DATA PREPROCESSING: A PRELIMINARY STEP FOR WEB DATA MINING

Huma Jamshed
Sir Syed University of Engineering and Technology. University of Karachi. Karachi (Pakistan)
E–mail: humajamshed@yahoo.com

M. Sadiq Ali Khan
Sir Syed University of Engineering and Technology. University of Karachi. Karachi (Pakistan)
E–mail: msakhan@uok.edu.pk

Muhammad Khurram
Sir Syed University of Engineering and Technology. University of Karachi. Karachi (Pakistan)
E–mail: muhammadkhurram@gmail.com

Syed Inayatullah
Sir Syed University of Engineering and Technology. University of Karachi. Karachi (Pakistan)
E–mail: inayat@uok.edu.pk

Sameen Athar
Sir Syed University of Engineering and Technology. University of Karachi. Karachi (Pakistan)
E–mail: sameenathar@yahoo.com

Recepción: 05/03/2019 **Aceptación:** 12/04/2019 **Publicación:** 17/05/2019

Citación sugerida:
Jamshed, H., Ali Khan, M. S., Khurram, M., Inayatullah, S. y Athar, S. (2019). Data Preprocessing: A preliminary step for web data mining. *3C Tecnología. Glosas de innovación aplicadas a la pyme. Edición Especial, Mayo 2019*, pp. 206–221. doi: http://dx.doi.org/10.17993/3ctecno.2019.specialissue2.206–221

Suggested citation:
Jamshed, H., Ali Khan, M. S., Khurram, M., Inayatullah, S. & Athar, S. (2019). Data Preprocessing: A preliminary step for web data mining. *3C Tecnología. Glosas de innovación aplicadas a la pyme. Special Issue, May 2019*, pp. 206–221. doi: http://dx.doi.org/10.17993/3ctecno.2019.specialissue2.206–221

ABSTRACT

In recent years immense growth of data i.e. big data is observed resulting in a brighter and more optimized future. Big Data demands large computational infrastructure with high–performance processing capabilities. Preparing big data for mining and analysis is a challenging task and requires data to be preprocessed to improve the quality of raw data. The data instance representation and quality are foremost. Data preprocessing is preliminary data mining practice in which raw data is transformed into a format suitable for another processing procedure. Data preprocessing improves the data quality by cleaning, normalizing, transforming and extracting relevant feature from raw data. Data preprocessing significantly improve the performance of machine learning algorithms which in turn leads to accurate data mining. Knowledge discovery from noisy, irrelevant and redundant data is a difficult task therefore precise identification of extreme values and outlier, filling up missing values poses challenges. This paper discusses various big data pre–processing techniques in order to prepare it for mining and analysis tasks.

KEYWORDS

Big Data, Data Pre–processing, Data mining, Data preparation, Text Pre–processing.

DOI: http://dx.doi.org/10.17993/3ctecno.2019.specialissue2.206-221

1. INTRODUCTION

Year after year, organizations have realized the benefits that big data analytics provides. Data scientist and researchers demands for the evolution of current practices for processing raw data. Automated Information extraction is impossible from the huge data repository as most data is unstructured. Cloud computing services have also lead us with a growing rate of data on the web as these services are cost–effective and easy to use. This phenomenon undoubtedly signifies a challenge for the data scientist and analyst, therefore Big Data characterized as very high volume, velocity and variety require new high–performance processing (Xindong, Xingquan, Gong–Qing & Ding, 2014). Process of extraction of relevant and useful information from the data deluge is known as data mining which is utterly dependent on the quality of data. The raw data is usually vulnerable to noise, and is incomplete or inconsistent and contain outlier values. Thus, this data has to be processed prior to the application of data mining (Alasadi & Bhaya, 2017).

Data preprocessing involves the transformation of the raw dataset into an understandable format. Preprocessing data is a fundamental stage in data mining to improve data efficiency. The data preprocessing methods directly affect the outcomes of any analytic algorithm; however, the methods of pre–processing may vary for the area of application. Data pre–processing is a significant stage in the data mining process. According to a report by Aberdeen Group, data preparation refers to any action intended to increase the quality, usability, accessibility, or portability of data. The ultimate objective of data preparation is to allow analytical systems with clean and consumable data to be transformed into actionable insights. Data preprocessing embrace numerous practices such as cleaning, integration, transformation and reduction. The preprocessing phase may consume a substantial amount of time but the outcome is a final data set, which is anticipated correct and beneficial for further data mining algorithms.

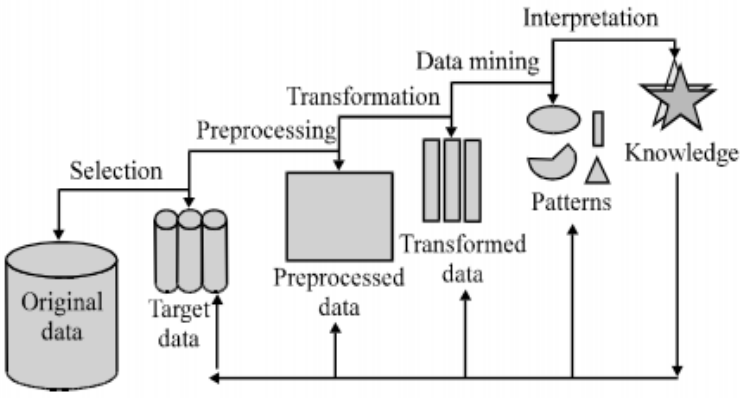

Figure 1. Knowledge Discovery Process in Data Mining.

2. BACKGROUND

The raw data available on data warehouse, data marts, database files (Jiawei, Micheline & Jian, 2012) are generally not organized for analysis as it may be incomplete, inconsistent or it may be distributed into a various table or represented in a different format, in short, it is dirty. The process of discovering knowledge from the massive chronological data sources is called as Knowledge Discovery in databases (KDD) or Data Mining (Malley, Ramazzotti & Wu, 2016; Gupta & Gurpreet, 2009). It is the era of big data and every field of life are generating data at a drastic level. The most challenging task is to gain the right information from present data sources.

The task of reorganizing data is known as data preparation. It is used to discover the anticipated knowledge. It incorporates understanding domain based problem under consideration and then a collection of targeted data to achieve anticipated goals (Gülser, İnci & Murat, 2011). Forrester estimates up to 80 per cent of data analyst time is consumed in preparing data (Goetz, 2015). The selected data can then be preprocessed for data mining. Data pre–processing is the finest solution to increase data quality. Data preprocessing includes cleansing of data, normalization of data, transformation, feature extraction and selection, etc. The processed data is the training set for the machine learning algorithm.

DOI: http://dx.doi.org/10.17993/3ctecno.2019.specialissue2.206-221

3. DATA PRE–PROCESSING STAGES

3.1. DATA CLEANING

The first stage of data preprocessing is Data cleaning which recognizes partial, incorrect, imprecise or inappropriate parts of the data from datasets (Tamraparni & Theodore, 2003). Data cleaning may eliminate typographical errors. It may ignore tuple contains missing values or alter values compared to a known list of entities. The data then becomes consistent with other data sets available in the system. Precisely, data cleaning comprises the following four basic steps as described in Table 1.

Table 1. Data Cleaning Steps.

Steps	Description
Data Analysis	Dirty data detection by reviewing dataset, quality of data, meta data.
Define Work Flow	Define the cleaning rules by considering heterogeneity degree among diverse data source, then make the work flow order of cleaning rules such as cleaning particular data type, condition, strategy to apply etc.
Execute defined rules	Rendering the defined rules on source dataset process, and display resulted in clean data to the user.
Verification	Verify the accuracy and efficiency of the cleaning rules whether it content user requirements.

Step 2–3 repetitively executed till all problems related to data quality get solved. Repeat steps 1–4 until user requirements are met to clean the data. Handling missing values is difficult as improperly handled the missing values may lead to poor knowledge extracted (Hai & Shouhong, 2009). Expectation–Maximization (EM) algorithm, Imputation, filtering are generally considered for handling missing values ("Expectation maximization algorithm"). Various data cleansing solutions apply validated data set on dirty data in–order to clean it. Some tools use data enhancement techniques which makes incomplete data set complete by the addition of related information. Binning methods can be used to remove noisy data. Clustering technique is used to detect outliers (Jiawei, *et al.*, 2012). Data can also be smooth out by fitting it into a regression function. Numerous regression procedures such as linear, multiple or logistic regression are used to regulate regression function.

3.2. DATA INTEGRATION

Data Integration is the method of merging data derived from different sources of data into a consistent dataset. Data on the web is expanding in size and complexity, and is either unstructured or semi–structures. Integration of data is an extremely cumbersome and iterative process. The considerations during the integration process are mostly related to standards of heterogeneous data sources. Secondly, the process of integrating new data sources to the existing dataset is time–consuming, ultimately results in inappropriate consumption of valuable information. ELT (Extract–Transform–Load) tools are used to handle a larger volume of data; it integrates diverse sources into a single physical location, provides uniform conceptual schemas and provides querying capabilities.

3.3. DATA TRANSFORMATION

Raw data is usually transformed into a format suitable for analysis. Data can be normalized for instance transformation of the numerical variable to a common range. Data normalization can be achieved using range normalization technique or z–score method. Categorical data can also be transformed using aggregation which merges two or more attributes into a single attribute. Generalization can be applied on low–level attributes which are transformed to a higher level.

3.4. DATA REDUCTION

Multifaceted exploration of huge data sources may consume considerable time or even be infeasible. When the number of predictor variables or the number of instances becomes large, mining algorithms suffer from dimensionality handling problems (Jiawei, *et al.*, 2012). The last stage of data preprocessing is data reduction. Data reduction makes input data more effective in representation without loosening its integrity. Data reduction may or may not be lossless. The end database may contain all the information of the original database in well–organized format (Bellatreche & Chakravarthy, 2017). Encoding techniques, hierarchy distribution data cube aggregation can be used to reduce the size of the dataset. Data reduction harmonizes feature selection process. Instance

DOI: http://dx.doi.org/10.17993/3ctecno.2019.specialissue2.206-221

selection (Vijayarani, Ilamathi & Nithya, 2015) and Instance generation are two approaches used by data mining algorithm to reduce data size.

4. WEB DATA PREPROCESSING FRAMEWORK

World Wide Web is a huge repository of an awful textual data most of it being created on a daily basis, reaching from structured to semi–structured to completely unstructured (Andrew, 2015). How can we utilize that data in a productive way? What can we do with it? The answer to these two questions is totally dependent on what is our objective.

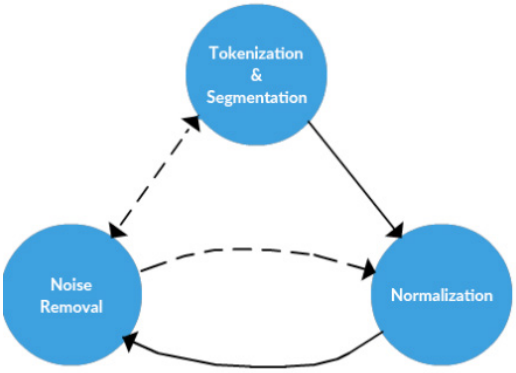

Figure 2. Framework for web content Pre–processing.

To leverage the availability of all of this data, it has to be preprocessed which entails various steps but it may or not apply to a given task, but usually plunge below the broad categories of tokenization, normalization, and substitution.

- Tokenization; in textual data preprocessing tokenization is used to spit long strings of text into smaller one for example sentences can be tokenized into words, etc. It is also known as text segmentation or lexical analysis.
- Normalization; It generally refers to a series of related tasks in order to places all words on equal footing or uniformity. For instance performing stemming, lemmatization, changing the case upper to lower or lower to upper, punctuation, space or stop words removal, the substitution of numbers with their equivalent words etc.

- Substitution or Noise Removal; text data on websites is wrapped in HTML or XML tags, pattern matching or regular expression can be used to extract desired text by removing HTML, XML, etc. markup and metadata.

5. CASE STUDY

Our objective is to do preprocessing on the predetermined body of text; so that we are left with artifact's which will be more valuable and meaningful for any text mining algorithm. The approach proposed here is fully applicable to any web page content. We will remove noise, in our case, it is HTML tags and substitutes English language contraction. Then the content will be tokenized and finally, we are going to normalize the text. We have used PHP as a scripting language to perform preprocessing on the text. We have explored PHP Natural Language Toolkit for tokenization. Figure 3 shows a dummy HTML page content certainly, but the steps to preprocessing this data are fully transferable.

```
sample = """<h1>Title Goes Here</h1>
<b>Bolded Text</b>
<i>Italicized Text</i>
<img src="this should all be gone"/>
<a href="this will be gone, too">But this will still be here!</a>
I run. He ran. She is running. Will they stop running?
I talked. She was talking. They talked to them about running. Who ran to the talking runner?
[Some text we don't want to keep is in here]
¡Sebastián, Nicolás, Alejandro and Jéronimo are going to the store tomorrow morning!
something... is! wrong() with.,; this :: sentence.
I can't do this anymore. I didn't know them.
Why couldn't you have dinner at the restaurant?
My favorite movie franchises, in order: Indiana Jones;
Marvel Cinematic Universe; Star Wars; Back to the Future; Harry Potter.
Don't do it.... Just don't. Billy! I know what you're doing.
This is a great little house you've got here.
[This is some other unwanted text]
John: "Well, well, well."
James: "There, there. There, there."
  
There are a lot of reasons not to do this. There are 101 reasons not to do it. 1000000 reasons, actually.
I have to go get 2 tutus from 2 different stores, too.
22   45   1067   445
{{Here is some stuff inside of double curly braces.}}
{Here is more stuff in single curly braces.}
[DELETE]
</body>
</html>"""
```

Figure 3. Basic HTML page content.

DOI: http://dx.doi.org/10.17993/3ctecno.2019.specialissue2.206-221

5.1. NOISE REMOVAL AND SUBSTITUTION

The data preprocessing pipeline will start with noise removal as it is not task depended. The line of codes in Figure 4 reads in the text file called sample.txt which contains dummy HTML data shown in Figure 3. It calls PHP built–in function to strip of HTML tags.

```
$file_to_read="C:\\xammp\\sample.txt";
$page_contents = file_get_contents($file_to_read);
$plain_text= strip_tags($page_contents);
```

Figure 4. Code to strip HTML tags.

It is beneficial to remove English language contraction with their expansion before tokenization as it will split word such as "didn't" into "did" and "n't" rather than "did" and "not". We implemented contradiction expansion by calling list of contraction available in MYSQL database and then comparing it with our content. It then replaced every occurrence of matched contraction will expansion.

```
$conn = mysqli_connect('servername', 'username', 'password');
 if (!$conn)
 {
   die("Database Connection Failed" . mysqli_error());
 }

 $select_db = mysqli_select_db($conn, 'databasename');
 if (!$select_db)
 {
   die("Database Selection Failed" . mysqli_error());
 }
$str = $plain_text;
$query=mysqli_query($conn, "select * from contradiction_list")
or die(mysqli_error($conn));
    while($row=mysqli_fetch_array($query))
         {
             $word=$row['contraction'];
             $str= str_replace($word, $row['meaning'], $str);

           }
```

Figure 5. Substitution of contractions.

```
Title Goes Here
Bolded Text
Italicized Text

But this will still be here!

I run. He ran. She is running. Will they stop running?

I talked. She was talking. They talked to them about running. Who ran to the talking


¡Sebastián, Nicolás, Alejandro and Jéronimo are going to the store tomorrow morning!

something... is! wrong() with.,; this :: sentence.

I cannot do this anymore. I did not know them. Why could not you have dinner at the

My favorite movie franchises, in order: Indiana Jones; Marvel Cinematic Universe; St

do not do it.... Just do not. Billy! I know what you are doing. This is a great litt


John: "Well, well, well."
James: "There, there. There, there."


There are a lot of reasons not to do this. There are 101 reasons not to do it. 10000
I have to go get 2 tutus from 2 different stores, too.

22    45   1067   445

{{Here is some stuff inside of double curly braces.}}
{Here is more stuff in single curly braces.}
```

Figure 6. Text after de–noising.

5.2. TOKENIZATION

For tokenization, we have used PHP Natural Language Processing (NLP) toolkit. PHP supports various kinds of tokenization under tokenizers namespace. We are using RegexTokenizer.

DOI: http://dx.doi.org/10.17993/3ctecno.2019.specialissue2.206-221

```php
//Toekenization
use NlpTools\Tokenizers\RegexTokenizer;

$s = $str;

$rtok = new RegexTokenizer(array(
    array("/\s+/"," "),
    // replace many spaces with a single space
    array("/'(m|ve|d|s)/", " '\$1"),
    // split I've, it's, we've, we'd, ..
    "/ /"
    // split on every space
));

print_r($rtok->tokenize($s));
```

Figure 7. Tokenization using PHP NLP toolkit.

```
['Title', 'Goes', 'Here', 'Bolded', 'Text', 'Italicized', 'Text', 'But', 'this', 'wil
'be', 'here', '!', 'I', 'run', '.', 'He', 'ran', '.', 'She', 'is', 'running', '.', 'W
'stop', 'running', '?', 'I', 'talked', '.', 'She', 'was', 'talking', '.', 'They', 'ta
'about', 'running', '.', 'Who', 'ran', 'to', 'the', 'talking', 'runner', '?', '¡Sebas
'Nicolás', ',', 'Alejandro', 'and', 'Jéronimo', 'are', 'going', 'tot', 'he', 'store',
'morning', '!', 'something', '...', 'is', '!', 'wrong', '(', ')', 'with.', ',', ';',
'sentence', '.', 'I', 'can', 'not', 'do', 'this', 'anymore', '.', 'I', 'did', 'not',
'Why', 'could', 'not', 'you', 'have', 'dinner', 'at', 'the', 'restaurant', '?', 'My',
'movie', 'franchises', ',', 'in', 'order', ':', 'Indiana', 'Jones', ';', 'Star', 'War
'Cinematic', 'Universe', ';', 'Back', 'to', 'the', 'Future', ';', 'Harry', 'Potter',
'do', 'it', '...', '.', 'Just', 'do', 'not', '.', 'Billy', '!', 'I', 'know', 'what',
'doing', '.', 'This', 'is', 'a', 'great', 'little', 'house', 'you', 'have', 'got', 'h
':', '``', 'Well', ',', 'well', ',', 'well', '.', "'", 'James', ':', '``', 'There',
'There', ',', 'there', '.', "'", 'There', 'are', 'a', 'lot', 'of', 'reasons', 'not',
'.', 'There', 'are', '101', 'reasons', 'not', 'to', 'do', 'it', '.', '1000000', 'reas
'actually', '.', 'I', 'have', 'to', 'go', 'get', '2', 'tutus', 'from', '2', 'differen
'too', '.', '22', '45', '1067', '445', '{', '{', 'Here', 'is', 'some', 'stuff', 'insi
'curly', 'braces', '.', '}', '}', '{', 'Here', 'is', 'more', 'stuff', 'in', 'single',
'.', '}']
```

Figure 8. Words Token.

5.3. NORMALIZATION

For text normalizing we will perform (1) stemming (2) everything else.

Stemming: The aim of this step is to condense inflectional forms of a word to a common base form. For instance: cars to car

```php
//stemming

foreach ($rtok as &$value) {
$stem_words[]= stemword($value, 'english', 'UTF_8');
}
```

Figure 9. Stemming English language words.

Everything Else: This step will transform all word into lowercase, remove non–ascii words, remove punctuations, replace numbers, and remove stop word.

```php
//remove non ascii
function convert_to_normal_text($stem_words) {
    $new_words;
    $normal_characters = "a-zA-Z0-9\s`~!@#$%^&*()_+-={}|::<>?,.\/\"\'\\\[\]";
    foreach ($stem_words as $value) {
    $normal_text = preg_replace("/[^$normal_characters]/", '', $value);
    $new_word[]=$normal_text;
    }

    return $new_word;
}
//remove stop words
function removeCommonWords($input){

    // EEEEEEK Stop words
    $new_words;
    foreach ($input as $value) {
    $new_word[]= preg_replace('/\b('.implode('|',$stopWords).')\b/','',$value);
    }
    return $new_word;
}

//convert to lowercase
function wordlowercase($words){

    $new_words;
    foreach ($words as $value) {
    $new_word[]= strtolower($value);
    }
    return $new_word;
}
//remove punctuation
function strip_punctuation($words) {

    $new_words;
    foreach ($words as $value) {
     $string = preg_replace("/[:punct:]+/", "", $value);
    $string = str_replace(" +", "", $string);
    if ($string !="")
    {
    $new_words[]=$string;
    }
    return $new_word;
}
/// number to word
function convert_number_toword($words) {

    $new_words;
    foreach ($words as $value) {
    if (is_numeric($value))
    {
    $value = number_to_word($vlaue);
    }
    $new_words[]=$value;
    }
    return $new_word;
}
```

Figure 10. PHP functions for text normalization.

DOI: http://dx.doi.org/10.17993/3ctecno.2019.specialissue2.206-221

```
['title', 'go', 'bolded', 'text', 'italicize', 'text', 'still', 'run', 'run', 'run',
'talk', 'talk', 'talk', 'run', 'run', 'talk', 'runner', 'sebastian', 'nicolas', 'alej
'go', 'store', 'tomorrow', 'morning', 'something', 'wrong', 'sentence', 'anymore', 'k
'dinner', 'restaurant', 'favorite', 'movie', 'franchise', 'order', 'indiana', 'jones'
'cinematic', 'universe', 'star', 'war', 'back', 'future', 'harry', 'potter', 'billy',
'little', 'house', 'get', 'john', 'well', 'well', 'well', 'jam', 'lot', 'reason', 'on
'reason', 'one million', 'reason', 'actually', 'go', 'get', 'two', 'tutus', 'two', 'd
'twenty-two', 'forty-five', 'one thousand and sixty-seven', 'four hundred and forty-f
'inside', 'double', 'curly', 'brace', 'stuff', 'single', 'curly', 'brace']
```

Figure 11. Final output after applying all preprocessing steps.

The simple text data preprocessing process results are shown in Figure 11.

6. CONCLUSION

Any data analysis algorithm will fail to discover hidden pattern or trend from data if the dataset under observation is inadequate, irrelevant or incomplete. Thus data preprocessing is a central phase in any data analysis process. The preprocessing of data resolves numerous kinds of problems such as noisy, redundancy, missing values, etc. High quality results are only achievable with high quality of data which in turn also reduce the cost for data mining. The foundation of decision making system in any organization is the three C's properties of data i.e. Completeness, Consistency and Correctness. Deprived quality of data quality effects decision making process which eventually decreases customer's satisfaction. Furthermore larger dataset affects the performance of any machine learning algorithm, therefore instance selection lessens data and is efficient approach to make machine learning algorithm work effectively.

ACKNOWLEDGEMENTS

This work was supported by Department of Computer Science University of Karachi. We are thankful to our colleagues from computer science department who provided awareness and expertise that significantly helped this work. The authors would like to thank the anonymous reviewers for their valuable and constructive comments on improving the paper.

REFERENCES

Alasadi, S. & Bhaya, W. (2017). Review of Data Preprocessing Techniques in Data Mining. *Journal of Engineering and Applied Sciences, 12(*16), pp. 4102–4102. doi: http://dx.doi.org/10.3923/jeasci.2017.4102.4107

Andrew, K. (2015). The research of text preprocessing effect on text documents classification efficiency. *International Conference Stability and Control Processes IEEE*, St. Petersburg, Russia.

Bellatreche, L. & Chakravarthy, S. (2017). Big Data Analytics and Knowledge Discovery. Proceeding of 19th International Conference DAWak Lyon France.

Expectation maximization algorithm, Wikipedia, Retrieved February 10, 2019, from https://en.wikipedia.org/wiki/Expectation%E2%80%93maximization_algorithm

Goetz, M. (2015). Three ways data preparation tools help you get ahead of Big Data. Retrieved from https://go.forrester.com/blogs/15–02–17–3_ways_data_preparation_tools_help_you_get_ahead_of_big_data/

Gülser, K., İnci, B. & Murat, C. (2011). A review of data mining applications for quality improvement in manufacturing industry. *Expert System with application, 38*(10), pp. 13448–13467. doi: http://dx.doi.org/10.1016/j.eswa.2011.04.063

Gupta, V. & Gurpreet, S. (2009). A Survey of Text Mining Techniques and Applications. *Journal of Emerging Technologies in Web Intelligence, 1*(1), pp. 60–76.

Hai, W. & Shouhong, W. (2009). Mining incomplete survey data through classification. *Knowledge and Information Systems Springer , 24*(2), pp. 221–233. doi: http://dx.doi.org/10.1007/s10115–009–0245–8

Jiawei, H., Micheline,K. & Jian, P. (2012). Data Mining Concepts and Techniques. (3rd ed.), USA: Morgan Kaufmann.

DOI: http://dx.doi.org/10.17993/3ctecno.2019.specialissue2.206-221

Malley, B., Ramazzotti, D. & Wu, J. (2016). Data Pre–processing; Secondary Analysis of Electronic Health Records. Springer. Retrieved from https://link.springer.com/book/10.1007/978–3–319–43742–2

Tamraparni, D. & Theodore, J. (2003). Exploratory data mining and data cleaning. New York, USA, John Wiley & Sons.

Vijayarani, S., Ilamathi, M., & Nithya, M. (2015). Preprocessing Techniques for Text Mining – An Overview. *International Journal of Computer Science & Communication Networks,* 5(1), pp. 7–16.

Xindong, W., Xingquan, Z., Gong–Qing, W. & Ding, W. (2014). Data Mining with Big Data. *IEEE transactions on knowledge and data engineering, 26*(1), pp. 97–107. doi: http://dx.doi.org/10.1109/TKDE.2013.109