

Investigation into the application of Model Transformation to Data Transformation

by

SRINIVAS BALASUBRAMANIAN

This project is submitted to the Gannon University graduate faculty in partial fulfillment for the degree Master of Science in Computer and Information Science.

Option: INFORMATION ANALYTICS

Approved:

Dr. Joshua C Nwokeji

Dr. Barry Brinkman

Print professor's name, Ph.D.

Advising Professor in Charge of Research

Print committee member name, degree

Committee Member

Dr. Sreela Sasi

Dr. Barry Brinkman

Print committee member name, degree

Committee Member

Print CIS Chair name and degrees

Chair, Computer and Information Science
Department

Gannon University
Erie, Pennsylvania 16541

April 2017

Acknowledgements

First, I thank god for showering his blessings on me and strengthening me on this project as a successful one.

I am grateful to **Dr. Brinkman Barry J** Chair, Computer and Information Science Department Gannon University, Erie PA, for permitting me to undertake this project.

I am thankful to **Dr. Nwokeji Joshua C** my advisor, without his help this project not have been a good one. I thank him for having instilled enthusiasm and for being an unfailing source of inspiration and constant encouragement for me.

I am thankful to my mates **Abhishek Kotian & Apoorva Anugu** for sharing their knowledge about this project and helping me in completing this project successfully.

I wish to express my sincere thanks to all my Professors for their help and guidance.

Also, I would like to thank my parents, almighty and friends who have contributed for the successful completion of this project.

Table of Contents

Acknowledgements	2
Abstract.....	3
List of Figures.....	5
List of Tables	6
1. Introduction.....	7
1.1 Overview	7
2. Review Related Work.....	8
2.1 General background, definition of terms	9
2.2 Review Objective.....	9
3. Methodology	19
4. Results	20
5. Conclusion and Future Work.....	33
6. References.....	34

Abstract

Business organizations are generating and using data for day to day transactions, some of these data are generated in an unstructured format which can be difficult to use [1]. Unstructured data refers to raw information that is not in a precise schema or structure [2]. Some of the examples of unstructured data are medical reports such as X-ray, scan, ECG report [3] and others include like call logs, E-mail reports, audio, video, images etc. [4]. One major challenge with unstructured data is difficult to store in a conventional database and process [1].

However, to manage process or extract useful business intelligence and knowledge from unstructured data, they need to be converted to a structured format using suitable techniques. Structured data are data that conforms to a common or specific format or schema [5]. Most common examples for structured data are a person's name, age, address, gender and phone number [6].

Available techniques such as pattern matching [2] and text analytics [7] for converting unstructured data to structured data have some drawbacks and limitations, and the result have not been very effective [2, 7, 8]. For instance, the main limitation of the pattern matching is low flexibility that is the match should be exact to the data that is to be transformed [2]. The limitation in text analytics extracting text from the language used [7].

Model to Model transformation (M2M) can provide a better alternative and thus to help overcome the drawbacks of current techniques for formatting unstructured data. In a typical M2M, a source model is transformed into target model using well defined transformation rules [9]. M2M offers many advantages such as reducing complexity this can be extended to reduce the complexity involved in formatting unstructured data. Another benefit of M2M is easy to use [10] which will be helpful when dealing with the unstructured data transformation.

The aim of this research is to investigate if M2M will be suitable for converting unstructured data to structured data.

List of Figures

Figure 1 – Overview of Model to Model transformation.....	19
Figure 2 - Source Metamodel.....	20
Figure 3 - Target Metamodel.....	21
Figure 4 - Source to Target Transformation.....	26
Figure 5 - Source – A simple table Example.....	27
Figure 6 - Target – Corresponding MS Office Excel Worksheet.....	28
Figure 7 - SpreadhseetML Simplified Metamodel.....	29
Figure 8 - Table Metamodel.....	30
Figure 9 - Simple XML Metamodel.....	30

List of Tables

Table 1 Limitations of Existing techniques..... 10

Table 2 Limitations of the existing techniques versus Benefits of proposed technique..... 17

1. Introduction

1.1 Overview

This research aims to do an investigation on the model driven engineering technique to check whether it is possible to use them in converting the unstructured data to structured data and saving to our databases.

Unstructured Data refers to raw information that is not in a precise schema or structure. Example for unstructured data are medical reports such as X-ray, Scan, ECG report and others like audio, video, images, call log, etc. Structured Data are data that conforms to a common or specific format or schema. Example for structured data are a person's name, age, address, gender and phone number.

The compilation time for the unstructured data is more as it is not in particular schema and it is an energy consuming task. The amount of data generated in an enterprise can be costly in terms of storage and it's so difficult to extract useful business intelligence and knowledge from the unstructured data.

Model Driven Engineering or Model Driven Development is a software engineering approach to raise the level of abstraction of developers who create a software with the goal of simplifying the process and tasks that are involved in the software life cycle [41, 35]. Model transformation is a process to convert one model to another model [43].

Benefits of Model to Model Transformation

1. Model transformation can be used for creating, filtering, and modifying models [34] and can be used in the different phases of the development of software life cycle [34]. Its portable i.e. it can work on various platforms like network of computers, middleware etc. [35, 36]. It also helps in reducing complexity like bridging the gap between the abstraction and the implementation. Model Transformation can be used with the information what is captured at the first instance [34,35,36].
2. Model Driven Development helps in achieving the goals very easy like transforming the source model to the desired target model also helps in standardizing the process from which the automation would be very easy. As a software engineering approach helps to define the architecture that will be followed to build a product [37].
3. Model Driven Development helps a company to increase the return on investment and to improve the productivity also helps the developers to increase the short term and long term productivity of the software product designed [38].

This research on “Investigation into the application of Model Transformation to Data Transformation” believes that model to model transformation can be used as an alternative to reduce the setbacks in the existing methods. However, a little consideration has been given to Model to Model transformation technique.

2. Review Related Work

In the review of the Model to Model transformations, some research papers focus on model to model transformation and model to data transformation using the model transformation language. Atlas Transformation Language (ATL) is a model transformation language which transforms source model to the target model. The research paper [39] explains about the use of ATL for transforming model from one format to another. In this paper a list of family names has been transformed to another list of names where the family names have been separated by their first name, last name and classifies whether they are male or female.

The research paper [40] illustrated a simple example of representing a set of books in the library has been transformed to a model, in this the books in the library are considered as the system and it has been represented as a model. The author of this paper also illustrated an example on converting a static chart from a Java program to represent it in an Excel spreadsheet or in an HTML Table using a transformation language ATL which was also actually contributed by Eclipse/GMT. The static chart has been taken as a source model and it has been transformed to the target model of spreadsheet.

The ATL transformation language plays a major role in model to model transformations, there are lot of examples available in converting one model to another model. Some of the most common transformation examples are transforming Circle to Squares and Table to Microsoft Excel Workbook.

The research paper [8] discussed about the conversion of unstructured data to structured data using text analytics. Text analytics or text mining is used to retrieve a valuable information from the unstructured data. The paper explained about the framework and the process of extracting valuable information from the unstructured text. The paper also discussed about the limitations of the text analytics, the major limitations of using text analytics are lot of external software programming must be involved for extracting textual sources from various sources and managing the master data extracted is a major task.

The research paper discusses about the frameworks and techniques available for converting unstructured data to structured data using Big data analytics in Health Care, their advantages and the limitations. Some

of the major limitations of big data analytics are ease of use, security and privacy. The lag between the data collection and the processing are still to be addressed.

2.1 General background, definition of terms, and other relevant background information related to research review

Model Driven Architecture – Model Driven architecture provides a framework for a software system to be built [42].

Model Transformation Languages - Model transformation languages are well defined to perform model transformation. Some of the model transformation languages are ATL, QVT, SmartQVT, ETL etc. [43].

ATL is known as Atlas Transformation Language. It is used in model to model transformation.

QVT is known as Query/View/Transformation and it is a standardized language for model transformation.

2.2 Review Objective

The main objective of this research review is to check whether the model driven engineering techniques can be applied for the conversion of unstructured data to structured data and saving this format of information in the system or database.

2.3

Table 1: Limitations of Existing Techniques

Sentiment Analysis	Predicting the mood of the sentence like whether it is positive or negative [1].	<p>“Product XYZ is good but expensive”</p> <p>The above statement states two aspects of the product XYZ where “product is good” shows the positive or favorable statement and “product XYZ is expensive” shows the negative or unfavorable statement [3].</p>	<p>Context – A decision cannot be made based on the words that are used in the context, as there will be two different meaning for the same word [5].</p> <p>Regional Variations – Language used in the context is a major limitation as different language words have different meaning [5].</p>
Optical Character Recognition in Natural Language Processing	Extracting useful information from the given image [1].	<p>Optical Character Recognition is used in Banking sectors to process the checks, just a scan of the check will process the transaction without any human involvement and it is also used in other industries like finance, education and legal industries to digitize the records or documents [4].</p> <p>OCR it is also used for the automatic number plate reading and also used as an aid for blind people [2].</p> <p>OCR is especially used to extract information from the unstructured data as the data can be in any format like image, text, graphics etc.</p> <p>In OCR the text has been scanned, preprocessed, segmented and data has</p>	<p>The main limitation in extracting information is because the data is often mixed with text and graphics [2].</p> <p>Variations in style and shape of the data [2].</p> <p>Variations because of subscripts and superscripts in the data [2].</p>

		been extracted.	
Data Mining in Information Retrieval	Process of analyzing and extracting useful information from different perspectives by using various data mining concepts such as clustering, classification etc. [6].	<p>Data mining is applied in Healthcare Industry for the evaluation of treatment effectiveness, by comparing the causes, symptoms, and course of treatments [7].</p> <p>Data mining is basically retrieving information from the different types of data. The data will be in any format, in healthcare industry the data can be in any format such as medical prescription, X-ray or scan reports. All these are analyzed and valuable information is extracted.</p>	Issues may arise with the missing, corrupted, inconsistent data as the information recorded will be in different format from different sources [7].
Text Analytics	Text analytics helps in retrieving the valuable text information from the unstructured and semi-structured data. Text analytics is also referred as text mining [9].	Text analytics or text mining is used in many fields like Publishing and media, Banks, Telecommunications etc. [11].	<p>Lot of software programming is needed to extract textual information from different sources [9].</p> <p>Managing the unstructured data from various sources is complicated [9].</p>
Big Data Analytics	Big data refers to large or complex data that are difficult to manage with the traditional software, hardware and data management tools [10].	Big data analytics is used in many industries. For instance, in the health care industry it is used to detect the disease at the earlier stage where they can be treated more easily and effectively [10].	<p>Data extraction and cleaning is one of the limitation in big data analytics.</p> <p>Data Integration is another limitation as the data will be coming from different databases or web portals.</p>
Part-of-speech tagging	Tagging the part of speech for each word in a sentence such as noun, verb, adverb, etc. [1].	For example, “a cotton shirt” this words are tagged as Nouns [16].	The main challenge in the POS is removing ambiguities, for instance “Flies like a flower” here the POS for the words in a

		<p>Part of speech tagging is used in speech recognition and information retrieval [15].</p> <p>In Speech recognition POS helps in recognize the pronunciation. It also helps in preprocessing.</p>	<p>sentence are</p> <p>Flies: noun or verb?</p> <p>like: preposition, adverb, conjunction, noun, or verb?</p> <p>a: article, noun, or preposition?</p> <p>flower: noun or verb?¹</p>
Web Crawling in Information Retrieval	Automatic Script that can download the contents from the World Wide Web automatically [6].	Web crawling in Information Retrieval used to retrieve information from various sources and this web crawler is one of the important component in search engines which will help us to extract valuable information from the web pages [6].	<p>Web crawling is very difficult because of its large volume and its rate of change [14].</p> <p>Large volume states that it can download only certain limit of web pages at a time.</p> <p>Rate of change implies that the web page either might be added or deleted as there will be lot of changes in the web pages on daily basis.</p>
Text Summarization in NLP	Summarizing the whole content of the text document in a shorter version without changing the overall meaning of the document [11].	<p>It is helpful in summarizing single and multi-document by extractive or abstractive text summarization method [13].</p> <p>The documents from various sources will have lengthy and unstructured text, this text summarization helps in converting and extracting useful information or structured text using two methods like abstractive</p>	Summarizing the whole content of the document from various sources in a right way (language, format, etc.) to a specific user [11].

¹ <https://www.cs.umd.edu/~nau/cmsc421/part-of-speech-tagging.pdf>

		and extractive.	
Stemming in Natural Language Processing	<p>Stemming is a process of deriving a word to their root [17].</p> <p>For instance, the word decided(Adjective), decision(Noun) is basically derived from the word “decide” so here the word decide is a root word or stem word [17].</p>	<p>Stemming is an important feature in today’s search systems. The main idea of this stemming process is reducing the word to their root while searching [18].</p> <p>Stem word usually represents broader concept than the original term which will be helpful in retrieving large number of documents [18].</p>	<p>Over stemming is where the two different words are stemmed to the same root [18].</p> <p>Under stemming is where the two different words should be stemmed to the same root but not [18].</p>
Boolean Model in Traditional Information Retrieval	<p>Boolean model in information retrieval is based on the Boolean Algebra. It represents the model by set of index terms which are viewed as Boolean variable and valued as True if it is present in document [19].</p>	<p>For instance, consider we have three documents and they are [20]</p> <p>Doc1: Information Retrieval has 2 models and Information.</p> <p>Doc2: Boolean is a basic Information Retrieval classic model.</p> <p>Doc3: Information is a data that processed, Information.</p> <p>If the Query for above documents be</p> <p>$(Data \wedge Information) \vee (\sim Retrieval)$</p> <p>Result: Data: Doc3 Information: Doc1, Doc2, Doc 3 Retrieval: Doc1, Doc2 [20]</p>	<p>Retrieval performance is very poor [19].</p>

Probabilistic Model in Traditional Information Retrieval	Probabilistic model attempts to estimate the probability of the user finding a document. The ranking in this model is based on the documents retrieved to the given query [20].	The information is extracted based on the query which has been passed by the user. Based on the query the best results are listed out first [20].	Probabilistic models are very hard to build and program.
Audio Analytics in Big data analytics	Audio analytics analyze and extract information from unstructured audio data. This is also referred to as speech analytics as this technique is used in speech recognition as well [21].	Audio analytics are used in customer call centers which will help to improve customer experience, enhance sales turnover rates, etc. [21]	It is difficult to have 100% accuracy in the identification of audio stream; it may not be able to handle accented words. Sometimes understanding the contextual meaning of the words would still be a challenge. ²
Video Analytics in Big data analytics	Video analytics analyze and extract meaningful information from video stream [21].	Video analytics is mostly used in security and surveillance systems. Video analytics can easily detect breaching, loitering, theft etc. [21].	There are three approaches to analyze video: <ul style="list-style-type: none"> • Server-based (the videos which are captured through cameras are routed back to a centralized server to perform analysis) • Edge based (the video analysis is performed locally or on raw data captured by the cameras) • Agent Vi's distributed architecture (in this the analysis work is distributed between the edge device and the server).

			The accuracy of server-based analysis is less as the data is generated in compressed form due to limited bandwidth, but it facilitates easier maintenance while in edge based the entire content is available for analysis give better results but it is costly to maintain and have lower processing power as compared to server-based analysis [22].
Social media analytics	<p>Analyzing structured and unstructured data from the social media channels is referred to as social media analytics.</p> <p>Social media are such as social networks like Facebook, Twitter and blogs like WordPress and other platforms like Instagram YouTube etc. [21].</p>	The main application of social media analytics is the marketing field where the companies can analyze the reaction of the people of their products [21, 22].	<p>Massive amount of data requires lot of storage space and processing power [23].</p> <p>Worldwide online accessibility provides more data in many languages [23].</p>
Predictive analytics	Predictive analytics is a method where the useful information is extracted from the existing data, basically it doesn't tell what will happen in the future instead it forecasts what might happen in the future [23].	<p>Predictive analysis is used in Customer relationship management fields like marketing, sales, customer services etc. to analyze the product in demand and predict the customer buying habits [23].</p> <p>It is also used in clinical decision support to predict whether the patients are at risk of developing certain</p>	<p>Predictive analysis technique is based on statistical methods but more statistical methods must be developed for big data as they are massive [22].</p> <p>Statistical methods are methods of collecting, summarizing, analyzing, and interpreting variable numerical data.³</p>

³ <http://www.encyclopedia.com/computing/dictionaries-thesauruses-pictures-and-press-releases/statistical-methods>

		conditions like asthma, diabetes etc. [23].	
--	--	--	--

Table 2: Limitations of the existing techniques versus Benefits of proposed technique

Method	Limitations of the Existing Techniques	Benefits of the Proposed Technique
Part of Speech Tagging	<p>The main challenge in the POS is removing ambiguities, for instance “Flies like a flower” here the POS for the words in a sentence are</p> <p>Flies: noun or verb?</p> <p>like: preposition, adverb, conjunction, noun, or verb?</p> <p>a: article, noun, or preposition?</p> <p>flower: noun or verb?⁴</p>	<p>Model transformation helps in reducing complexity like bridging the gap between the abstraction and the implementation. [34, 36].</p> <p>In software development models play a major role, these models help developers to develop a technology or software to solve a problem.</p>
Optical Character Recognition	<p>The main limitation in extracting information is because the data is often mixed with text and graphics [44].</p> <p>Variations in style and shape of the data [44].</p> <p>Variations because of subscripts and superscripts in the data [44].</p>	<p>Model Transformation can be used with the information what is captured at the first instance [35].</p> <p>Model Transformation helps in creating a model from the information captured at the first instance instead of starting from the scratch.</p>
Big Data Analytics	<p>Data extraction and cleaning is one of the limitation in big data analytics.</p> <p>Data Integration is another limitation as the data will be coming from different databases or web portals.</p>	<p>Model Driven Development helps in standardizing the process from which the automation would be very easy [37].</p>
Web Crawling in Information Retrieval	<p>Web crawling is very difficult because of its large volume and its rate of change [48].</p> <p>Large volume states that it can download only certain limit of web pages in each time.</p> <p>Rate of change implies that the web page either might be added or deleted as there will be lot of changes in the web pages on daily basis.</p>	<p>Model Driven Development as a software engineering approach helps to define the architecture that will be followed to build a product [37].</p>

⁴<https://www.cs.umd.edu/~nau/cmsc421/part-of-speech-tagging.pdf>

Data Mining	Issues may arise with the missing, corrupted, inconsistent data as the information recorded will be in different format from different sources [45].	Model Transformation can be used with the information what is captured at the first instance [35]. Model Transformation helps in creating a model from the information captured at the first instance instead of starting from the scratch.
Text Analytics	Lot of software programming is needed to extract textual information from different sources [46]. Managing the unstructured data from various sources is complicated [46].	In MDA, a model to text transformation (M2T) is a transformation definition (set of transformation rules) that transforms an expressed Specific Model to target source code or documentation. M2T transformation tool allows writing transformation definitions, running transformations, and produce texts (target source code or documentation of a system) as outputs [47].
Probabilistic Model in Traditional Information Retrieval	Probabilistic models are very hard to build and program.	Model Driven Development helps in achieving the goals very easy like transforming the source model to the desired target model [37]. For instance, if the target model should be in specific format or type it can be easily achieved by giving specific instructions during the conversion.
Text Summarization	Summarizing the whole content of the document from various sources in a right way (language, format, etc.) to a specific user [44].	In MDA, a model to text transformation (M2T) is a transformation definition (set of transformation rules) that transforms an expressed Specific Model to target source code or documentation. M2T transformation tool allows writing transformation definitions, running transformations, and produce texts (target source code or documentation of a system) as outputs [47].

3. Methodology

The aim of this research is to investigate whether the Model to Model Transformation can be used for converting unstructured data to the structured data.

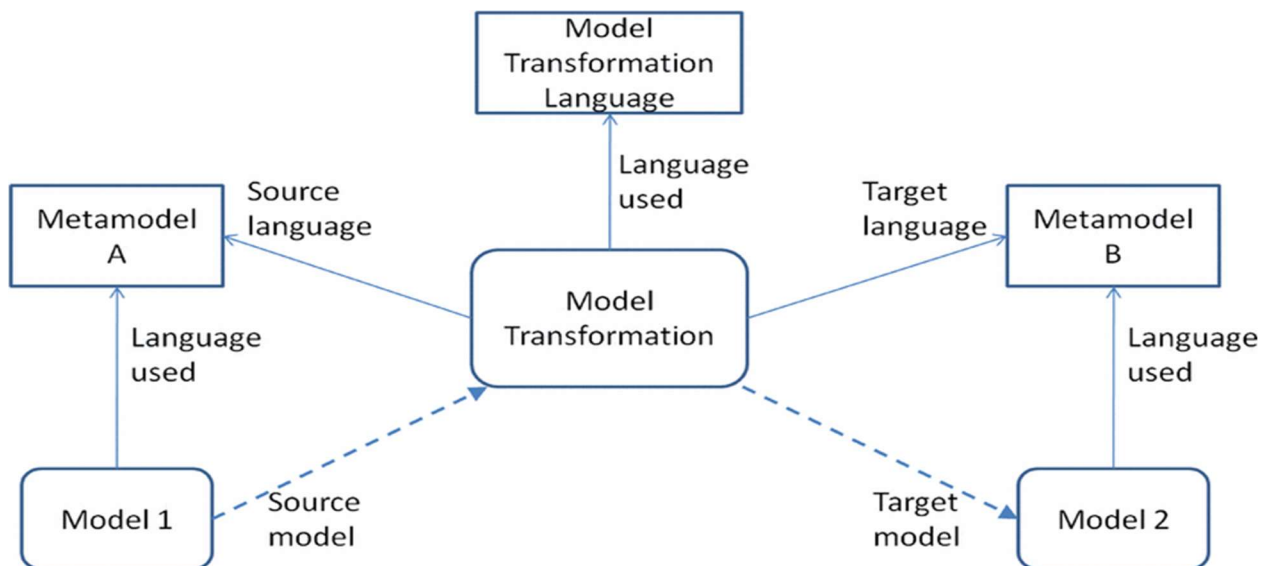
Model to Model Transformation is a process of converting one model to the other model. In this process the source model can be converted to the desired target model with the help of model transformation languages such as ATL, QVT etc.

Figure 1 shows the overview of model to model transformation. The Source model which is to be transformed should conform to the Source Metamodel and the Model Transformation Language is used to do the transformation. The target model should also conform to the target metamodel.

ATL is known as Atlas Transformation Language which is used for transforming models.

In this research Eclipse Modeling Framework, has been used to execute the Model to Model transformation. Eclipse modeling framework is highly helpful as the code is generated automatically for the metamodel drawn which helps in converting the source model to target model.

Figure 1 – Overview of Model to Model transformation



Source: Metamodeling and Model Transformations in Modeling and Simulation, Simulation Conference (WSC), Proceedings of the 2011 Winter 2011, 11-14 Dec, 2011

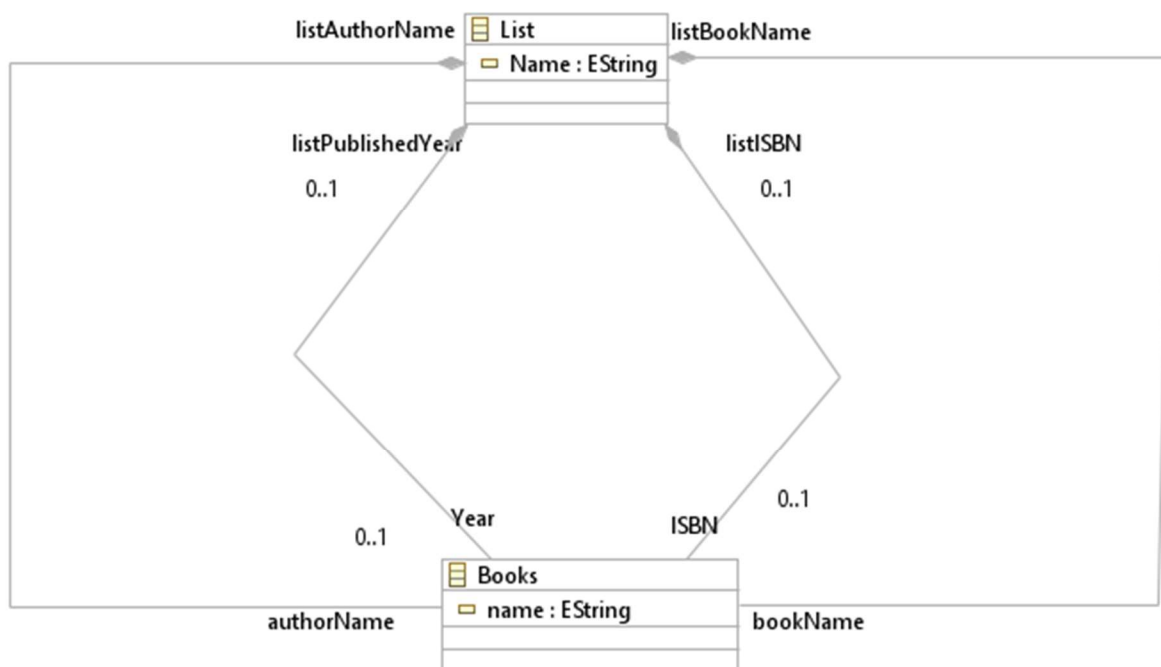
4. Results

Example 1: List to Books

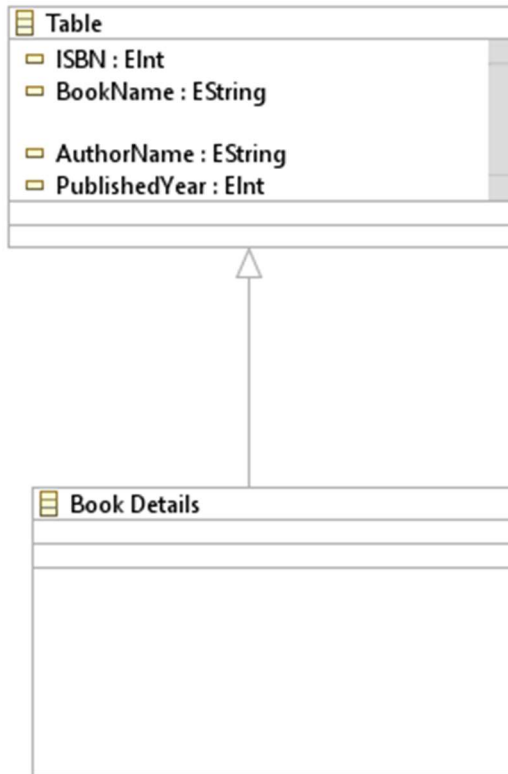
The example shows the process of converting a list of Author & Book Name which is scattered or which is not properly defined to a structured one. The list is too large so just took a sample of five values to do the conversion.

The list has ISBN Number, Book Name, Author name & Year of Publication which is not in a precise schema so tried to transform the Book Name and their Corresponding Details.

Figure 2 - Source Metamodel



The above diagram shows the source metamodel which conforms to the source model or input which is to be converted to the desired target model.

Figure 3 - Target Metamodel

The above figure shows the target metamodel which conforms to the target or output model.

List.ecore

```

<?xml version="1.0" encoding="UTF-8"?>
<ecore:EPackage xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:ecore="http://www.eclipse.org/emf/2002/Ecore" name="Lists">
  <eClassifiers xsi:type="ecore:EClass" name="List">
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="Name"
eType="ecore:EDatatype http://www.eclipse.org/emf/2002/Ecore#/EString"/>
    <eStructuralFeatures xsi:type="ecore:EReference" name="ISBN" eType="#//Books"
      containment="true"/>
    <eStructuralFeatures xsi:type="ecore:EReference" name="Year" eType="#//Books"
      containment="true"/>
    <eStructuralFeatures xsi:type="ecore:EReference" name="listBookName"
eType="#//Books"
      containment="true"/>
    <eStructuralFeatures xsi:type="ecore:EReference" eType="#//Books"
containment="true"/>
  </eClassifiers>
  <eClassifiers xsi:type="ecore:EClass" name="Books">
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="name" lowerBound="1"
eType="ecore:EDatatype http://www.eclipse.org/emf/2002/Ecore#/EString"/>
  </eClassifiers>
</ecore:EPackage>

```

Table.ecore

```

<?xml version="1.0" encoding="UTF-8"?>
<ecore:EPackage xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:ecore="http://www.eclipse.org/emf/2002/Ecore" name="Tables">
  <eClassifiers xsi:type="ecore:EClass" name="Table" abstract="true">
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="completeName"
lowerBound="1"
    eType="ecore:EDatatype http://www.eclipse.org/emf/2002/Ecore#/EString"/>
  </eClassifiers>
  <eClassifiers xsi:type="ecore:EClass" name="BookName" eSuperTypes="#//Table"/>
  <eClassifiers xsi:type="ecore:EClass" name="Details" eSuperTypes="#//Table"/>
</ecore:EPackage>
</xmi:XMI>

```

Sample Input - Source

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xmi:XMI xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI" xmlns="Lists">
  <List Name="Sample">
    <authorName fullName="Robin Sharma"/>
    <bookName fullName="Effective Objective-C 2.0"/>
    <isbn number="9780321917010"/>
    <publishedYear year="2013"/>
    <authorName fullName="Matt Galloway"/>
    <bookName fullName="The Mastery Manual"/>
    <isbn number="9788184954081"/>
    <publishedYear year="2015"/>
    <authorName fullName="Roger S.Pressman"/>
    <bookName fullName="PHP"/>
    <isbn number="0071508546"/>
    <publishedYear year="2008"/>
    <authorName fullName="Steven Holzner"/>
    <bookName fullName="Software Engineering"/>
    <isbn number="0071240837"/>
    <publishedYear year ="2013"/>
  </List>
</xmi:XMI>
```


Sample Output - Target

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xmi:XMI xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI" xmlns="Table">
<ISBN Details = "9788184954081"/>
<Year Details="2015"/>
<AuthorName="Robin Sharma"/>
<BookName ="The Mastery Manual"/>
<ISBN Details = "0071240837"/>
<Year Details="2008"/>
<AuthorName="Roger S.Pressman"/>
<BookName ="Software Engineering"/>
<ISBN Details = "9780321917010"/>
<Year Details="2013"/>
<AuthorName="Matt Galloway"/>
<BookName ="Effective Objective-C 2.0"/>
<ISBN Details = "0071508546"/>
<Year Details="2013"/>
<AuthorName="Steven Holzner"/>
<BookName ="PHP"/>
</xmi:XMI>
```

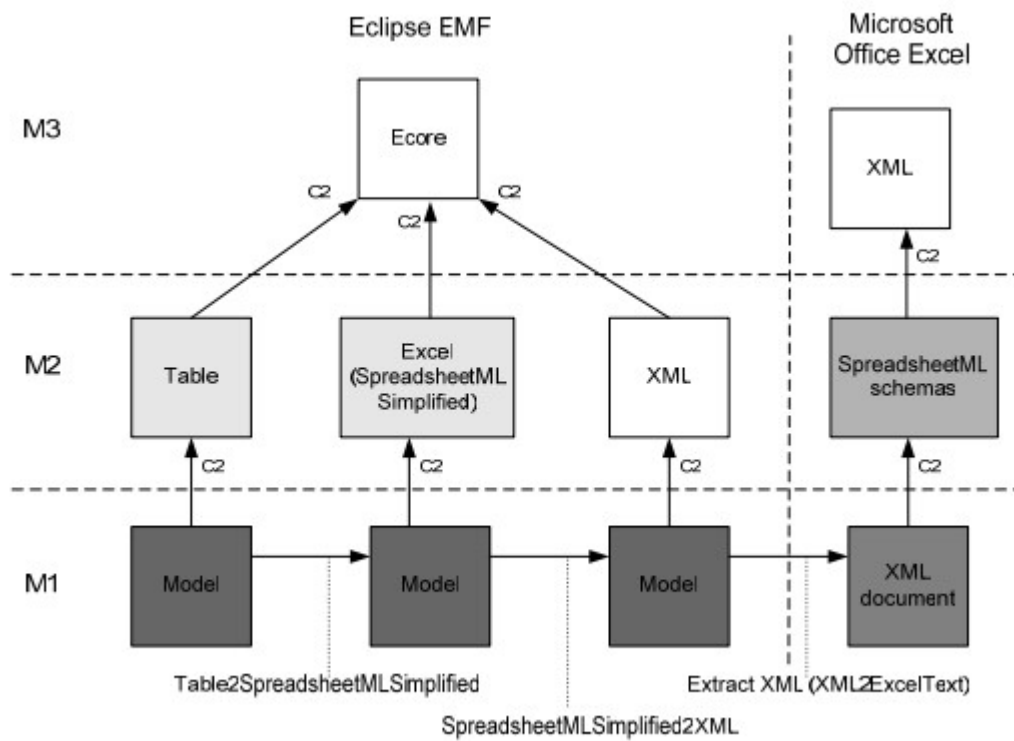
Example 2 – List to Simple Microsoft Excel Workbook

The below example shows a simple transformation of List to MS Office Excel Sheet.

To make the Table2MicrosoftOfficeExcel transformation we proceed in three steps.

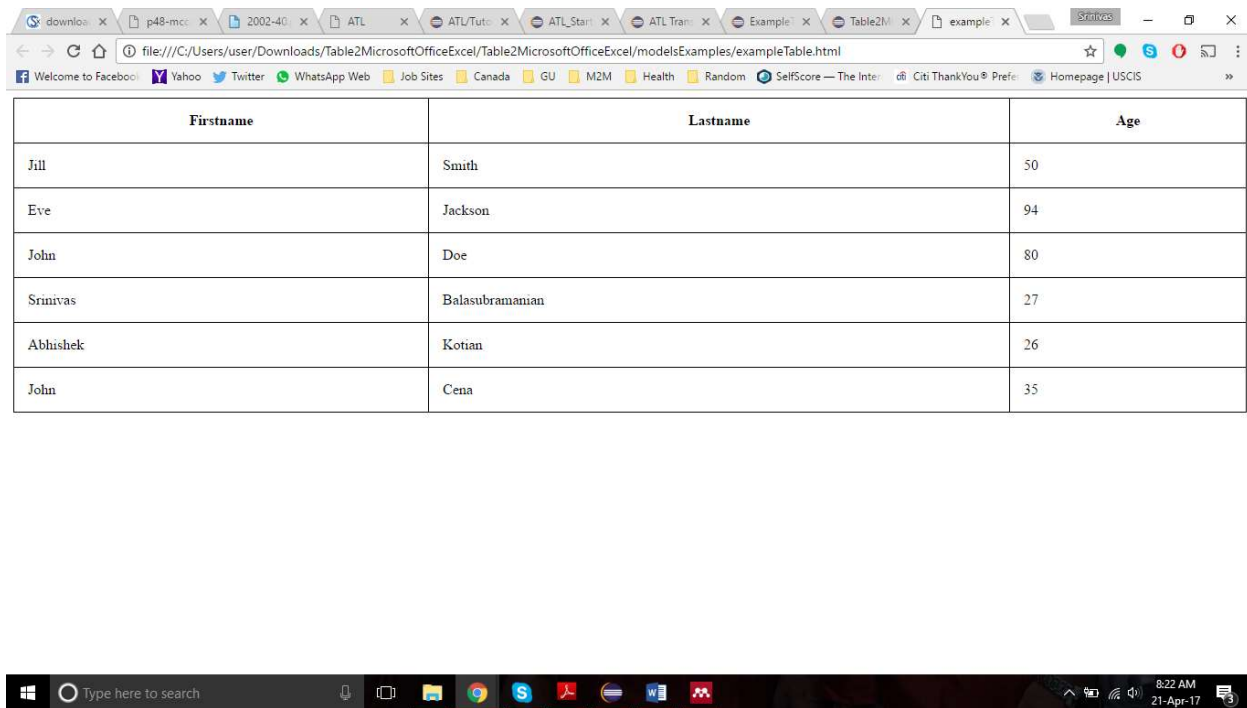
- from Table to SpreadsheetMLSimplified
- from SpreadsheetMLSimplified to XML
- from XML to Excel text which is explained in the below figure.

Figure 4 - Source to Target Transformation



The above diagram shows the transformation process of List to Corresponding MS Office Excel Worksheet. The input over here in this transformation is a table which is a source model and it is converted to the target Model MS Office Excel.

Figure 5 - Source – A simple table Example



Firstname	Lastname	Age
Jill	Smith	50
Eve	Jackson	94
John	Doe	80
Srinivas	Balasubramanian	27
Abhishek	Kotian	26
John	Cena	35

The above figure shows the sample list of Firstname, Lastname and Age of 5 people in the HTML format. This sample list in HTML is an input which is to be converted to MS office Excel Worksheet.

Figure 6 - Target – Corresponding MS Office Excel Worksheet

1	FirstName	LastName	Age							
2	Jill	Smith		50						
3	Eve	Jackson		94						
4	John	Doe		80						
5	Srinivas	Balasubramanian		27						
6	Abhishek	Kotian		26						
7	John	Cena		35						
8										
9										
10										
11										
12										
13										
14										
15										
16										
17										
18										
19										
20										
21										
22										
23										

The above figure is the corresponding MS office Excel Worksheet output for the given input.

Figure 7 - SpreadhseetML Simplified Metamodel

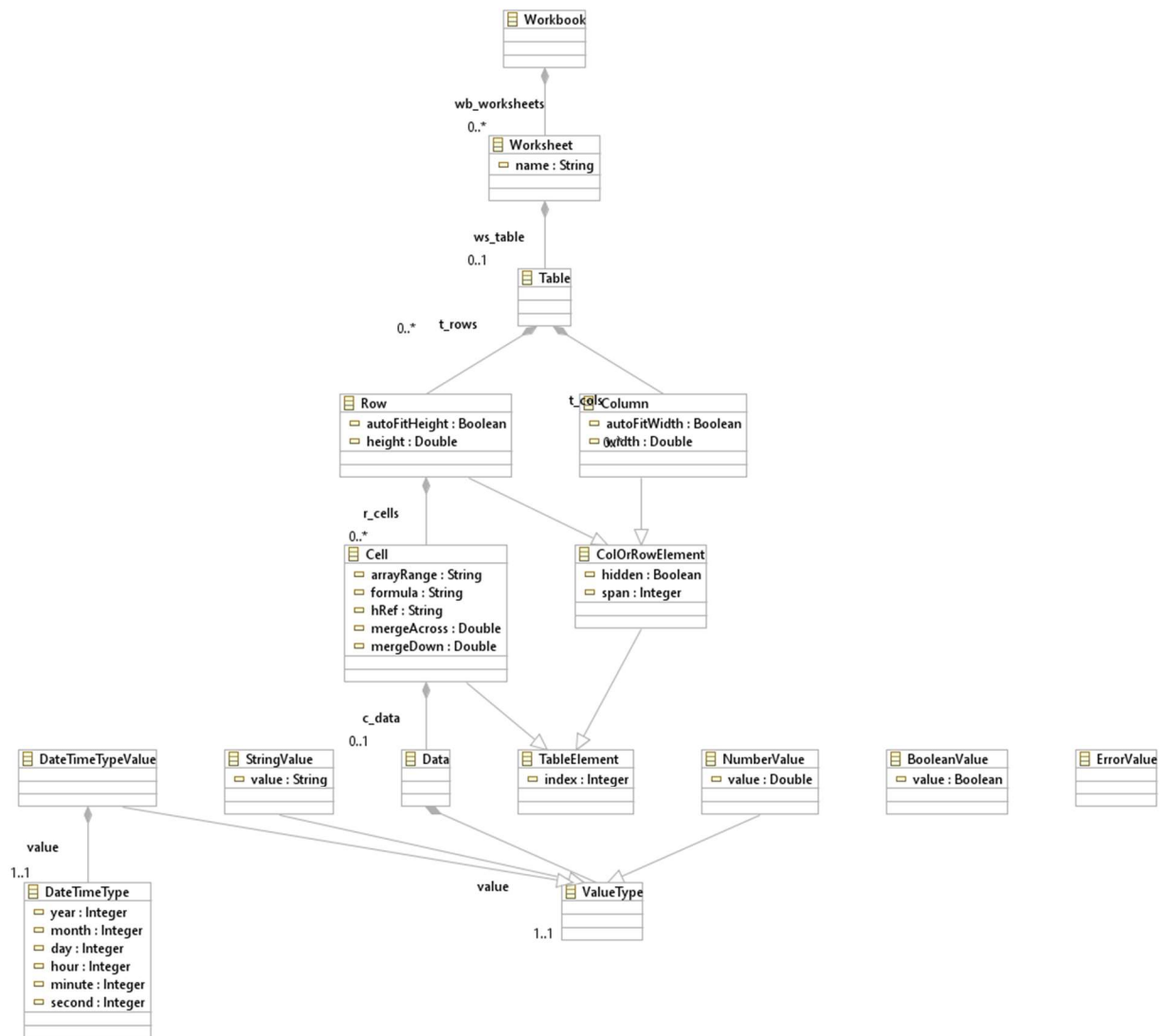


Figure 8 - Table Metamodel

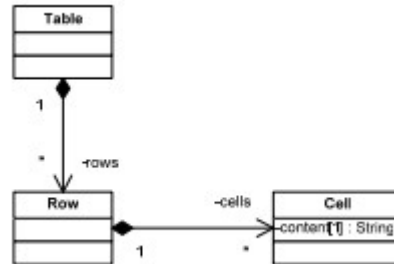
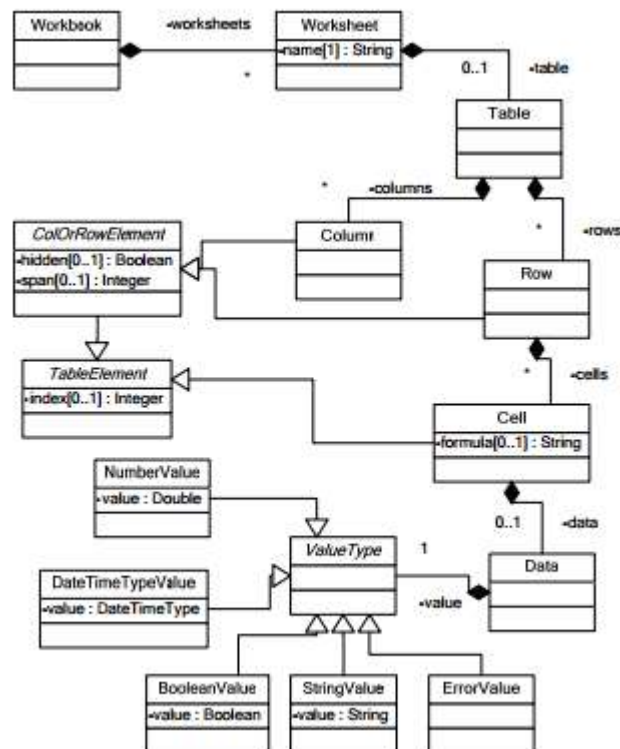


Figure 9 - Simple XML Metamodel



Coding to Convert List to Excel

```
<?xml version="1.0"?>
<?mso-application progid="Excel.Sheet"?>
<Workbook xmlns="urn:schemas-microsoft-com:office:spreadsheet" xmlns:ss="urn:schemas-
microsoft-com:office:spreadsheet">
  <Worksheet ss:Name="Java source code Info">
    <Table>
      <Column ss:Width="150.0"/>
      <Column ss:Width="150.0"/>
      <Column ss:Width="150.0"/>
      <Column ss:Width="150.0"/>
      <Column ss:Width="150.0"/>
      <Row>

        <Cell>
          <Data ss:Type="String">FirstName</Data>
        </Cell>
        <Cell>
          <Data ss:Type="String">LastName</Data>
        </Cell>
        <Cell>
          <Data ss:Type="String">Age</Data>
        </Cell>
      </Row>

      <Row>
        <Cell>
          <Data ss:Type="String">Jill</Data>
        </Cell>
        <Cell>
          <Data ss:Type="String">Smith</Data>
        </Cell>
        <Cell>
          <Data ss:Type="Number">50</Data>
        </Cell>
      </Row>

      <Row>
        <Cell>
          <Data ss:Type="String">Eve</Data>
        </Cell>
        <Cell>
          <Data ss:Type="String">Jackson</Data>
        </Cell>
        <Cell>
          <Data ss:Type="Number">94</Data>
        </Cell>
      </Row>

      <Row>
        <Cell>
          <Data ss:Type="String">John</Data>
        </Cell>
      </Row>
    </Table>
  </Worksheet>
</Workbook>
```

```

</Cell>
<Cell>
  <Data ss:Type="String">Doe</Data>
</Cell>
<Cell>
  <Data ss:Type="Number">80</Data>
</Cell>
</Row>

<Row>
<Cell>
  <Data ss:Type="String">Srinivas</Data>
</Cell>
<Cell>
  <Data ss:Type="String">Balasubramanian</Data>
</Cell>
<Cell>
  <Data ss:Type="Number">27</Data>
</Cell>
</Row>

<Row>
<Cell>
  <Data ss:Type="String">Abhishek</Data>
</Cell>
<Cell>
  <Data ss:Type="String">Kotian</Data>
</Cell>
<Cell>
  <Data ss:Type="Number">26</Data>
</Cell>
</Row>

<Row>
<Cell>
  <Data ss:Type="String">John</Data>
</Cell>
<Cell>
  <Data ss:Type="String">Cena</Data>
</Cell>
<Cell>
  <Data ss:Type="Number">35</Data>
</Cell>
</Row>

</Table>
</Worksheet>
</Workbook>

```


5. Conclusion and Future Work

The existing techniques has a lot of limitations in converting unstructured data to structured data. The aim of this research is to investigate whether the model driven engineering technique such as model to model transformation helps in converting unstructured data to the structured data.

The model driven engineering technique can be used to convert unstructured data to structured data by using model transformation languages such as ATL, QVT etc. The model to model transformation doesn't need lot of investment and it's time consuming. Based on the results from this research it is identified that model to model transformation will be highly useful in converting unstructured data to structured data.

Firstly, we identified that a random list of book details and the author name which is not in an order can be transformed to a list of Book name with their corresponding Author name and other details such as ISBN Number, Year of publication in a finite order. Secondly we identified a table in a webpage can be transformed and saved to Microsoft Excel Workbook.

Even the model to model transformation can be used to convert unstructured data to structured data the consideration to this technique has been given less importance. More expertise people are required to involve in this model to model transformation technique and should be well expertise in writing the code in Model Transformation Languages such as ATL or QVT. Also, there are lot of chances of increase in complexity when dealing with the model transformations as the models are manually created or while writing the transformation rules.

The research carried out in this paper dealt only with the sample of minimum five values, so more investigation is needed for the large number of data.

Also, we presented a new dimension of Model to Model Transformation which requires more adoption and recognition of the Model Engineering Technique. The proposed technique is to be validated and the limitations mentioned above are to be addressed with the utmost attention. If all the imitations are addressed, then this technique for converting unstructured data to structured data would be highly useful.

6. References

1. Wullianallur Raghupathi and Vijju Raghupathi, Big data analytics in healthcare: promise and potential, 2013
2. Jose Carvalho, Unstructured Data Management System: an approach to standard
3. Aravind Arasu, Hector Garcia-Molina, Extracting Structured Data from Web Pages
4. David A.Grossman, Integrating Structured Data and Text: A Relational Approach, 1997
5. Sven Fortuin, Model Driven Engineering: the decomposition of environments, 2016
6. Gabriele Bavota, Mining Unstructured Data in Software Repositories: Current and Future Trends, 2016, IEEE
7. Robert Blumberg, Shaku Atre, The problem with unstructured data
8. Kalli Srinivasa Nageswara Prasad, Prof S. Ramakrishna, Text Analytics to Data Warehousing.
9. Xia Hu, Huan Liu, Text Analytics in Social media
10. Nora Koch, Santiago Meliá-Beigbeder, Nathalie Moreno-Vergara, Vicente Pelechano-Ferragud, Fernando Sánchez-Figueroa, and Juan-Manuel Vara-Mesa, Model Driven Web Engineering
11. Gabriele Bavota, Mining Unstructured Data in Software Repositories: Current and Future Trends, Software Analysis, Evolution, and Reengineering (SANER), 2016 IEEE 23rd International Conference on 14-18 March 2016
12. Line Eikvil, Optical Character recognition, December 1993
13. Tetsuya Nasukawa, Jeonghee Yi, Sentiment Analysis: Capturing Favorability using Natural Language Processing
14. Ravina Mithe, Supriya Indalkar, Nilam Divekar, Optical Character Recognition, International Journal of Recent technology and Engineering, March 2013
15. V Nagarjana Devi Duvvuri, V.Rajanikanth Thatiparthi, Rajashekar Pantangi, Akhil Gangavarapu, Sentiment Analysis Using Harn Algorithm, IT Convergence and Security (ICITCS), 2016 6th International Conference on 26-26 Sept. 2016
16. Chandni Saini, Vinay Arora, Information retrieval in web crawling: A survey, Advances in Computing, Communications and Informatics (ICACCI), 2016 International Conference on 21-24 Sept 2016
17. Hian Chye Koh, Gerald Tan, Data Mining Applications in Healthcare
18. Namrata H.S, B.S. Satpute, Pramod Patil, Web Forum Crawling Techniques, International Journal of Computer Applications (0975 – 8887) Volume 85 – No 17, January 2014
19. Kalli Srinivasa Nageswara Prasad, Prof S. Ramakrishna, Text Analytics to Data Warehousing.
20. Wullianallur Raghupathi and Vijju Raghupathi, Big data analytics in healthcare: promise and potential, 07 Feb 2014

21. Vishal Gupta, Gurpreet S. Lehal, A survey of Text Mining Techniques and Applications, Journal of Emerging Technologies in Web Intelligence, August 2009
22. Saumya Salian, Challenges with Big Data Analytics, International Journal of Science and Research
23. Udo Hahn, Inderjeet Mani, The challenges of Automatic Summarization, November 2000
24. Carlos Castillo, Effective Web Crawling, November 2004
25. Eric Brill, Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part-of-Speech Tagging
26. Beatrice Santorini, Part-of-Speech Tagging Guidelines for the Penn Treebank Project (3rd Revision), July 1990
27. Rahul S. Dudhabaware, Mangala S. Madankar, Review on Natural Language Processing Tasks for Text Documents, Computational Intelligence and Computing Research (ICCIC), 2014 IEEE International Conference on, 18-20 Dec. 2014
28. Anjali Ganesh Jivani, A comparative study of Stemming Algorithms. November 2011
29. V.N. Gudivada, W.I. Grosky, V.V. Raghavan, Information retrieval on the World Wide Web, IEEE Internet Computing, 06 August 2002
30. Arash Habibi Lashkari, Fereshteh Mahdavi, Vahid Ghomi, A Boolean Model in Information Retrieval for Search Engines, Information Management and Engineering, 2009. ICIME '09. International Conference on 3-5 April 2009
31. Amir Gandomi, Murtaza Haider, Beyond the hype: Big data concepts, methods, and analytics, International Journal of Information Management
32. Poonam Vashisht, Vishal Gupta, Big data analytics techniques: A survey, Green Computing and Internet of Things International Conference, 8-10 Oct 2015
33. Jai Prakash Verma, Smita Agrawal, Bankim Patel and Atul Patel, Big data analytics: Challenges and applications for Text, Audio, Video and social media data, International Journal on Soft Computing, Artificial Intelligence and Applications (IJSCAI), February 2016
34. Robert France, Bernhard Rumpe, Model-driven Development of Complex Software: A Research Roadmap,
35. Matthias Biehl, Literature Study on Model Transformations, July 2010
36. Gabriel Costa Silva, Louis M. Rose, and Radu Calinescu, A Qualitative Study of Model Transformation Development Approaches: Supporting Novice Developers
37. B. Hailpern, P. Tarr, Model Driven Development: The good, the bad and the ugly.
38. Colin Atkinson, Thomas Kuhne, Model Driven Development: A Metamodeling Foundation

39. Deniz Cetinkaya, Alexander Verbraeck, Metamodeling and Model Transformations in Modeling and Simulation, 2011, IEEE
40. Jean Bezivin, Atlas Group: INRIA and LINA, Model Driven Engineering: An Emerging Technical Space 2006, Springer
41. Sven Fortuin, Model Driven Engineering: the decomposition of environments, 2016
42. Stephen J. Mellor, Kendall Scott, Axel Uhl, Dirk Weise, Model-Driven Architecture,
43. Matthias Biehl, Literature Study on Model Transformations, July 2010
44. Line Eikvil, Optical Character recognition, December 1993
45. HianChye Koh, Gerald Tan, Data Mining Applications in Healthcare
46. William M. Darling, Michael J. Paul, Fei Song, Unsupervised Part-of-Speech Tagging in Noisy and Esoteric Domains with a Syntactic-Semantic Bayesian HMM
47. K.M Arif Aziz, Evaluating Model Transformation Technologies, May 2011
48. Namrata H.S Bamrah, B.S. Satpute, Pramod Patil, Web Forum Crawling Techniques, January 2014