Chapter 3 Keyword Enrichment, Overlapped Semantic and Belongingness

In Chapter 2, we presented a review of statistical, context based and collaborative approaches for Document Classification. We re-visited some recently used context oriented features and approaches employed by researchers. We analyzed the advantages and potential of context and mixed approaches vis-a-vis statistical approaches.

In this chapter, we present knowledge based supervised learning document classification technique that extracts categories’ ontological information to generate a compact set of keywords. It calculates Belongingness, a measure of a document’s affinity towards a category, of the unlabelled test documents to each category using Keyword Strength of keywords and Term Frequencies of tokens. The document is labeled to the category for which it obtains highest degree of Belongingness. Further the algorithm enriches each category’s keyword list by choosing semantically related tokens from well classified documents.

Section 3.1 introduces the main approaches used in this work. Section 3.2 presents the detailed scheme for semantic Keyword Enrichment, Overlapped Semantics and Belongingness. Section 3.3 illustrates experimental results and observations which show the efficiency of our approach. Section 3.4 compares the quantitative and qualitative characteristics of the proposed DC scheme with prior work. Section 3.5 investigates the strengths and limitations of the DC scheme.

3.1) Introduction

The proposed Document Classification (DC) system is called “knowledge-based” because we use two sources of existing knowledge, viz WordNet ontology and Wikipedia encyclopedia to determine the semantic content of categories and documents and map them. Figure 3.1 depicts the overall architecture of the knowledge based DC scheme proposed in this chapter. The main features of the scheme are highlighted below.
**Lexical Semantics as keywords:** We initiate the process of classification by populating each category with related concept terms that define the feature-space of the category. In [16], the authors demonstrated how well-organized background knowledge in the form of a simple ontology can improve document classification result significantly. WordNet is designed primarily as a lexical database and it can be used as ontology [69]. Many words have several meanings when used in different contexts. They are called polysemous words.

WordNet expresses the meaning of each sense of a base word with its Synonym set. For each non-trivial token which is present in a document and for each category name, their lexically related terms can be found in WordNet starting with their relevant senses and
transitively following different relation types. Lexical cohesion refers to a range of
textual cohesion that allows the use of similar meaning words on Synonyms,
generalization the concept on Hypernyms, specialized versions of a concept called
Hyponyms or constituent parts of an object called Meronyms. Terms that share a
common Hypernym are called Coordinate terms. The set of lexically related terms of a
base word are called Lexical Semantics. The concern about generating a suitable list of
keywords for a given category can be addressed by extracting all Lexical Semantics of
the category from the WordNet.

- **Overlapped Semantics**: Our next concern is to extract meaningful words from a
document. For accomplishing this, we use a modified version of the concept of
Overlapping Semantics (OS) inspired from [35]. The fundamental premise of OS is that if
two tokens in a document have a common list of Lexical Semantics, then the
corresponding features represent the document’s meaningful intent greatly. Therefore,
such tokens can be retained and other tokens can be pruned as their presence is incidental.
This reduces the feature space of each document.

- **Belongingness based categorization**: The tokens of the reduced documents are
matched with keywords of each category. Based on the Keyword Strengths and term
frequencies of matched tokens. The Belongingness of a document to each category is
calculated. Finally, the document is ascribed to the category with highest Belongingness
value.

- **Wikipedia aided Keyword Enrichment**: With the help of the Wikipedia, the algorithm
enriches category-keyword lists by using the tokens of classified documents. Wikipedia is
a free, open content online encyclopaedia created through the collaborative effort of a
community of users known as Wikipedians [106]. Wikipedia's articles are peppered with
hundreds of millions of links. It has functions to handle many fundamental tasks in
computational linguistics, such as Word Sense Disambiguation (WSD), Information
Retrieval (IR), word and text clustering, and error correction [107]. We tap these
powerful features of Wikipedia to augment more keywords and enhance the feature space
of each category. We now detail our proposal work in subsequent sections.
3.2) Framework for knowledge based DC

The stepwise detailed description of the knowledge based DC system now follows.

1) Acquire a keyword-list per category: We initiate our algorithm by collecting lexically related keywords for each category. We input category name to the WordNet. First synonyms are extracted. Then, Hypernyms are collected up to the first level only. Next Hyponyms at the immediate lower level are located. Restricting generalized and specific terms to only single level helps retain the cohesive strength between terms. Meronyms and Coordinate terms are next extracted. At the end of this process, the system is armed with an initial set of keywords $\textbf{CK}$ that explicitly represent the semantic domain of the category.

2) Pre-processing document: The documents to be classified need to be pre-processed. Pre-processing includes the following steps.

a) Stop-word removal: All the stop words, i.e. words that appear frequently but do not affect the context are removed from the document. Examples of such words include ‘a’, ‘an’, ‘and’, ‘the’, ‘that’, ‘it’, ‘he’, ‘she’ etc. The stop word list contains 1205 words, which do not play any significant role in classification [108].

b) Stemming: The resulting set of tokens is replaced by their base forms to avoid treating different forms of the same word as different features. This reduces the size of the feature set.

However, instead of employing conventional porter algorithm [109], we modify it by integrating exceptions used in WordNet. These exceptions stem the words without robbing their semantic significance. For example, consider the word ‘relativity’. The conventional porter algorithm would reduce the term to “relative” which destroy one possible context. However, with modified algorithm the context is safe now.
c) **Term Weighting:** For each token $w$ in the document’s token set, its number of occurrence is computed. This is called as Term Frequency $TF_w$.

3) **Reducing dimensionality of the document by Overlapped Semantics:** The pseudocode in *Figure 3.2* explains the process of Overlapped Semantics (OS).

<table>
<thead>
<tr>
<th>Algorithm: Overlapping_Semantics(.)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input:</strong> 1. A set of tokens in a pre-processed document $D$.</td>
</tr>
<tr>
<td>2. Overlapping Threshold $\pi_{os}$.</td>
</tr>
<tr>
<td><strong>Output:</strong> A set of semantically related tokens in new document $D'$.</td>
</tr>
</tbody>
</table>

1. Begin
2. {
3. \hspace{1em} $D' = \phi$
4. \hspace{1em} For each token $w \in D$
5. \hspace{2em} Obtain Lexical Semantic Set $\text{LexS}_w$ from WordNet;
6. \hspace{1em} For each token $w \in D$
7. \hspace{2em} \hspace{1em} For each other token $z \neq w$, $z \in D$
8. \hspace{3em} \hspace{1em} If $|\text{LexS}_w \cap \text{LexS}_z| \geq \pi_{os}$ then
9. \hspace{3em} \hspace{3em} $D' = D' \cup \{w, z\}$
10. \hspace{1em} Delete $D$
11. \hspace{1em} Return $D'$
12. End;

*Figure 3.2* Pseudocode to reduce the dimensionality of the document.

By OS, the algorithm extracts strongly related words of the document, which are important for classification. It inputs each token $w$ of a document to the WordNet to obtain its Lexical Semantics *viz.*, *Synonyms, Hypernyms, Hyponyms, Meronyms* and *Coordinate terms*. This is termed as its Lexical-semantics set $\text{LexS}_w$. Now, starting with first token, the system takes each token $w$ and finds the intersection of $\text{LexS}_w$.
with those of all other tokens in the document. If the intersection exceeds a pre-
decided threshold $\pi_0$, the token is retained. Otherwise it is dropped from the
document.

The tokens thus collected from the pre-processed document signify the document’s
meaning to a greater extent and reduce the dimensionality of document
representation. Let $N_w$ be the final number of tokens representing the document.

4) Generating Keyword-Strength (KS) matrix: We now construct a
Keyword-Strength matrix for each document. In this matrix the number of rows is
equal to the number of tokens in the document which have been derived by using
overlapping semantics. The number of columns is equal to the number of categories.

The elements of this matrix denote the Keyword Strength ($KS$) of a keyword for the
corresponding category. The presence or absence of the token is checked in the
keyword lists of the categories. If the token is present in any of the Category-
Keyword lists, it is a keyword for that category.

Let $P(w,C_k)$ be a Boolean function that defines the presence (=1) or absence (=0) of a
token $w$ in the category $C_k$. The Keyword Strength $KS$ of a matched token
(equivalently keyword) $w$ to a given category $C_k$ is obtained by the formula below.

$$KS_w, k = \sum_{j=1}^{n \text{cat}} \frac{P(w,C_k)}{P(w,C_j)}$$  \hspace{1cm} 3.1

The Keyword Strengths ($KS$) are pre-calculated and kept stored in the Keyword
Category matrix.

5) Computation of Belongingness metric: For each column in the matrix, a
category Belongingness metric $B_k$ that reflects the degree of Belongingness of a test
document to a given category $C_k$, is computed using the frequencies of the tokens in
the document and their corresponding Keyword Strength. This metric, given in equation 3.2, is computed for all the columns, *i.e.* for the predefined categories.

\[ B_k = \frac{\sum_{w=1}^{Nw} TF_w \times KS_{w,k}}{\sum_{w=1}^{Nw} TF_w} \quad 3.2 \]

6) **Assigning a document to category:** The *Belongingness* measures obtained for all the columns for a given document are compared. The document is classified to that category for which this metric has the maximum value.

\[ \text{Cat}(D_i) = C_k : k = \arg \max_{j=1 \ldots N_{\text{cat}}} (B_j) \quad 3.3 \]

Where \( \text{Cat}(D_i) \) be a function that returns the category of a document \( D_i \) and \( N_{\text{cat}} \) is the total number of categories.

7) **Keyword Enrichment:** The pseudocode in Figure 3.3 explains the process of Keyword Enrichment (KE).

The KE is done by utilizing the tokens of newly classified documents and Wikipedia. The motivation of using Wikipedia is its ability to yield a rich set of semantically related words of a given topic. This way, terms which are conceptually related to a category are also added to the prior created keyword list \( CK \) of Lexical Semantics of that category obtained from WordNet, thus yielding an enriched list \( CK' \).

The tokens with membership metric of zero (line 5) in the document classified to a category undergo training to find their relevance with the respective category. The first-level hyperlink lists of these tokens are extracted from the Wikipedia (line 6, 7, 8). These lists are intersected with the respective category keyword list (line 9). If the intersection is greater than a specific threshold \( \pi_{KE} \), those keywords are added in the category list (line 10), thus resulting in an enriched keyword list.
Algorithm: KE(.)

Input: 1) A classified document D
        2) The relevant category $k = \text{Cat}(D)$
        3) Its keyword list $CK_k$.

Output: Enriched category keyword list $CK_k'$.

Begin
{ Initialize $CK_k' = CK_k$,
  For each token $w \in d$ {
    If ($P(w, C_k) = 0$) then {
      Ping $w$ to Wikipedia to get its article page \text{wiki\_page}(w).
      Create list of Hyperlinks \text{Hyl}(w) from the article page \text{wiki\_page}(w).
      If ($|\text{Hyl}(w) \cap CK_k'| \geq \pi_{KE}$) then
        $CK' = CK' \cup w$;
    } Endif.
  } Delete $CK_k$.
  Return $CK'$
} End;

Figure 3.3 Pseudocode for Keyword Enrichment using Wikipedia knowledge

The effectiveness of $KE$ can be elaborated with the help of the fact that initially after using WordNet, the category list for *Space* did not have the names of any of the planets. It was realised that Wikipedia can be used as an effective learning tool. After $KE$ with classified documents and Wikipedia, the names of the planets were added to the category list for Space.

Our proposed knowledge based supervised learning did not use any prior labeled documents. Therefore wrongly classified documents can be used to enrich category keyword lists. However, note that only those tokens of the classified document are
being added to the current keyword list which carried out an acceptable intersection with current keywords. This double assurance strategy protects against enrichment by irrelevant tokens.

3.3) Experiments and Discussion

All steps of the proposed knowledge based DC algorithm were automated using a server-side scripting language PHP 5.3.4 on the Windows OS on an i7 quadcore processor 2.4GHz with Windows 7.

**Dataset:** We chose two well known datasets for our experiments. These datasets have been widely used in the area of DC.

(i) **Reuters 21578:** This dataset was collected from the Reuters Newswire in 1987 and was made available for research purpose mainly in the area of Information Retrieval (IR) in 1989 [49]. This dataset contains emails of personnel of Reuters Newswire. It has 90 specialized categories. The emails were manually classified by the personnel from Reuters Limited.

(ii) **20 News Group (20 NG):** It consists of 20 categories and approximately 20,000 documents [71]. Each category contains the documents for a different topic. These documents were collected from the 20 Usenet news group and evenly partitioned across all categories.

We proposed a mixed corpus with four categories namely, Fuel, Computer, Space and Sports. The documents of Fuel category have been taken from Reuters-21578. The datasets for other three categories were obtained from 20 Newsgroup. The DC system that we constructed by applying KE with **Belongingness** showed as classification accuracy of 85.75%. The semantic keyword enrichment technique with **Belongingness** is a new direction for document classification.

We experimented on two sizes of the same corpus. For Large_Dataset, we took 100 documents for each category. We selected 10 random documents for each category of the Large_Dataset to generate the Small_Dataset. Therefore, we had 400 documents in
Large_Dataset and 40 documents in Small_Dataset. **Figure 3.4** details the key approaches while experimentation and **Figure 3.5** shows the preset threshold values.

<table>
<thead>
<tr>
<th>Key approaches</th>
<th>Datasets</th>
</tr>
</thead>
<tbody>
<tr>
<td>WordNet prior keywords and Overlapping Semantics.</td>
<td>Small_Dataset</td>
</tr>
<tr>
<td>WordNet prior keywords, Overlapping Semantics and Keyword Enrichment.</td>
<td>Large_Dataset</td>
</tr>
</tbody>
</table>

**Figure 3.4** List of key approaches.

<table>
<thead>
<tr>
<th>Symbols</th>
<th>predetermined values</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>πₚₚ</td>
<td>3</td>
<td>To get the semantically related tokens from the document.</td>
</tr>
<tr>
<td>πₚₑ</td>
<td>20</td>
<td>To insert the token as keyword to respective category.</td>
</tr>
</tbody>
</table>

**Figure 3.5** Lists of thresholds and their values

**3.3.1) With WordNet keyword lists CK and OS**

We conducted two experiments using WordNet derived initial keyword lists of categories.

a) **On four files for each category:** We started our experiment on a set of four predefined categories namely: *Fuel, Computers, Sports* and *Space*. To illustrate the effect of each step, we took four files from the dataset, each belonging to one of the categories, namely, *fuel file, space file, comp file* and *sports file*. The
results so obtained are summarized in Table 3.1, which explicitly mentions the final \textit{Belongingness} values of each file to different categories.

<table>
<thead>
<tr>
<th>Category/Documents</th>
<th>Fuel</th>
<th>Computer</th>
<th>Sports</th>
<th>Space</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuelfile</td>
<td>0.280</td>
<td>0.056</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>Spacefile</td>
<td>0</td>
<td>0.055</td>
<td>0.058</td>
<td>0.1</td>
</tr>
<tr>
<td>Compfile</td>
<td>0</td>
<td>0.33</td>
<td>0.069</td>
<td>0.02</td>
</tr>
<tr>
<td>Sportsfile</td>
<td>0</td>
<td>0</td>
<td>0.244</td>
<td>0</td>
</tr>
</tbody>
</table>

\textbf{Table 3.1 Belongingness metric of four documents for each category}

\textit{Dimensionality reduction of Fuelfile document}: This document was pre-processed as per the proposed steps. After pre-processing, the size of the file reduced from 900 bytes to 291 bytes resulting in a compression of 67.67\%. The semantic sets of all tokens were retrieved from the WordNet source. The application of the concept of overlapping semantics on these semantic sets resulted in a final document of reduced dimensionality with only six tokens representing the meaning of the document greatly. The achieved compression was 81.44\% from 291 bytes to 54 bytes. This illustrates the significance of overlapped semantics in reducing the computational involved in DC.

\textit{Keyword-Strength Matrix}: The KC relationship matrix thus generated had rows corresponding to the six tokens extracted and the four categories represented by four columns. The presence or absence of each token in the given category was checked from the ontological keyword lists of the categories. \textbf{Table 3.2} shows the presence/absence matrix accordingly derived.
Next we extracted the Keyword Strengths of matched tokens for each category. The derived results are tabulated in Table 3.3.

Table 3.2 Presence of Tokens in different Category–Keyword list.

<table>
<thead>
<tr>
<th>Categories/Keywords</th>
<th>Fuel</th>
<th>Computers</th>
<th>Sports</th>
<th>Space</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Distillate</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2. Fuel</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3. Stocks</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4. Cargo</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5. Explosion</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6. Gasoline</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 3.3 Keyword Strength of matched tokens for each category
We already stored the Term Frequencies of these tokens while pre-processing. Finally, the Belongingness metrics showing the extent to which the Fuel file belongs to each category were calculated by the equation \(3.2\).

It was found that the given document had the maximum Belongingness metric for the category Fuel with 0.28 degree of Belongingness and was thus ascribed to this category. We can observe the result from the Table 3.1.

b) On Small_Dataset: Now, for the sake of observing the efficiency of the approach, we followed the same steps as described above and experimented on Small_Dataset. Here we utilized only initial WordNet based category keyword lists. The obtained results are summarized in Table 3.4

<table>
<thead>
<tr>
<th>Category</th>
<th>No of Documents</th>
<th>Correctly Classified</th>
<th>Incorrectly Classified</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer</td>
<td>10</td>
<td>06</td>
<td>04</td>
<td>60%</td>
</tr>
<tr>
<td>Fuel</td>
<td>10</td>
<td>10</td>
<td>00</td>
<td>100%</td>
</tr>
<tr>
<td>Space</td>
<td>10</td>
<td>09</td>
<td>01</td>
<td>90%</td>
</tr>
<tr>
<td>Sports</td>
<td>10</td>
<td>04</td>
<td>06</td>
<td>40%</td>
</tr>
<tr>
<td>Average</td>
<td>40</td>
<td>29</td>
<td>11</td>
<td>72.50%</td>
</tr>
</tbody>
</table>

Table 3.4 Classification Accuracy on Small_Dataset using WordNet and OS

Observation: As demonstrated in Table 3.4, the CK-OS driven algorithm achieved only 72.5% Classification Accuracy. when we analyzed the rationale behind this poor result, we found that documents had conceptually related terms which may helped in achieving good classification, but were missing from the present keyword lists. This observation led us to gradually enrich the category keyword lists by using each classified document as and when they classified.
3.3.2) With Enriched Keyword lists CK' and OS

The experiment was next performed on Large_Dataset by integrating KE to the proposed algorithm.

On Large_Dataset: The cumulative result thus obtained from the experiment is compiled in Table 3.5. The overall achieved average classification accuracy is 85.75%.

<table>
<thead>
<tr>
<th>Category</th>
<th>No. of Documents</th>
<th>Correctly Classified</th>
<th>Incorrectly Classified</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer</td>
<td>100</td>
<td>85</td>
<td>15</td>
<td>85%</td>
</tr>
<tr>
<td>Fuel</td>
<td>100</td>
<td>87</td>
<td>13</td>
<td>87%</td>
</tr>
<tr>
<td>Space</td>
<td>100</td>
<td>82</td>
<td>18</td>
<td>82%</td>
</tr>
<tr>
<td>Sports</td>
<td>100</td>
<td>89</td>
<td>11</td>
<td>89%</td>
</tr>
<tr>
<td>Average</td>
<td>400</td>
<td>343</td>
<td>57</td>
<td>85.75%</td>
</tr>
</tbody>
</table>

Table 3.5 Results on Large_Dataset using WordNet and KE

Observation: We can conclude here that after including Wikipedia, the category-keyword lists were enriched with good keywords. It improved the document classification accuracy from 72.50% to 85.75%. This is a 13.25% improvement which shows the good performance of KE. Figure 3.6 depicts category wise classification accuracy with both approaches viz with and without KE, graphically.
3.3.3) Analysis of Keyword Enrichment (KE)

Although the document was classified, but it was necessary to supervise the algorithm by knowledge based learning which can enhance the classification accuracy of the system. The classified document had tokens which did not originally save as keywords for the respective category but had a high potential to represent the category keywords. Hence, it became necessary to train the system using these classified documents. Now, Wikipedia was used to quantify the semantic relatedness of tokens to the pre-existing keywords. This way, keyword enrichment step was performed through the resourceful use of Wikipedia.

Figure 3.7 depicts the enhancement in degree of Belongingness of the document as the experiment was carried out after iteratively adding more keywords. KE gave improved results as the degree of Belongingness of Fuelfile increased from 0.280 to 0.644 as shown in Table 3.6.

<table>
<thead>
<tr>
<th>Category/Document</th>
<th>Fuel</th>
<th>Computer</th>
<th>Sports</th>
<th>Space</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuelfile</td>
<td>0.644</td>
<td>0.056</td>
<td>0.000</td>
<td>0.000</td>
</tr>
</tbody>
</table>

Table 3.6 Belongingness metric when Wikipedia was used
Improvement in Belongingness with successive KE: In the Figure 3.7 the X axis shows instances. Every instance indicates addition of few more important keywords to the category keyword list after keyword enrichment with classified documents. The Y axis shows Belongingness of a test document named as Fuel file in Table 3.1 to the category, which increased from an initial value of 0.28 to the final value 0.64.

![Figure 3.7 Degree of enhancement for the Fuelfile after incremented instances of KE](image)

New relevant keywords: The importance of KE can be further elaborated with another example. Initially after using WordNet, the category keyword list for ‘Space’ did not have the name of the planets. But as a resultant of KE, the names of the planets were added to the keyword list. Same as, for fuel category, the initial list did not contain the keywords such as ‘hydrocarbon’, ‘biofuel’ and ‘oxidation’, which were included to the list after KE.

As a performance measures, we used Micro average F1-measure, Precision, Recall and Classification Accuracy. Table 3.7 shows the values for different performance measures.
<table>
<thead>
<tr>
<th>Categories</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F1-measure (%)</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Space</td>
<td>88.17</td>
<td>82.00</td>
<td>83.91</td>
<td>82.00</td>
</tr>
<tr>
<td>Sports</td>
<td>90.81</td>
<td>89.00</td>
<td>89.89</td>
<td>89.00</td>
</tr>
<tr>
<td>Fuel</td>
<td>96.66</td>
<td>87.00</td>
<td>91.57</td>
<td>87.00</td>
</tr>
<tr>
<td>Computer</td>
<td>90.42</td>
<td>85.00</td>
<td>87.61</td>
<td>85.00</td>
</tr>
<tr>
<td>Average</td>
<td>91.51</td>
<td>85.75</td>
<td>88.24</td>
<td>85.75</td>
</tr>
</tbody>
</table>

Table 3.7 Performance measures for proposed scheme

In [89], by using document term features enrichment using Wikipedia concepts, the authors reported 73.64% average accuracy, whereas with keyword enrichment, we obtained an accuracy of 85.75%. Thus, we obtained 12.11% improved classification accuracy. Pu Wang et al reported 81.76% F1-measure [90]. They augmented the features of a training document using Wikipedia concepts which are absent in the document but present in any other training document of the same category. We reported comparatively 6.48% improved F1-measure by enriching the category keyword list using semantically related tokens of the documents.

3.4) Comparative Evaluation

Many past works relied on manually generated keywords for each category [68], [110]. This is a complex and time-consuming job. In our scheme, we build the representative keyword-list per category using WordNet as ontology. We have intentionally derived a broad spectrum of lexical relationships from the WordNet, including Synonyms, Hypernyms, Hyponyms as well as Meronyms and coordinate terms. This allows the extraction of a richer subset of keywords that bear a strong semantic relationship to the category name. This is further refined by Wikipedia enabled keyword enrichment by utilizing classified documents.
The idea behind using the Belongingness metric is that the degree of truth of a statement can range between 0 and 1 and is not constrained to the two truth values of classic propositional logic [111], [112]. The document is finally classified in that category for which the Belongingness metric has the maximum value. This, however, does not neglect the fact that the document does have a degree of Belongingness to other chosen categories as well. Our results demonstrate that this approach efficiently classifies the given document into its most relevant category.

Using Wikipedia to enrich the keyword list helps to increase classification accuracy as the category lists get enhanced with each successive document it classifies. In [89], [90] and [88], the authors extracted features to augment the term features of the documents using Wikipedia concepts and then employed a classification technique. In sharp contrast, we extracted semantically related tokens/features from the classified documents with help of Wikipedia hyperlinks and augmented the category keyword list of the respective category. This approach aids in identifying new documents’ categories with greater accuracy.

3.5) Summary

In this chapter, we implemented document categorization using important semantic features. The initial WordNet based Lexical Semantics keywords were augmented with Wiki-enabled semantically matched tokens from classified documents. Overlapped Semantics was used to reduce the dimensionality of documents. A new Belongingness metric has defined.

In the next chapter, in addition to the above features, we will generate noun and verb based Lexical Chains by using overlapping semantics. These chains will be used to contextually link together the tokens of a document and present more powerful semantic features which will help in enhancing classification accuracy. We will also introduce a new statistical metric Relative Category Frequency (RCF) which is an improvement over the classical Term Frequency-Inverse Document Frequency.