CHAPTER 6

SENTENCE BASED TEXT CLUSTERING
USING THE WORDNET

6.1 INTRODUCTION

Text Clustering is an unsupervised method that is used to group similar text documents. In text clustering, the documents are first pre-processed, and features are extracted using various techniques. The retrieved features will then be clustered. There are various methods that are available for both feature extraction and text clustering. Usually, in the clustering process, the key terms are extracted in the form of words or phrases. In this work, first the features are extracted in the form of sentences using various formulae. Then, the sentences are weighed and a certain ratio of sentences with the highest weightage is extracted from each document. Then the summarized sentences are clustered semantically using the WordNet. Wordnet is an online dictionary, and using that, the semantics between the sentences are found. Then, based on the similarity between the sentences, the documents are clustered. Semantics is an important factor that must be considered in the clustering process, and in this the work semantic relationship between sentences is taken into consideration. The results are tested using Reuters-21578 dataset.
6.2 SYSTEM ARCHITECTURE

![System Architecture Diagram]

Figure 6.1 System Architecture
Figure 6.1 shows the architecture of the entire system. The input documents are first pre-processed and the sentences are ranked. The sentences are ranked based on various features, like sentence position, similarity between sentences, number of proper nouns, verbs, adverbs, adjectives, Positive and negative keywords, etc. A weight value is calculated for each sentence based on these features, and then they are ranked. The topmost ranking sentences are extracted and clustered using the WordNet.

6.3 SENTENCE EXTRACTION

In this work, the entire process is divided into two phases, namely, sentence extraction, and clustering using the WordNet. In the first phase of the work, feature reduction is done based on sentences. The important sentences are extracted from each document based on their weightage. To calculate the weightage various parameters like sentence position, number of words matching with the title, number of verbs, nouns, adverbs, adjectives present in the sentence etc are taken into consideration. The various formulae that are used are given in Equations (6.1) to (6.9) as follows:

**Step1: Position of the sentence**

The position of a sentence determines its importance. The first sentence of a document may be more important than the last sentence. So, taking this into consideration, the position of a sentence is calculated as follows:

\[
W_1 = \frac{\text{sentence \_ position}}{\text{Total \_ number \_ of \_ sentences \_ in \_ the \_ document}}
\]  

(6.1)

Consider a document containing 8 sentences; then the weight of the first sentence is calculated as 1%8=1, 2%8=2 for the second statement, and finally, 8%8=0. The Modulo operation is used for this.
Step 2: Positive keywords in the sentence

Positive keywords are the words that give good meaning to the document. A list of positive keywords is used, and the words in the document are compared with those in the list. The weight of a sentence based on the positive keywords is given by the formula.

\[
W_2 = \frac{\text{Total number of positive keywords in the sentence}}{\text{Total number of words in the sentence}}
\]  
(6.2)

Step 3: Find the Negative Keyword in the sentence

Similarly, the negative keyword list is also identified. The words in the document are compared with those negative keywords in the list and the weight is calculated for each sentence. The formula is given as follows:

\[
W_3 = \frac{\text{Total number of negative keywords in the sentence}}{\text{Total number of words in the sentence}}
\]  
(6.3)

Step 4: Find the similarity between the sentence using the keyword match in all sentences

The similarity between the sentences are calculated, based on the keywords that match in both the sentences. The formula for calculating the similarity between the sentences is given by

\[
W_4 = \frac{\text{Keyword in sentence}}{\text{\sim keyword in other sentences}} \cdot \frac{\text{Total number of words in summary}}{\text{Total number of words in summa}}
\]  
(6.4)

Step 5: Find the Keywords that match with the title

The keywords in some sentences will match with the title. Those sentences will carry extra information about the document. Such sentences are
identified, and the sentences are given a weightage value. The formula is given as follows:

\[ W_5 = \frac{\text{keyword match in sentence} \cap \text{Keyword match in title}}{\text{Total keywords in both}} \]  

(6.5)

**Step 6: Finding the proper noun in the sentence**

Sentences having proper nouns will carry more information than other sentences. Taking this into consideration, the proper nouns are identified, using the Stanford POS tagger and the weight of those sentences is calculated using the following formula. It is given as the ratio of the total number of proper nouns in a sentence to the sentence length.

\[ W_6 = \frac{\text{Total number of proper nouns in a sentence}}{\text{length of a sentence}} \]  

(6.6)

**Step 7: Finding the verbs in the sentence**

A verb is a word that conveys an action, an occurrence, or a state of being. Based on this, sentences containing verbs are given extra weightage. Verbs are identified using the Stanford POS tagger, and the weight is calculated based on the ratio of the total number of verbs in the sentence to the sentence length.

\[ W_7 = \frac{\text{Total number of verbs in a sentence}}{\text{length of a sentence}} \]  

(6.7)

**Step 8: Finding the adverbs in the sentence**

An adverb is a word that gives an extra meaning of a verb, or adjective. The sentence weight in accordance with the number of adverbs is calculated, as a ratio of the number of adverbs to the sentence length.
$W_s = \frac{\text{Total number of adverbs in a sentence}}{\text{Length of a sentence}}$ 

(6.8)

**Step 9: Finding the adjective in the sentence**

An adjective is a ‘describing’ word which qualifies a noun. The formula is given as the ratio of the total number of adjectives in a sentence to the sentence length.

$W_s = \frac{\text{Total number of adjectives in a sentence}}{\text{Length of a sentence}}$ 

(6.9)

Based on all these formulae, the weight is calculated for each sentence. Then the weights are summed up for each sentence, and the sentences are arranged in the descending order. Then, based on the size of the file, a certain number of the sentences are extracted from each document. For example, if the document contains 10 sentences, then 80% of the document is extracted as important sentences, which means 8 sentences are extracted. If the document contains 5 sentences, then 80% of this implies that 4 sentences are extracted. Therefore, the number of sentences extracted in each document is not fixed. The advantage of doing this is that, the ratio of the sentences being extracted, depends on the document size rather than a threshold value. If the number of sentences to be extracted is fixed, then if the document size is small then it will lead to problems. The extracted sentences are termed as important sentences for each document. Then, they are given as input to the clustering module.

### 6.4 WordNet Based Clustering Algorithm

In this phase, the extracted sentences are clustered using the WordNet. Usually, the text documents are not clustered based on semantics, which is an important factor. There are various problems like synonymy, polysemy etc., which should be considered while dealing with text
documents. Taking this into consideration, the text documents are clustered, based on semantics using the WordNet.

Some words may be similar in the documents, but when the meaning is considered they will mean differently. For example, consider the word, bank. If we match the words exactly between documents, they will be similar and will be grouped under one category. But if we consider the meaning of the word, it will be different based on the context in which it is used. So, the clustering performance will degrade.

So, in order to overcome all these disadvantages, in this work clustering is done based on the meaning of the documents. The meaning of the documents is found using the WordNet, which is an online dictionary. Using this, the words which belong to the same synset are found, and grouped together. If more than 80% of the sentences from two documents belong to the same synset, then they are clustered. The same procedure is repeated for the entire document set, and they are clustered.

6.5 SAMPLE OUTPUT

![Text Summarization Using Ten Merits](image)

Figure 6.2 Reading XML file
Figure 6.3 Choosing the XML File

Figure 6.4 Parsed output
Figure 6.5 Text Summarization

Figure 6.6 File Selection for text summarization

Sentence position 1: dry period mean temporao late year

dry
period
mean
temporao
late
year
Length of sentence 1 :7
Reading POS tagger model from stanford-postagger/models/english-bidirectional-distsim.tagger ... each count : 7
done [1.4 sec].
Sentences : arriv JJ week_NN end_NN februari_NN 22 155 221 CD
bag_NN 60_CD kilo_NNS make_VBP cumul JJ total JJ season_NN 5_CD
Total nouns : 6
Total verbs : 1
Total Adjectives : 3
Total Adverbs : 0

Figure 6.7 Sample Summarization Output
shower, shower bath
shower, rain shower
shower, cascade
exhibitor, exhibitioner, shower
shower
lavish, shower
shower
shower, shower down
shower
week, hebdonad
week, calendar week
workweek, week
cocoa, chocolate, hot chocolate, drinking chocolate
cocoa
zone
zone, geographical zone
zone
zone, zona
zone, district
partition, zone
drought, drouth
drought, drouth
prospect, chance
6.6 TESTING

The performance of clusters is evaluated using Micro Measures like Micro Averaged Precision (MicroP), Micro Averaged Recall (MicroR), and
Micro Averaged F Measure (MicroF). Reuters-21578 dataset is used for the testing purpose. The formulae are given below in Equations (6.10) to (6.12):

\[
\text{MicroP} = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n} (TP_i + FP_i)} \tag{6.10}
\]

\[
\text{MicroR} = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n} (TP_i + FN_i)} \tag{6.11}
\]

\[
\text{MicroF} = \frac{2 \times \text{MicroP} \times \text{MicroR}}{\text{MicroP} + \text{MicroR}} \tag{6.12}
\]

6.6.1 REUTERS 21578

The Reuter-21578 has been widely used for testing clustering algorithms. Reuters-21578 is an experimental data collection that appeared on Reuters newswire of the year 1987. The dataset was obtained from http://kdd.ics.uci.edu/databases/reuters21578/reuters21578.

<table>
<thead>
<tr>
<th>No of Documents</th>
<th>MLCL Clustering</th>
<th>WordNet based</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>94.32</td>
<td>67.11</td>
</tr>
<tr>
<td>50</td>
<td>98.65</td>
<td>615.7</td>
</tr>
<tr>
<td>75</td>
<td>99.2</td>
<td>1769.6</td>
</tr>
<tr>
<td>100</td>
<td>99.08</td>
<td>4585</td>
</tr>
</tbody>
</table>
Figure 6.14 MicroR Values

Table 6.2 MicroP Values

<table>
<thead>
<tr>
<th>No of Documents</th>
<th>MLCL Clustering</th>
<th>WordNet based</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>82.34</td>
<td>67.11</td>
</tr>
<tr>
<td>50</td>
<td>93.95</td>
<td>596.9</td>
</tr>
<tr>
<td>75</td>
<td>90.40</td>
<td>1695.1</td>
</tr>
<tr>
<td>100</td>
<td>92.09</td>
<td>3141.8</td>
</tr>
</tbody>
</table>

Figure 6.15 MicroP Values
Table 6.3 MicroF values

<table>
<thead>
<tr>
<th>No of Documents</th>
<th>MLCL Clustering</th>
<th>WordNet based</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>87.92</td>
<td>67.11</td>
</tr>
<tr>
<td>50</td>
<td>96.24</td>
<td>604.6</td>
</tr>
<tr>
<td>75</td>
<td>94.59</td>
<td>1720</td>
</tr>
<tr>
<td>100</td>
<td>95.45</td>
<td>3237.4</td>
</tr>
</tbody>
</table>

Figure 6.16 MicroF Values

The Tables 6.1, 6.2 and 6.3 has shown that the WordNet based method outperforms Must Link and Cannot Link algorithm. Figures 6.14, 6.15, and 6.16 show the diagrammatic representation of the same.

6.7 SUMMARY

The clustering process discussed in this chapter is based on the semantics on the document. In this work, first the documents are pre-processed and feature extraction is done based on sentences. The sentences are weighed based on various metrics and then arranged in descending order based on the weight value. From that, a certain number of sentences are
extracted for each document. Then, based on those sentences the documents are clustered. The clustering process is done based on semantics because it is an important factor to be considered in text documents. The WordNet is used to find the semantics between documents. The documents that are more similar to one another in meaning are clustered. The process is repeated for the entire document set and the results are evaluated. The results have shown a better performance than the other methods. Future work on this work may include considering polysemy, hyponymy, etc., for clustering which may provide more improved results.