not necessarily confirm the group and its confidence.

The computation of RFDs can also be interpreted as a probability of the occurrences of elements.

For example, the RFD of SEKSKANT on SKRU from the pair SEKSKANT SKRU was computed as the number of co-occurrence of the phrase SEKSKANT SKRU divided by the occurrence SEKSKANT. This can also be computed as the conditional probability of the token being a SKRU given that it is SEKSKANT token, i.e., P(SKRU|SEKSKANT) = P(#SEKSKANT SKRU)/ P(#SEKSKANT) and RFD of SKRU on SEKSKANT is the conditional probability that a token is SEKSKANT given that it is SKRU, i.e., P(SEKSKANT|SKRU) = P(#SEKSKANT SKRU)/ P(#SKRU); where:

- #SKRU is the number of occurrence of the token SKRU in the dataset,
- #SEKSKANT is the number of occurrence of the token SEKSKANT in the dataset
- #SEKSKANT SKRU is the number of co-occurrences of the pair SEKSKANT SKRU in the dataset.

## 5.2.2 Design Evaluation Methods

### 5.2.2.1 Functional Testing

The various findings such as: frequencies, the existence of different kinds of relationships (eg, RFDs, GRFDs), synonyms, abbreviations that were established in the dataset will have to be evaluated against the domain ontology and the SMEs will also have to verify their correctness.

The SMEs will also have to confirm group validity by way of inspection.

Since these findings are to be used by ETL developers, they will also provide some testing as to how complete and reliable were the contribution of such findings to their standardization and verification tasks.

### 5.2.2.2 Descriptive Methods

If by our findings, we deduce that a token/phrase such as SEKSKANT is a member of some group and that another token/phrase such as SKRU has an RFD of 85% to this group; then there is need to provide some level of certainty that the findings are indeed valid for the whole domain given that these findings were based on a sample input from the domain.

Profilers can use the principle of the rule of thumb (a way of estimation made according to some practical observation but not based on exact measurement) to show the level of confidence; given that the sample data input is quite large and representative of the dataset under investigation.

The rule of thumb arguments are from the fact that more evidence or statistical figures were collected from a larger volume of dataset (like a corpus) compared to a smaller amount of dataset.

To further illustrate the argument, consider a dataset of 100 records. Suppose that data profiling on such a record shows that the RFD of SEKSKANT on SKRU is 97%; this is quite a high RFD on a token given the implication of RFDs on token meanings.

It can also be noted that this 97% dependency of SEKSKANT on SKRU shows that the token

SEKSKANT is quite dominant and occurs more than half of the dataset.

The deduction here is that the RFD on the token SKRU is quite high; the record is small and quite limited to provide other statistical evidence. Therefore, the rule of thumb can not be applied to such dataset with smaller records given that fewer evidence would be collected them.

Since the records were small and not very representative enough; the derived statistics cannot be used as evidence for processing an entirely new dataset within the same domain, i.e., of fasteners.

A new domain could be that of nuts or bolts since screws was considered.

However, if the dataset had thousands or millions of records, statistics from such datasets can be quite representative since more evidence would be collected.

Larger corpora provide more evidence and are therefore considered representative of the dataset under investigation since larger volume of data, statistics, and terminologies that would be collected from them can be applicable to different datasets within the same domain.

The principle of the rule of thumb can be used to approximate the level of confidence on dataset with larger records to be relatively high since their samples are considered quite representative and statistically significant, i.e., SKRU as a token might have 1000 occurrences in such dataset.

When identifying group membership, setting lower RFDs limits involve many consultation with the SMEs and the need to accommodate many group members; this increases the confidence and the validity of the group and its members.

The use of stored/extracted knowledge about a domain can also give a good confidence when identifying a term or a relationship. This is due to the fact that the knowledge is based on the domain and other observation from data recording. For example, if the profilers know that:

- 6K SKRU is a commonly used term for SEKSKANT SKRU or HEXAGON SCREW
- M10X2 is a valid size format
- 8.8, 12.9 are common measures for steel quality used on SEKSKANT SKRU or HEXAGON SCREWS
- DIN933 is a valid standards measure of SEKSKANT SKRU or HEXAGON SCREW.

Then identification of string patterns and other relationships in the record/dataset would be made easier since the profilers would be using valid domain knowledge to quantify and define their findings to be used in data standardisation and verification.

# Chapter 6

## 6.0 Research Conclusions

This research was set out to produce results/metrics for data standardization and verification. Therefore, data profiling had to reveal meaningful structures from the records under investigation so that ETL developers can use the meaningful structures for the standardization and verification tasks.

It is quite important that the phrase 'meaningful structures' in the context of this research need to be explained. Mansuri and Sarawagi (2006)[16] observed that it was challenging to effectively exploit useful clues which are scattered in various ways across structured database and unstructured text records.

For data profiling to reveal 'meaningful structures' from structured databases and unstructured text records/dataset under investigation, the following tasks should be performed by the data profilers:

i. Input strings are organized into some easily recognizable structures or patterns;

ii. Input strings are broken down into tokens/ phrases;

iii. Tokens/phrases are organized in such a way that their occurrences in the dataset are reflected in a format that is easy and quick to analyse, i.e., listing the tokens/phrases from their highest to lowest occurrences in the dataset;

iv. Tokens/phrases are organized in such a way that their underlying structures, and inherent relationships between the different tokens/phrases are easily revealed;

v. The identified relationships, inherent structures, and the tokens/phrases organization should suggest meaningful terms and candidates for grouping in the dataset.

## 6.1 Research Summary and Result Outcome

Data profiling on a general basis was defined in Chapter 1, Section 1.2.3.1, as the process of revealing structures, patterns in the contents of data and any other information. This information should be helpful for ETL developer(s) to make the right modelling decisions and precautions in processing the data so that the results can be reliable.

This definition leads to the following questions which summarise and present the outcome of the research:

i. What figures and results were needed by the ETL developers?

ii. How did this research provide the figures and the results needed by the ETL developers?

iii. And what were other findings of this research that were also helpful to the developers?

When the ETL developers are measuring data accuracy, they need to know the frequencies, RFDs, and the GRFDs figures from the dataset under investigation.

The frequency figures were provided from the tokenization process: listing of different tokens/phrases and computing their occurrences in the dataset.

By listing the frequencies of the various tuple combinations; RFDs of pairs, triples, etc., were computed. The RFD figures were used in identifying token/phrases relationships. The relationships led to the identification of terms and more importantly, the suggestion of groups.

The computation of RFDs also helped in identifying tokens/phrases that had lower dependencies on other tokens/phrases but had significant meaning to the strings. Grouping such tokens/phrases given some strong similarities in their relationships increased the lower individual dependencies to much higher dependencies of groups.

The grouping of tokens/phrases was not only for increasing token/phrase dependencies but also significantly improved string meanings. The term used for such RFDs was group relative functional dependencies (GRFDs).

GRFDs further improved the identification of relationships, terms, and further grouping of tokens/phrases aided the formation of more generic groups, i.e., screw head type, screw drive type, screw thread type can be generalised as screw type, etc.

Grouping tokens/phrases were extended to those considered as string/substring delimiters like prepositions, which have less meaning to the string. Such tokens helped in the identification of other string/substring relationships in the records/dataset.

Integrating RFDs and contextual information like pattern usage, synonyms and abbreviations in the records/dataset were easily identified.

The profiles provided should also be able to display grouping of tokens/phrases which may not have been grouped by chance, but have some meaningful information for the SMEs/ETL developers.

This meaningful information could be found by showing some special string/substring relationships such as substring: dividers, identifiers, connectors, and descriptors.

Data profiling is therefore a complete process of data extraction and data analysis. The results of this process are the set of statistics, token/phrase relationships, and other findings in the dataset that provided an overall view of the data source's inherent properties and structures.

## 6.2 Research Recommendations and Way Forward

This research has developed some ideas for acquiring knowledge and some tacit information from both structured and unstructured records/dataset so as to translate their contents into a relational database; it is therefore important to automate some of these activities.

Manual inspection/scans of millions of records is less effective and quite time consuming, automating such as a process would save time and other production resources.

Ideas from this research such as identification of violations of the normal forms using patterns/contents combinations to enable disambiguation and classification of data could also be used/tested in other domains like medicine, finance, etc.

# References

**[1]** A. McCallum, “Information extraction: Distilling structured data from unstructured text,” *ACM Queue*, vol. 3, pp. 48–57, 2005.

**[2]** Alon Y. Halevy , “Answering Queries using Views”, A Survey The International J*ournal on Very Large Data Bases archive* Vol 10 , Issue 4, pages : 270 – 294, December 2001.

**[3]** Amjad Umar, George Karabatis, Linda Ness, Bruce Horowitz, and A. Elmagarmid, "Enterprise Data Quality: A Pragmatic Approach," J*ournal of Information Systems Frontiers, Kluwer, Dordrecht, The Netherlands*, Vol 1, No. 3, pp. 279-301 , October 1999 .

**[4]** Claire Cardie. Empirical methods in information extraction. *AI Magazine*, 18(4):65-79, 1997.

**[5]** Codd, E.F. “A relational model of data for large shared data banks”. *Communications of the ACM*, vol 18, no.6, pages 377-387, June 1970.

**[6]** Codd, E.F. “Further normalization of the data base relational model”. *In Data Base Systems, Courant Inst. Computer. Science. Symposia.* Series 6, R. Rustin, Ed., Prentice-Hall, pages. 33-64, 1972.

**[7]** Cormen, Leiserson, and Rivest. (1990).“Introduction to Algorithms” Chapter 16.

**[8]** Cowie, J., and Lehnert, W. “Information extraction”. In *Special natural language processing issue of the communications of the ACM* Vol. 39, pp. 80-91, 1996. New York, NY, USA.

**[9]** Daille Beatrice. “Study and implementation of combined techniques for automatic extraction of terminology”. *In proceeding of the 32nd Annual Meeting of the Association for Computational Linguistics*, 1994.

**[10]** Dey Lipika and Haque S. K. Mirajul. “Opinion mining from noisy text data”. *In Proceedings of the second workshop on Analytics for noisy unstructured text data*, page. 83–90, New York, NY, USA : ACM, 2008.

**[11]** E. Agichtein and L. Gravano, “Querying text databases for efficient information extraction,” *in ICDE*, 2003.

**[12]** Eckerson, W.W., “Data Quality and the Bottom Line: Achieving Business Success Through a Commitment to High Quality Data”, *The Data Warehousing Institute Report Series*, No.101, Chatsworth, USA, 2002.

**[13]** Erhard Rahm and Hong Hai Do. “Problems and Current Approaches”, *IEEE Data Engineering Bulletin,* 23(3), September, 2000.

**[14]** F. Popowich, “Using text mining and natural language processing for health care claims processing,” *SIGKDD Explorartion Newsletter,* vol. 7, pp. 59–66, 2005.

**[15]** Hevner, A.R., March, S.T., Park, J., and Ram, S. “Design Science in Information Systems Research”. *MIS Quaterly*, Vol. 28 No. 1, pp. 75-105,2004.

**[16]** Imran.R Mansuri and Sunita Sarawagi, “A system for integrating unstructured data into relational databases,” *in Proceedings of the 22nd IEEE International Conference on Data Engineering (ICDE)*, 2006.

**[17]** Institute of Medicine, “To Err is Human: Building a Safer Health System”. *National Academy Press, Washington D.C.*, USA.2000.

**[18]** Jae Kyeong Kim, Hee Seok Song, Tae Seoung Kim, Hyea Kyeong Kim, “Detecting the change of customer behaviour based on decision tree analysis”, *Expert Systems*, V.22 / No.4, 2005.

**[19]** Jurafsky, D. and Martin, J.H. “Speech and Language Processing. An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition”. Prentice Hall, 2000.

**[20]** Kang, Bo-Yeong and Sang.-Jo Lee . "Document indexing: a concept-based approach to term weight estimation." *Information Processing and Management: an International Journal* 41(5): 1065 – 1080, 2005.

**[21]** M. Bhide, A. Gupta, R. Gupta, P. Roy, M. K. Mohania, and Z. Ichhaporia, “Liptus: Associating structured and unstructured information in a banking environment,” *in SIGMOD Conference*, pp. 915–924, 2007.

**[22]** M. Jansche and S. P. Abney, "Information extraction from voicemail transcripts," *in EMNLP '02: Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing*, pp. 320–327, USA, Morristown, NJ: Association for Computational Linguistics, 2002.

**[23]** Mathew Michelson and Craig A. Knoblock, "Creating relational data from unstructured and ungrammatical data sources," *Journal of Artificial Intelligence Research (JAIR)*, vol. 31, pp. 543–590, 2008.

**[24]** March, S. T. & Smith, G. F."Design and natural science research on information technology". *Decision. Support System*., 15, 251-266, 1995.

**[25]** Naomi Dushay and D. Hillmann. "Analyzing metadata for effective use and re-use". In Dublin Core 2003. Seattle, Washington. http://dc2003.ischool.washington.edu/Archive-03/03dushay.pdf.

**[26]** National Information Standards Organization (NISO). "understanding metadata", 2004. http://www.niso.org/publications/press/UnderstandingMetadata.pdf.

**[27]** Pantel Patrick and Lin Dekang "Discovering word senses from text". *In Proceedings of ACM SIGKDD Conference on Knowledge Discovery and Data Minin*g, pgs 613–619, 2002.

**[28]** R. Ghani, K. Probst, Y. Liu, M. Krema, and A. Fano, "Text mining for product attribute extraction," *SIGKDD Explorations Newsletter*, vol. 8, pp. 41–48, 2006.

**[29]** R. Grishman, "Information extraction: Techniques and challenges," *in SCIE*, 1997.

**[30]** R. Grishman, S. Huttunen, and R. Yangarber, "Information extraction for enhanced access to disease outbreak reports," *Journal of Biomedical Informatics*, vol. 35, pp. 236–246, 2002.

**[31]** Rajman, M. and Besancon, R. "Text mining: natural language techniques and text mining applications". *Proceedings 7th IFIP 2.6 Working Conference on Database Semantics (DS-7)*, 1997.

**[32]** Raymond. J. Mooney and Razvan. C. Bunescu, "Mining knowledge from text using information extraction," *SIGKDD Explorations*, vol. 7, pp. 3–10, 2005.

**[33]** Redman, T.C. “Data Quality for the Information Age”. Artech House, Boston, USA, 1996.

**[34]** Redman, TC, “The Impact of Poor Data Quality on the Typical Enterprise”, *Communications of the ACM*, vol. 41, no. 2, pages 79-82, February, 1998.

**[35]** Rema Ananthanarayanan, Vijil Chenthamarakshan, Prasad M Deshpande, “Rule based synonyms for entity extraction from noisy text”, *In Proceedings of the second workshop on Analytics for noisy unstructured text data, ACM International Conference Proceeding Series*; Vol. 303, pages 31-38 , 2008.

**[36]** Shehata, S., Karray, F. and Kamel, M.,“A concept-based model for enhancing text categorization ”, 13th, ACM KDD, pp. 629-637, August, 2007.

**[37]** Sunita Sarawagi. (2008). “Information extraction" *FnT Databases*, 1(3), 2008.

**[38]** Tomek Strzalkowski. “Robust text processing in automated information retrieval”.*In Proceedings of the 4th Conference on Applied Natural Language Processing, Stuttgart*, pages 168-173, 1994.

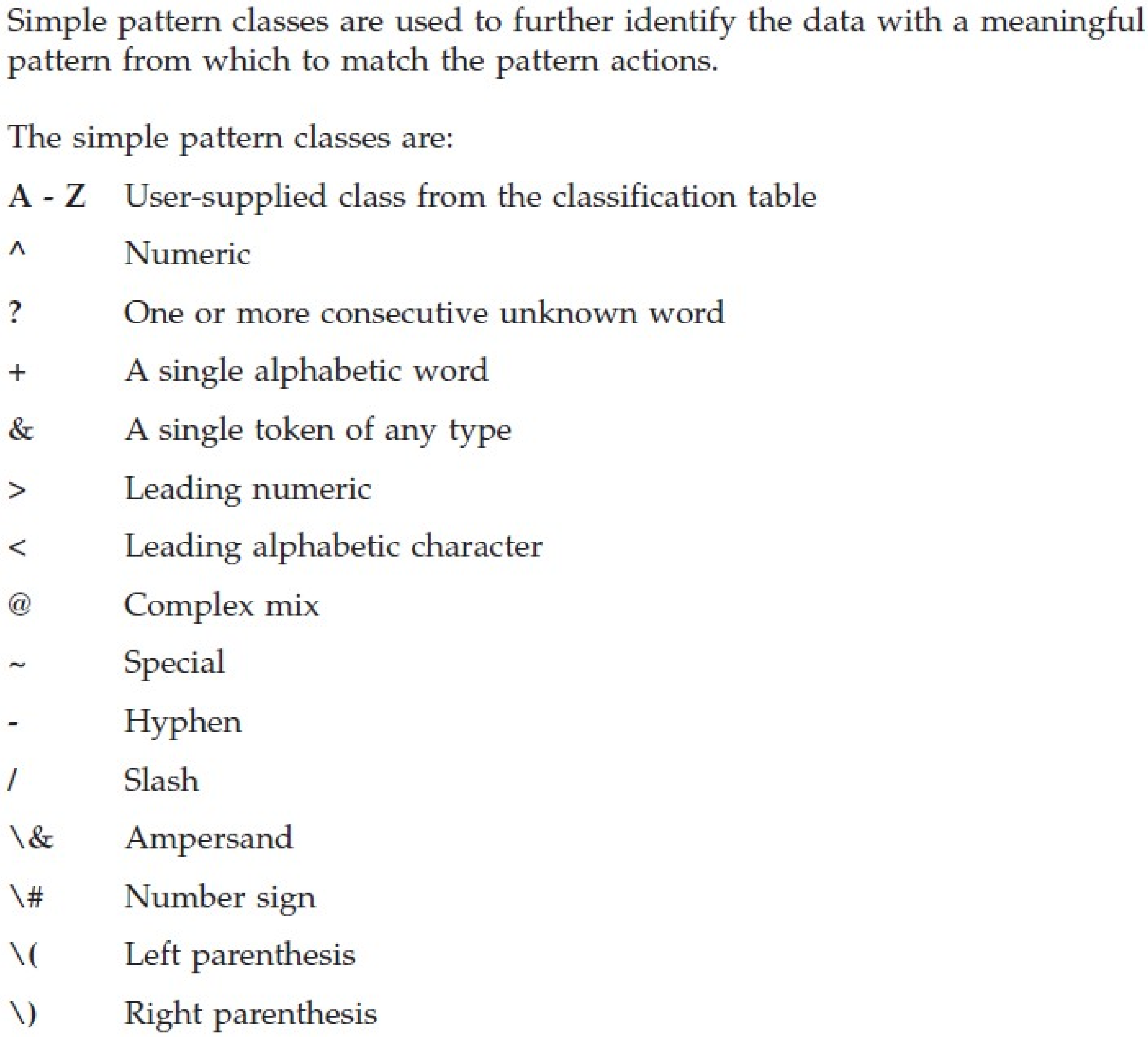
**[39]** Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth (1996). "From Data Mining to Knowledge Discover in Databases". Retrieved on 12.02.2008.

**[40]** Wang, Richard Y., Ziad and Lee, “Data Quality”, *Kluwer Academic Publishers*, 2000.

**[41]** Wentian Li. “Random Texts Exhibit Zipf's-Law-Like Word Frequency Distribution”. *IEEE Transactions on Information Theory*, 38(6). 1842- 1845, 1992.

# Appendix

## Appendix A. Simple pattern classes



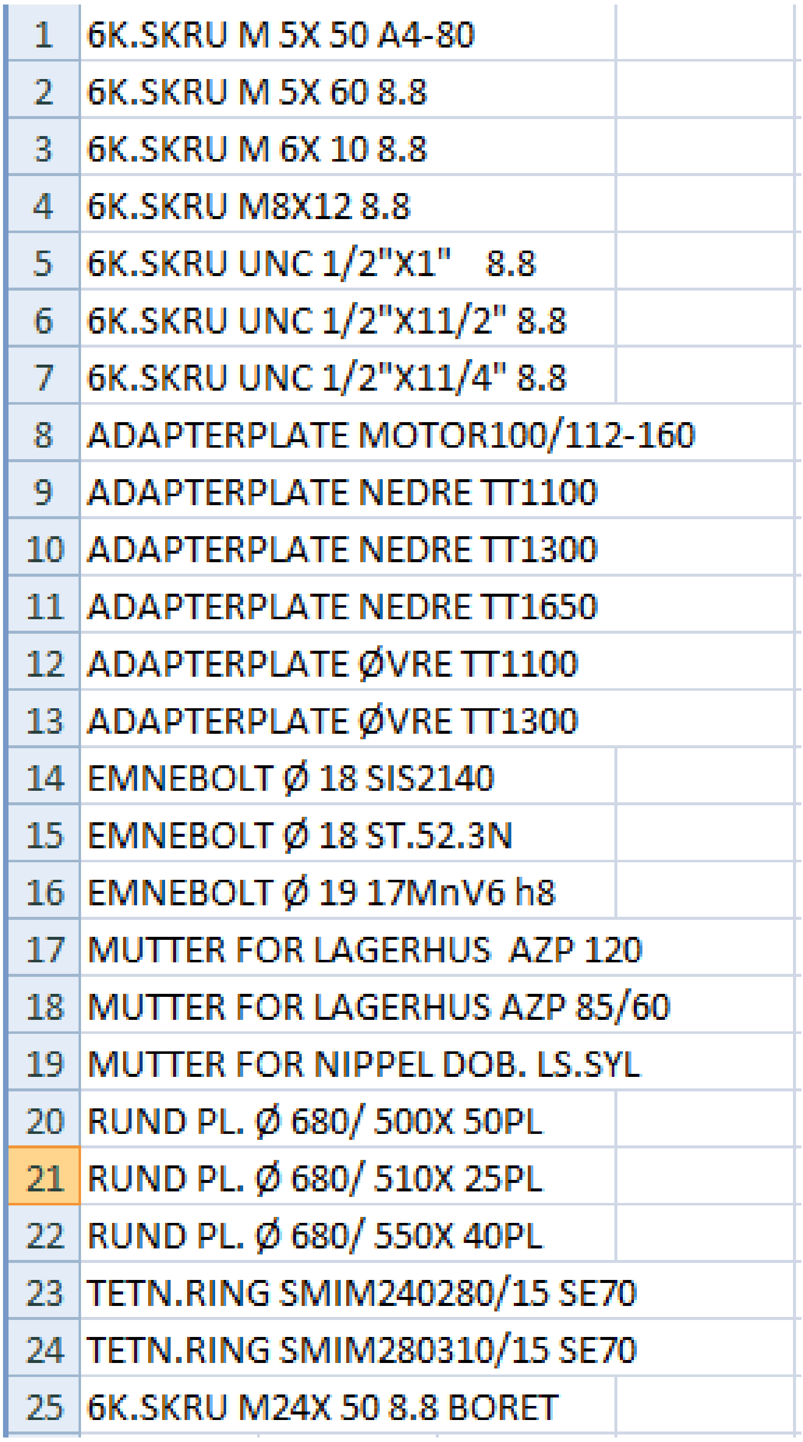
Simple pattern classes are used to further identify the data with a meaningful pattern from which to match the pattern actions.

The simple pattern classes are:

| | |
|---|---|
| **A - Z** | User-supplied class from the classification table |
| ^ | Numeric |
| ? | One or more consecutive unknown word |
| + | A single alphabetic word |
| & | A single token of any type |
| > | Leading numeric |
| < | Leading alphabetic character |
| @ | Complex mix |
| ~ | Special |
| - | Hyphen |
| / | Slash |
| \& | Ampersand |
| \# | Number sign |
| \( | Left parenthesis |
| \) | Right parenthesis |

Figure 4: Pattern class adapted from IBM Pattern Action Reference Guide

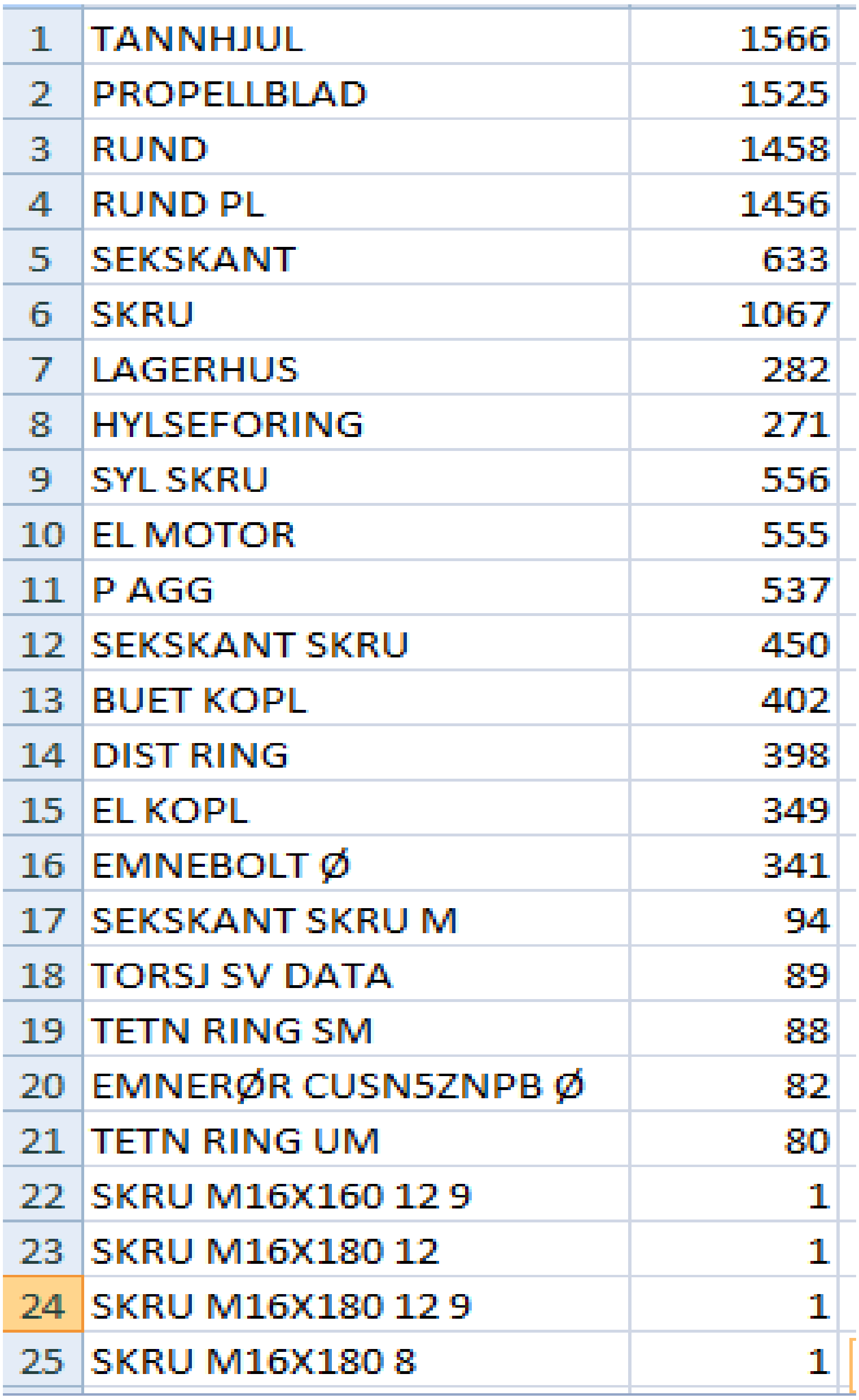
# Appendix B:  Sample input dataset/record.



| | |
|---|---|
| 1 | 6K.SKRU M 5X 50 A4-80 |
| 2 | 6K.SKRU M 5X 60 8.8 |
| 3 | 6K.SKRU M 6X 10 8.8 |
| 4 | 6K.SKRU M8X12 8.8 |
| 5 | 6K.SKRU UNC 1/2"X1"    8.8 |
| 6 | 6K.SKRU UNC 1/2"X11/2" 8.8 |
| 7 | 6K.SKRU UNC 1/2"X11/4" 8.8 |
| 8 | ADAPTERPLATE MOTOR100/112-160 |
| 9 | ADAPTERPLATE NEDRE TT1100 |
| 10 | ADAPTERPLATE NEDRE TT1300 |
| 11 | ADAPTERPLATE NEDRE TT1650 |
| 12 | ADAPTERPLATE ØVRE TT1100 |
| 13 | ADAPTERPLATE ØVRE TT1300 |
| 14 | EMNEBOLT Ø 18 SIS2140 |
| 15 | EMNEBOLT Ø 18 ST.52.3N |
| 16 | EMNEBOLT Ø 19 17MnV6 h8 |
| 17 | MUTTER FOR LAGERHUS  AZP 120 |
| 18 | MUTTER FOR LAGERHUS AZP 85/60 |
| 19 | MUTTER FOR NIPPEL DOB. LS.SYL |
| 20 | RUND PL. Ø 680/ 500X 50PL |
| 21 | RUND PL. Ø 680/ 510X 25PL |
| 22 | RUND PL. Ø 680/ 550X 40PL |
| 23 | TETN.RING SMIM240280/15 SE70 |
| 24 | TETN.RING SMIM280310/15 SE70 |
| 25 | 6K.SKRU M24X 50 8.8 BORET |

Figure 5: Sample input dataset/record

## Appendix C:  Sample report of the tokenization process.



| | | |
|---|---|---|
| 1 | TANNHJUL | 1566 |
| 2 | PROPELLBLAD | 1525 |
| 3 | RUND | 1458 |
| 4 | RUND PL | 1456 |
| 5 | SEKSKANT | 633 |
| 6 | SKRU | 1067 |
| 7 | LAGERHUS | 282 |
| 8 | HYLSEFORING | 271 |
| 9 | SYL SKRU | 556 |
| 10 | EL MOTOR | 555 |
| 11 | P AGG | 537 |
| 12 | SEKSKANT SKRU | 450 |
| 13 | BUET KOPL | 402 |
| 14 | DIST RING | 398 |
| 15 | EL KOPL | 349 |
| 16 | EMNEBOLT Ø | 341 |
| 17 | SEKSKANT SKRU M | 94 |
| 18 | TORSJ SV DATA | 89 |
| 19 | TETN RING SM | 88 |
| 20 | EMNERØR CUSN5ZNPB Ø | 82 |
| 21 | TETN RING UM | 80 |
| 22 | SKRU M16X160 12 9 | 1 |
| 23 | SKRU M16X180 12 | 1 |
| 24 | SKRU M16X180 12 9 | 1 |
| 25 | SKRU M16X180 8 | 1 |

Figure 6: Sample report of the tokenization process.