International Journal of Electronics Communication and Computer Engineering
Volume 2, Issue 2, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

Preprocessing and Morphological Analysis in Text Mining

Krishna Kumar Mohbey[1], Sachin Tiwari[2]
Lecturer, Department of Computer Applications[12]
kmohbey@gmail.com[1],sachinmcavds97@gmail.com[2]
Samrat Ashok Technological Institute, Vidisha (M.P.) [12]

Abstract- This paper is based on the preprocessing activities which is performed by the software or language translators before applying mining algorithms on the huge data. Text mining is an important area of Data mining and it plays a vital role for extracting useful information from the huge database or data ware house. But before applying the text mining or information extraction process, preprocessing is must because the given data or dataset have the noisy, incomplete, inconsistent, dirty and unformatted data. In this paper we try to collect the necessary requirements for preprocessing. When we complete the preprocess task then we can easily extract the knowledgful information using mining strategy. This paper also provides the information about the analysis of data like tokenization, stemming and semantic analysis like phrase recognition and parsing. This paper also collect the procedures for preprocessing data i.e. it describe that how the stemming, tokenization or parsing are applied.

***Keywords*—Morphological analysis, parsing, stemming, Tokenization.**

I. INTRODUCTION

Text mining is a new area of computer science which fosters strong connections with natural language processing, data mining, machine learning, information retrieval and knowledge management. Text mining seeks to extract useful information from unstructured textual data through the identification and exploration of interesting patterns. Text mining can represent flexible approaches to information management, research and analysis. Thus text mining can expand the fists of data mining to the ability to deal with textual materials.

Text Mining (TM) is the process of extracting novel, undetected and unstructured knowledge “hidden” in a large collection of unstructured text documents, using advanced technology. Text mining executes several processes, each one consisting of multiple phases. Text mining process consists of following steps: text collection, Text preprocessing, text mining algorithm and so on. It enables the knowledge worker to uncover relationships in a text Collection and to explore them in order to discover new knowledge. Text mining is particularly relevant today because of the enormous amount of knowledge that resides in text documents whether on the Internet, within the enterprise, elsewhere, or any combination of these sources. It is similar to Data Mining in that both deal with large amounts of data and aim at knowledge discovery within that data. Data Mining, however, focuses on discovery within already structured collections — in databases, data warehouses, and other corporate and external repositories. TM, by contrast, concentrates on the ever increasing flow of text data of all kinds that come to the attention of the knowledge worker. In just dealing with the Web, for example, the average knowledge worker needs tools that help him or her cut through the irrelevant or already known information and focuses their attention on the truly important or novel. Additional sources such as news feed, email, work product, letters and publications add to the need to intelligently mine textual data. To sum up with, TM is of critical value to any organization that needs to process text data and to make that information available across its organizational structure. Knowledge management consists of the initiatives and systems that sustain and support the storage, dissemination, assessment, application, refinement, and creation of relevant knowledge. This definition of knowledge management is adequate, but it relies on an understanding of the word "relevant". In this case it implies a strong tie to organizational goals and strategy, and it refers to knowledge that is considered useful for some purpose.

It involves the understanding of:

1. Where and in what forms knowledge exists;
2. How to make the right knowledge available to the right people;
3. What the organization needs to know;
4. How to best generate or acquire new relevant knowledge;
5. How to promote a culture conducive to learning, sharing, and knowledge creation;

6. How to manage all of these factors so as to enhance performance in light of the organization's strategic goals and short term opportunities and threats.

SOME BASIC TECHNOLOGIES FOR TEXT MINING

Information Retrieval

Information Retrieval (IR) is the first step in text mining.
Something like looking for ore from rock, the goal of Information Retrieval is to help users find documents that
Satisfy their information needs.

Computational Linguistics

Since text mining deals with the textual information based on natural language, we can easily get the critical conflict between natural language and the limited ability of computer to understand natural language. Computers lack the human's ability to easily distinguish and apply linguistic patterns to text and overcome obstacles handling such as slang, spelling variations and contextual meaning. However, human lack the computer's ability to process text in large volumes or at high speeds. Fortunately, there have been technological advances that have begun to close the obvious gap between natural languages and the ability of computer to process languages. The field of Computational Linguistics (also known as Natural
Language Processing) has produced technologies that teach computers natural languages so that they may analyze, understand, and even generate text.

Pattern Recognition

Pattern Recognition is the process of searching for Predefined sequences in text. Unlike pattern matching with
regular expressions in programming languages, this type of
pattern recognition works with words as well as morphological and syntactic properties.

Text mining Issues

Some of the Natural Language issues that should be considered during the text mining process are listed in table 1 and some of them are discussed in this paper.

Table1.Issues of Text Mining

Issue	Details
Stop List	Should We take into account stop words?
stemming	Should we reduce the words to their stem?
Noisy data	Should the text be clear of noise data?
Tagging	What about data annotation and/or part of speech characteristics?
Grammar/syntax	Should we make a syntactic or grammatical analysis? What about data dependency.
Tokenization	Should be tokenize by words or phrases and if so, how?
Text Representation	Which terms are important? Words pr phrase? Noun or objectives? Which text model should we use? What about word order, context and background knowledge?
Automated Learning	Should we use categorization? Which similarities measures should be applied?

Text Preprocessing

The preprocessing phase is crucial to the efficiency of the process, since according to the results in different domain areas and applications. Preprocessing can require as much as 80 per cent of the total effort. There are certain special aspects in the preprocessing of textual data. Text consists of words, special characters, and structural information. The preprocessing required depends heavily on the intended use of the results. Typically, the data is homogenized by replacing special characters and structural information e.g. SGML (Standard Generalized Markup Language) tags with symbols. Punctuation marks and structural information often need to be handled separately. Preprocessing may involve some amount of natural language analysis; Morphological analysis gives us detailed information of the data. We may use this analysis to generalize the data, e.g., by replacing words by their parts of speech, which allows us to identify constructs such as (preposition, noun) instead of combinations of specific words.

International Journal of Electronics Communication and Computer Engineering
Volume 2, Issue 2, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

II. RELATIONSHIP BETWEEN KNOWLEDGE MANAGEMENT, DATA MINING & TEXT MINING

Knowledge management, data mining, and text mining techniques have been widely used in many important applications in both scientific and business domains in recent years.

Knowledge management is the system and managerial approach to the gathering, management, use, analysis, sharing, and discovery of knowledge in an organization or a community in order to maximize performance. Although there is no universal definition of what constitutes knowledge, it is generally agreed there is a continuum of data, information, and knowledge. Data are mostly structured, factual, and oftentimes numeric, and reside in database management systems. Information is factual, but unstructured, and in many cases textual. Knowledge is inferential, abstract, and is needed to support decision making or hypothesis generation. The concept of knowledge has become prevalent in many disciplines and business practices. For example, information scientists consider taxonomies, subject headings, and classification schemes as representations of knowledge. Consulting firms also have been actively promoting practices and methodologies to capture corporate knowledge assets and organizational memory. In the biomedical context, knowledge management practices often need to leverage existing clinical decision support, information retrieval, and digital library techniques to capture and deliver tacit and explicit biomedical knowledge.

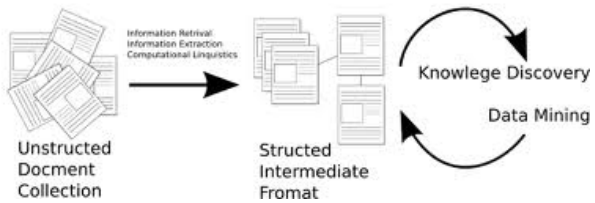

Figure:-1 Data Mining Process

III. WHY DATA PREPROCESSING?

In Text mining data preprocessing is required because data in the real world is dirty

***Incomplete*:** lacking attribute values, lacking certain attributes of interest, or containing only aggregate data

e.g., occupation=“ ”

***Noisy*:** containing errors or outliers

e.g., Salary=“-10”

***Inconsistent*:** containing discrepancies in codes or names

e.g., Age=“42” Birthday=“03/07/1997”

e.g., Was rating “1,2,3”, now rating “A, B, C”

e.g., discrepancy between duplicate records

Why Is Data Dirty?

A. Incomplete data may come from

B. “Not applicable” data value when collected

C.Different considerations between the time when the data was collected and when it is analyzed.

D.Human/hardware/software problems

- Noisy data (incorrect values) may come from
- Faulty data collection instruments
- Human or computer error at data entry
- Errors in data transmission

E.Inconsistent data may come from

- Different data sources
- Functional dependency violation (e.g., modify some linked data)

F.Duplicate records also need data cleaning

IV. WHY IS DATA PREPROCESSING IMPORTANT?

- No quality data, no quality mining results!
- Quality decisions must be based on quality data
- Data warehouse needs consistent integration of quality data
- Data extraction, cleaning, and transformation comprise the majority of the work of building a data warehouse.

V. PREPROCESSING STEPS FOR TEXT MINING

A. MORPHOLOGICAL ANALYSIS

The first step in text-preprocessing is the morphological analyses. Morphological, or Structural, Analysis is the process of breaking down morphologically complex words into their constituent morphemes (word meaning parts). For instance, the word *worker* is comprised of two meaning units, the base *work*, and the inclusion of *–er*, which conveys the meaning of an agent (person or thing) that does whatever is implied in the base. Thus, the worker is one who works. Morphology is a part of linguistics which is dealing with words. Therefore, it deals with the smallest, useful unit of a

document. One could say that characters are the smallest unit. Nonetheless, characters do not carry any valuable information for information retrieval. Firstly, information retrieval requires the words and the endings of a document.

It is divided into three subcategories:
1. Tokenization
2. Stemming and
3. Recognition of ending of records.

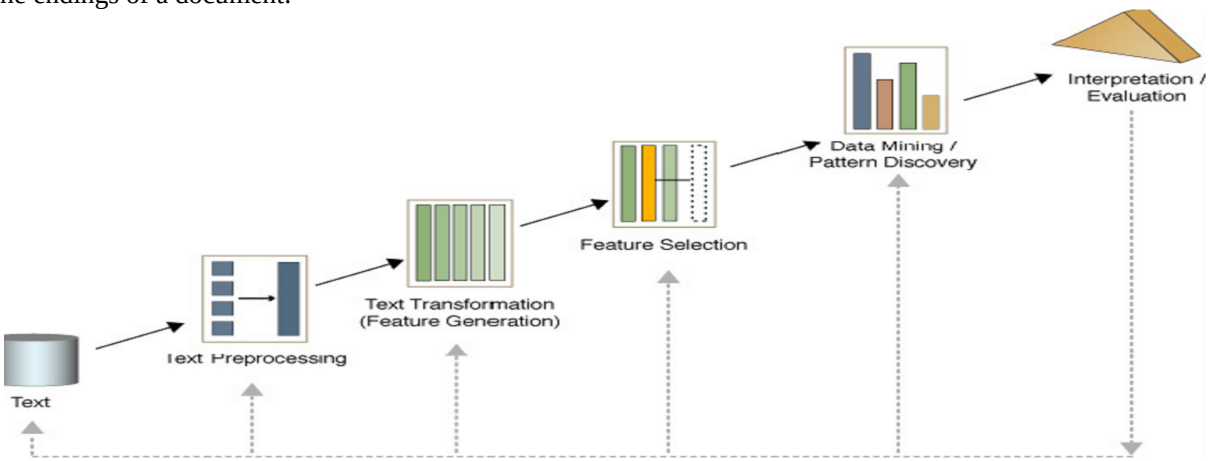

Figure:-2 Preprocessing Phase in Data Mining Process

1.-Tokenization

The first step of Morphological Analyses is the tokenization. The aim of the tokenization is the exploration of the words in a sentence. Textual data is only a block of characters at the beginning. All following processes in information Retrieval require the words of the data set. Hence, the requirement for a parser which processes the Tokenization of the documents.

For example-

Sentence “The quick brown fox jumps over the lazy dog”

```
<sentence>
  <word>The</word>
  <word>quick</word>
  <word>brown</word>
  <word>fox</word>
  <word>jumps</word>
  <word>over</word>
  <word>the</word>
  <word>lazy</word>
  <word>dog</word>
</sentence>
```

2.-*Stemming*-*Stemming and lemmatization*

The stemming process is an important pre-processing task before indexing input documents for text mining. A stemming algorithm is defined as a computational procedure that will reduce all the inflectional derivational variants of words to a common form called the stem. For grammatical reasons, documents are going to use different forms of a word, such as *organize*, *organizes*, and *organizing*. Additionally, there are families of derivationally related words with similar meanings, such as *democracy*, *democratic*, and *democratization*. In many situations, it seems as if it would be useful for a search for one of these words to return documents that contain another word in the set. The goal of both stemming and lemmatization is to reduce inflectional forms and sometimes derivationally related forms of a word to a common base form. For instance:

am, are, is $\Rightarrow$ be

car, cars, car's, cars' $\Rightarrow$ car

The result of this mapping of text will be something like:

The boy's cars are different colors $\Rightarrow$ *the boy car be differ color*

However, the two words differ in their flavor. *Stemming* usually refers to a crude heuristic process that chops off the ends of words in the hope of achieving this goal correctly most of the time, and often includes the removal of derivational affixes. *Lemmatization* usually refers to doing things properly with the use of a vocabulary and morphological analysis of words, normally aiming to remove inflectional endings only and to return the base or dictionary form of a word, which is known as the *lemma*. If confronted with the token *saw*, stemming might return just *s*, whereas lemmatization would attempt to return either *see* or *saw* depending on whether the use of the token was as a verb or a noun. The two may also differ in that stemming most

commonly collapses derivationally related words, whereas lemmatization commonly only collapses the different inflectional forms of a lemma. Linguistic processing for stemming or lemmatization is often done by an additional plug-in component to the indexing process, and a number of such components exist, both commercial and open-source.
Following a selection of suffixes and prefixes for removal during stemming:

suffixes: ly, ness, ion, ize, ant, ent , ic, al , ical, able, ance, ary, ate, ce, y, dom , ed, ee, eer, ence,ency, ery, ess, ful, hood, ible, icity, ify, ing, ish, ism, ist, istic, ity, ive, less, let, like, ment, ory, ty,ship, some, ure

prefixes: anti, bi, co, contra, counter, de, di, dis, en, extra, in, inter, intra, micro, mid, mini, multi,non, over, para, poly, post, pre, pro, re, semi, sub, super, supra, sur, trans, tri, ultra, un

3.-Recognition of Ending of Records

The most data reduction techniques in information retrieval use document vectors or term by document Matrixes. Using sentences for the comparison of similarities is the recognition of endings of records a inevitability. An algorithm searches for all endings of records like "." and "!". The result of this step are sentences which can be treated as an own unit. The text parser needs the ability to recognize if a punctuation
mark is a part of a word or sentence or used as an end of a sentence.

B. SYNTACTIC ANALYSIS

The purpose of syntactic analysis is to determine the structure of the input text. This structure consists of a hierarchy of *phrases*, the smallest of which are the *basic symbols* and the largest of which is the *sentence*. It can be described by a tree with one node for each phrase. Basic symbols are represented by leaf nodes and other phrases by interior nodes. The root of the tree represents the sentence. The knowledge about the syntax of sentences in natural language can improve the precision of the information retrieval systems. A sentence in English language contains nouns, verb, adverbs and other parts. Some parts are more valuable than others. For instance, queries for internet search engines often contain only nouns. The reason for this is the nouns make a sentence or document characteristic .Documents with a couple of similar terms (nouns) also has a similar topic. The syntactical analyses are divided into three subcategories:

- part of speech tagging,
- phrase recognition
- And parsing.

I. Part of Speech Tagging

The recognition of the elements of a sentence like nouns, verbs, adjectives, prepositions, etc. is realized through part of speech tagging (POS tagging). It is of importance for the IR process to have knowledge about the word type. It enables a correct decision about the following processing of that word. The identification of nouns is a necessity for a comparison of two or more documents. Unfortunately, a list of words with their part of speech is not possible as words can represent different parts of speech at different positions. POS tagging requires fast and powerful methods to provide a high precision and correctness of the results.

II. Phrase Recognition

The phrase recognition is closely related to part of speech tagging. The phrase recognition (PR) caters for the locating of groups of words, the phrases. PR is needed to keep relations between word groups which would lose their meaning if disjoined. Phrases are similar to compounds in linguistics, but are more complex. Phrases exist of different classes:

1. Prepositional phrase (e.g. in love)
2. Noun phrase (e.g. the queen of England)
3. Verb phrase (e.g. do business)
4. Adjectival phrase (e.g. large trousers)
5. Adverbial phrase (e.g. very quickly)

III. Parsing

Parsing in information retrieval is the process of structuring a sentence with the given grammar. Therefore, the sentences are fractionalized into the grammatical units. The structure of a sentence is represented in a tree structure. Each word of a sentence gets annotated with its type of grammar. This process allows the extraction of information from chosen syntactical units. Parsing is not the next step in the line of POS tagging and phrase recognition. The parsing approach can take over both parts. It is often used for phrase recognition. The tree structure is very useful for the recognition of groups of words.

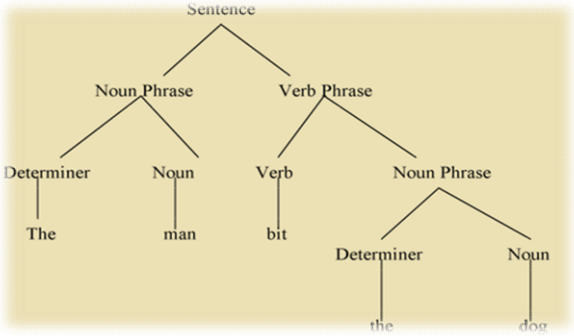

Figure:-3 Parsing Process

The figure shows the top-down parsing of a tree with the different grammar. The root element S describes the sentence itself. The acronym N stands for noun, V for verb, PN for pronoun, NP for noun phrase, VP for verb phrase. The elements of this tree are defined by phrase structure rules.

i.Sentence->Noun Phrase| Verb Phrase
ii.Noun Phrase->Pronoun
iii.Verb Phrase->Verb
iv.Verb Phrase>Verb| Noun Phrase

C. SEMANTICAL ANALYSES

Parsing only verifies that the program consists of tokens arranged in a syntactically valid combination. Now we'll move forward to *semantic analysis*, where we delve even deeper to check whether they form a sensible set of instructions in the programming language. Whereas any old noun phrase followed by some verb phrase makes a syntactically correct English sentence, a semantically correct one has subject-verb agreement, proper use of gender, and the components go together to express an idea that makes sense. For a program to be semantically valid, all variables, functions, classes, etc. must be properly defined, expressions and variables must be used in ways that respect the type system, access control must be respected, and so forth. Semantic analysis is the front end's penultimate phase and the compiler's last chance to weed out incorrect programs. We need to ensure the program is sound enough to carry on to Code generation. A large part of semantic analysis consists of tracking variable/function/type Declarations and type checking. In many languages, identifiers have to be declared before they're used. As the compiler encounters a new declaration, it records the type information assigned to that identifier. Then, as it continues examining the rest of the program, it verifies that the type of an identifier is respected in terms of the operations being performed. For example, the type of the right side expression of an assignment statement should match the type of the left side, and the left side needs to be a properly declared and assignable identifier. The parameters of a function should match the arguments of a function call in both number and type. The language may require that identifiers be unique, thereby forbidding two global declarations from sharing the same name. Arithmetic operands will need to be of numeric—perhaps even the exact same type (no automatic int-to-double conversion, for instance). These are examples of the things checked in the semantic analysis phase. Some semantic analysis might be done right in the middle of parsing. As a particular construct is recognized, say an addition expression, the parser action could check the two operands and verify they are of numeric type and compatible for this operation. In fact, in a one-pass compiler, the code is generated right then and there as well. In a compiler that runs in more than one pass (such as the one we are building for Decaf), the first pass digests the syntax and builds a parse tree representation of the program. A second pass traverses the tree to verify that the program respects all semantic rules as well. The single-pass strategy is typically more efficient, but multiple passes allow for better modularity and flexibility (i.e., can often order things arbitrarily in the source program).

D. DATA TRANSFORMATION (DIMENSION REDUCTION TECHNIQUES)

The transformation of the data into an efficient model for knowledge mining is the next step after the text analysis. The text is now ready to be processed. However, a transformation into a suitable model is required. The dimension of the data is far to high to process it in an acceptable speed. Therefore, a selection of the most common dimension reduction techniques may be used.

CONCLUSION

In this paper we collected the information about the preprocessing activities performed on the text before getting the knowledgeful information. Here we also identify the statement that why we have the requirements of the preprocessing. If we never preprocess the data in mining field then it has the lots of problem and noise which may generate problems while we perform next steps of mining on that data.

REFERENCES

[1] LIAO YUANYUAN, WANG JIANHU RESEARCH ON TEXT MINING, American Journal of Engineering and Technology Research Vol. 11, No.9, 2011
[2]Marwick, A.D. (2001) "Knowledge management technology". IBM Systems Journal, v. 40, n. 4, p.814-830.

[3] Muhamad Taufik Abdullah, Fatimah Ahmad, Ramlan Mahmod and Tengku Mohd Tengku Sembok Rules Frequency Order Stemmer for Malay Language, IJCSNS International Journal of Computer Science and Network Security, VOL.9 No.2, February 2009
[4]Morphological Analysis and Vocabulary Development: Critical Criteria Tom S. Bellomo Daytona State College, The Reading Matrix 2009 Volume 9, Number 1 , April 2009
[5]http://en.wikipedia.org/wiki/Lexical_analysis
[6] Semantic Analysis CS143 Handout 14Autumn 2007 October 24, 2007
[7] CSE 634 –Data Mining: Text Mining Munyaradzi Chiwara ,Mahmoud Al-Ayyoub Mohammad SajjadHossain, RajanGupta Professor Anita Wasilewska
[8]NASUKAWA, T. AND NAGANO, T. 2001. Text analysis and knowledge mining system. IBM Systems journal 40(4), 967–984.
[9] LUCAS, M. 1999/2000. Mining in textual mountains, an interview with Marti Hearst. Mappa Mundi Magazine, Trip-M, 005, 1–3. http:// mappa.mundi.net/trip-m/hearst/.
[10] Jochen Dorre, Peter Gersti, Roland Seiffert (1999), Text Mining: Finding Nuggets in Mountains of Textual Data, ACM KDD 1999 in San Diego, CA, USA.
[11] Ah-Hwee Tan, (1999), Text Mining: The state of art and the challenges, In proceedings, PAKDD'99 Workshop on Knowledge discovery from Advanced Databases (KDAD'99), Beijing, pp. 71-76, April 1999.
[12] Danial Tkach, (1998), Text Mining Technology Turning Information into Knowledge A white paper from IBM.
[13] Li Gao, Elizabeth Chang, and Song Han Powerful Tool to Expand Business Intelligence: Text Mining, World Academy of Science, Engineering and Technology 8 2005
[14] Valeriana G. Roncero, Myrian C. A. Costa, and Nelson F. F. Ebecken Using Stemming Algorithms on a Grid Environment, COPPE/Federal University of Rio de Janeiro
[15] Literature Review on Preprocessing for Text Mining by Keno Buss1, STRL, De Montfort University
[16] http://nlp.stanford.edu/IRbook/html/htmledition/stemming-and-lemmatization-1.html
[17] M. Natarajan Role of Text Mining in Information Extraction and Information Management, DESIDOC Bulletin of Information Technology , Vol. 25, No.4, July 2005, pp. 31-38
[18] Anna stavrianou, periklis andritsos and Nicolas Nicoloyannis "Overview and semantic issues of Text Mining",SIGMOD Record, September 2007(Vol. 36, No.3)
[19] J. Han and M. Kamber. Data Mining: Concepts and Techniques. Morgan Kaufmann, SanFrancisco, 2000.
[20]Helena Ahonen Oskari Heinonen, Mika Klemettinen

AUTHOR'S PROFILE

Mr. Sachin Tiwari
Lecturer in, Department of Computer applications at Samrat Ashok technological institute, Vidisha (M.P), India. He has published 05 papers in international and national conference proceedings. His research interest includes data mining and network security.

Mr. Krishna Kumar Mohbey
completed his Master degree at Rajiv Gandhi Technical University Bhopal (M.P.), India in 2009. At present he is a Lecturer in Computer Applications Department of Samrat Ashok Technological Institute, Vidisha (M.P.), India. His research interest includes data mining, Text Mining and Future scope of Internet. A. Inkeri Verkamo Applying Data Mining Techniques in Text Analysis , University of Helsinki_ Department of Computer Science University of Helsinki Finland.