2.4 SUMMARY

This chapter discusses the review of various authors that are proposed to describe various technologies used for text mining approaches. About this chapter considers clustering and rule mining approaches, semantic approaches in document mining and various text clustering evaluations’ are previewed by various authors .also reviews the problem statements.

CHAPTER 3

A GRAPH BASED SENTENCE LEVEL SEMANTIC LINKAGE WEIGHING MODEL FOR EFFICIENT TEXT CLUSTERING

3.1 INTRODUCTION

Text clustering is the way of organizing the natural language documents under different classes or groups based on set of similarity measures. The document set Ds may have N number of documents but has to be grouped under k number of clusters. To perform clustering, the textual content is retrieved from the document Di which is represented as T. The text
T has different parts of natural language processing like verb, noun and constants. It is considered about the verbs and nouns because the constants has no influence in the process of clustering. For example the sentence “Clustering similar Network Nodes is the process of grid computing”, has three nouns namely “grid” and “Node”, “Network” and there are two verbs namely “clustering” and “computing” but the constant “process” has no meaning or useful impact in the process of clustering.

From the above sentence mentioned, it can be noticed that it has terms belongs to three different domains like data mining, grid computing and networking. Such sentence set SD, has to be identified from the document and can be represented in form of graph, where each sentence can be formed as a graph Gi, and each graph has N number of nodes according to the terms present in the sentence Si. The graph based clustering approach is more useful where there is a necessary that the linkage between different domain documents has to be identified. So that by representing the sentences in form of graph, we can compute the similarity between the statements and can compute the overall similarity of the document towards each category.

Data mining is the process of identifying the related information from the large set of input domain. For example, there exist a large set of document as Ds, and the input query is the above discussed statement Si, then the ultimate aim is to identify the set of text or document which is related to the input query. To perform such retrieval of text documents, the clustering has to be done in more efficient manner, there are many clustering algorithm has been discussed and available earlier like term based clustering. In term based clustering, the document similarity is computed only based on the term level frequency and in the concept based approaches, the methods looks for the concept weight or concept to be discussed in the text document. It is not necessary that the document to be discuss about a single concept or single
domain, there may be different content could be discussed, so that identifying the class of document is the challenging issue.

In earlier days the clustering has been performed by maintaining the term sets belongs to different category in a taxonomy and based on the terms present in the taxonomy the class of document has been identified by computing different similarity measures like Euclidean distance by K-Means clustering. Similarly, the popular Fuzzy approach has clustered the document by computing the measures and maintaining range of values for each category of documents. Hierarchical clustering approaches also has been used in this problem of text clustering, which identifies the root class and then looks for the next level to assign the document class. Graph based approaches construct tree from the terms of taxonomy and computes similarity from the input term set to identify the class of the document. All these methods are based on the term values and term set extracted from the input text document which has poor clustering accuracy because, if we look at the previously mentioned statement, we can identify the terms belongs to three different category and while clustering such statements of document, the methods suffers with the accuracy and produces false indexing.

To overcome the issues of clustering, the semantic ontology can be used which has different class and for each class there exist different attributes can be considered as terms, and for each property or attribute there exist a relation with other terms or attributes. By using semantic ontology, the clustering of text document can be performed in efficient manner by identifying an efficient similarity measure. The sentence level semantic weighting is such one, which measures the similarity of documents in efficient manner and will be discussed in detail in the coming sections.
3.2 PROBLEM DEFINITION

There are many clustering techniques has been discussed for the text documents and few of them will be discussed here in detail.

- Documents are represented by keywords. These words are grouped into clusters, based on efficient similarity computations.

- Documents with related words are grouped into clusters. The clusters are characterized by similarity equations, graph based similarity measures and Gaussian parameters.

- As words are been given into the system, clusters would be generated automatically. The hybrid mechanism works with membership algorithms to identify documents that match with one another and can be grouped into clusters. The method works to find the real distribution of words in the text documents.

- The words in the feature vector of a document set are grouped into clusters, based on similarity test. Words that are similar to each other are grouped into the same cluster.

- Each cluster is characterized by a membership function with statistical mean and deviation. When all the words have been fed in, a desired number of clusters are formed automatically.

- The extracted feature, corresponding to a cluster, is a weighted combination of the words contained in the cluster. By this algorithm, the derived membership functions match closely with and describe properly the real distribution of the training data. Besides, the user need not specify the number of extracted
features in advance, and trial-and-error for determining the appropriate number of extracted features can then be avoided.

- The degree of mutual auxiliary is based on the ratio of long texts and short texts in a corpus. Statistical model for topically segmented documents, presents a generative model that exploits a given decomposition of documents in smaller text blocks which are topically cohesive (segments).

  - A new variable is introduced to model the within-document segments: using this variable at document-level, word generation is related not only to the topics but also to the segments, while the topic latent variable is directly associated to the segments, rather than to the document as a whole.

- Semantic Evaluation of Text Clustering investigates the problem of quality analysis of clustering results using semantic annotations given by experts.

All the above discussed approaches produce poor results with multi domain documents and produce more false indexing.

### 3.3 GRAPH BASED SENTENCE LEVEL SEMANTIC LINKAGE WEIGHTING

The graph based sentence level semantic linkage weighting model has been presented here which generates semantic graph for each of the sentence being identified. With the graph being generated, we compute the sentence level semantic linkage weight for each of the graph towards all the class considered. Finally a top weighted class is identified as the resultant class.
Figure 3.1 Sentence Level Semantic Linkage Clustering Architecture

Figure 3.1 shows the architecture diagram of the proposed graph-based sentence level semantic linkage weight model clustering and its functional components. Each of the functional components will be explained in detail in this section.

3.3.1 Sentence Semantic Graph Generation

At this stage, the input document is processed to extract the textual content. The extracted text is split into number of sentence by splitting by punctuation marks. Each sentence is split into number of distinct terms by splitting by space character. The extracted term are added to the term set related to the sentence. For each term of the sentence, the method performs stop word removal, stemming and tagging process.

From identified nouns, a distinct semantic graph is generated and for each term of the sentence a node is added to the graph. Now the semantic ontology is loaded into the system, and looked up for the label in the ontology.
and for each of the term, the property and the relation with other terms are identified.

If there is any relation present between the terms then a link is generated between them and for each node is identified the semantic terms related with that and specify interior and exterior links. The generated graph will be used in the later stage of clustering.

**Algorithm 3.1**

**Input** : Document Set $D_s$, Semantic Ontology Set $S_oS$.

**Output** : Semantic Graph Set $S_GS$.

---

Start

1. Initialize Term Set $T_{ss}$
2. Initialize Graph Set $G_s$.
3. For each document $D_i$ from $D_s$
   - Text $T = $ Extract Text from $D_i$.
   - Sentence set $S_s = (\sum_{n=1}^{\text{size}(D_s)} \text{Text}_n \text{Di}) \times \text{Splitby(.)}$
4. For each sentence $S_i$ from $S_S$
   - Generate Graph $G_i$.
   - Term Set $T_i = \sum \text{Terms}@S_i$
   - For each term $T_n$ from $T_i$
     - If $T_n \in \text{StopwordSet}$ then $T_i = T_i \setminus T_n$
     - Else
       - Perform Stemming
     - Apply Part of speech tagging.
   - End
5. For each term $T_k$ from $T_i$
   - Create Node $N_i$.
   - Add node to $G_i$.
6. $G_i = \sum (\text{Nodes}_G_i) + N_i$.
7. Read Domain ontology $D_o$.
8. For each domain $D_i$ from $D_o$
For each term $T_k$ from $T_i$
    If $D_i \in$ then
        Identify relation it has with other concepts.
        Relation Set $R_s = \sum (\text{Concepts} \in D_i) + C_i$.
        Add relations to the Node $N_i$.
    End
End
End
Stop

The above discussed algorithm performs preprocessing of the text documents and generates semantic graph for each of the sentence being identified from the document.

3.3.2 Sentence Semantic Link Weight Computation

At this stage, we compute the sentence level semantic link weight for each of the sentence towards each of the cluster being considered. For each graph, which represents a sentence in the document, the semantic link weight is computed. The method computes the number of relations it has and number of links a graph has. Then the method identifies the set of interior and exterior relations it has with the concepts of different domain. Based on all these measures of each sentence, we compute the semantic linkage weight for each of the sentence. Finally a cumulative weight will be computed for each of the class, and a single domain or class will be selected as the result which has more weight.

Algorithm 3.2

<table>
<thead>
<tr>
<th>Input</th>
<th>Semantic Graph Set $Sgs.$, Semantic Ontology $So.$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output</td>
<td>Class Name.</td>
</tr>
</tbody>
</table>
Start
    For each graph $G_i$ from $Sgs$
Stop
For each domain Di from So

Compute Number of relations it has.

\[ Nr = \sum \text{Relations} \in G_i \]

Compute Number of incoming links.

\[ NIL = \sum \text{Links}(G_i) < \sum G_k(Sgs) \]

Compute the value of interior links.

\[ ILV = \sum \text{Concept(Links}(G_i) \in Di \]

Compute the value of exterior links value.

\[ ELV = \sum \text{Concept(Links}(G_i)) \in \sum \text{Concept(Dj)} \neq Di \]

Compute semantic linkage weight Slw.

\[ Slw = \left( \frac{ILV}{Nr} \times \frac{ELV}{Nr} \right) + NIL \]

Add to weight set \( Ws = \sum Ws(Di) + Slw \)

End

End

For each domain Di from So

Compute mean value of semantic linkage weight.

\[ \text{Mean Slw} = \frac{\sum Slw(Ws(Di))}{\text{size}(Di)} \]

End

Choose the most valued Domain Di.

\[ \text{Class C} = \max(\text{mean}) \]

Stop.

The above discussed algorithm computes the sentence level semantic link weight, which is used to perform clustering the text documents.

### 3.3.3 Document Clustering and Results

The clustering of documents is performed by the proposed approach using the above discussed two algorithms, initially whether it is a clustering process or a retrieval process, the input content is passed through the first algorithm which generates the semantic graph. Then once the graph has been generated, them for clustering the document, the method computes the sentence
level semantic linkage weight and based on that the method assigns a class to
the document and the document will be indexed to the concern category. In
case of retrieval at the training phase, the method stores the computed sentence
level semantic linkage weight and the method computes the same for the input
query and identifies the domain of query. Once the category of the query has
been identified then using the pre-computed linkage weight, the indexed
documents are ranked and returned as a result to the user.

**Algorithm 3.3**

<table>
<thead>
<tr>
<th>Input : Document/Query</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output : Variant</td>
</tr>
</tbody>
</table>

**Start**

```plaintext
If Input==Document then
    Generate semantic graph.
    Compute sentence level semantic linkage weight slw.
    Identify the class of document and index the document to the cluster.
Else
    Generate semantic graph.
    Compute sentence level semantic linkage weight slw.
    Identify the class of document.
    Retrieve the trained semantic linkage weight.
    Sort the documents according to the Slw.
    Choose most popular documents as result and return.
End
```

**End**

The above discussed algorithm performs clustering of documents
according to the graph based sentence level semantic linkage weight model and
produced more efficient clusters.
3.4 SIMULATED RESULTS

In this section, the proposed Method produced a text mining in document processing and the results are displayed below. Figure 3.2 shows a graph based sentence level semantic linkage weighing model for efficient text clustering. Also, the results of various qualities of service parameters are discussed.

![Graph Based Sentence Level Semantic Linkage Weighting](image)

**Figure 3.2 Input process in document analyse model**

Figure 3.2, shows the result of graph based sentence level semantic linkage weighing model estimation produced by the proposed method. The result shows that the input of text clustering is classified from document dataset.
Figure 3.3 Analysis weighting factor

Figure 3.3, shows the result of graph based sentence level semantic linkage weighing model estimation produced by the proposed method. The result shows that the false indexing ratio text clustering is classified from document dataset.
Figure 3.4 Probabilistic factor for clustering

Figure 3.4, shows the result of graph based sentence level semantic linkage weighing model estimation produced by the proposed method. The result shows that the probabilistic factor is processed from document dataset.

3.5 SUMMARY

In this chapter we discussed and evaluated an Graph Orient Sentence Based Semantic Linkage Weighting (GSSWM) algorithm has been presented to cluster text documents. The method extract the text content from the document and split them into terms, performs stop word removal and stemming operation. Then the terms are applied with part of speech tagging which yields the pure nouns and verbs. Using these terms, the method generates semantic
graph for each of the sentence from the document. For each graph, the method computes sentence level semantic linkage weight towards each class being considered. Finally we compute the mean value of all semantic linkage weights of the sentences from the document towards each domain available. A top valued class is selected and the document is indexed into the selected class. The proposed approach performs 95.7% clustering in efficient manner and reduces the false indexing ratio largely with less time complexity.