4.1 Overview

Language is primary means of communication. Language is used by humans to express views and ideas. The field of natural processing is focused on human computer interaction. It allows the computer to understand and analyse the information generated by various sources. It deals with either developing automated tools for language processing or formulating methods for better and efficient understanding of human language by the computer systems.

Natural language processing has an essential role in the field of Information retrieval. The storage and processing of information in response to user’s query requires an information retrieval system that processes human or natural language [41].

Natural language processing (NLP) is the automatic or semi-automatic processing of human language. NLP includes a large number of methods
for automated generation, manipulation and analysis of natural or human languages. NLP is a multidisciplinary field. It includes the fields of linguistic theory, cognitive science, psychology, philosophy, mathematics, formal language theory, compiler techniques, theorem proving, machine learning and human-computer interaction. Essential problems that make the processing of natural languages complex and different are:

(i) natural languages are implicitly ambiguous. The same word has multiple meanings in different context. Example book as a thing and book as a verb.

(ii) complexity of semantic information associated in sentences.\[41,42\]

The input to Natural language processing systems is strings of words and the output is structured representations defining the meaning of those strings. A natural language understanding system provides answers to users’ queries. It interfaces with a database and answers or output results of the input or query [11].

The major information available online is textual. The information retrieval systems are mostly keyword based. Only keyword based traditional methods of retrieval face challenges due to ambiguity in natural languages. The restrictions imposed due to polysemy and synonymy in natural languages such as English need a solution. The traditional methods ignore the semantics and contextual information in the retrieval process. This ignored and lost information extraction results require improvement in retrieval methods. One of the ways is to capture the semantics associated with textual data.
4.2 Essential terminologies in Natural Language Processing

- **Token**: Linguistic units such as words, punctuation, numbers or alphanumeric. For examples, each word is a token.

- **Sentence**: An ordered sequence of tokens.

- **Tokenization**: The method of splitting a sentence into its constituent tokens.

- **Corpus**: A body of text containing a large number of sentences. It is a collection of machine-readable texts. Example: A collection of medical journals.

- **Lexicon**: Words and their associated meanings. Example: English dictionary.

- **Part-of-speech (POS) Tag**: A POS tag is a symbol representing a lexical category such as Noun, Verb, Adjective, Article. The tag specifies syntactic role of a word. Parts-of-speech are called *lexical categories*. The set of tags used for a particular task is called as a *tagset*. A POS tagger assigns a part of speech tag to each word.

- **Morphology**: It defines the structure of words.

- **Syntax**: It defines the format or the method in which words are used to form phrases.

- **Semantics**: It defines the meaning based on syntax.

- **Pragmatics**: It is meaning in context.

- **Stemming and Lemmatization**: Both are pre-processing and similar operations of Natural language processing. The major difference between
is stemming may result in create non-existent words and lemmas are actual words of dictionary.

- **Stopwords**: words that have little or no semantic meaning associated. Example conjunctions.

- **WordNet**: It is semantic concept hierarchy (Miller, 1990). [41]

### 4.3 Information Retrieval and Natural Language Processing

Information retrieval (IR) is the practice of representing, storing, organizing, and allowing access to information repository. It finds and retrieves relevant text documents [42]. In IR systems, the information is unstructured. It is contained in free form in text such as web pages or other documents or in multimedia content. IR deals with searching for documents, for information within documents. It also includes searching for metadata about documents, searching structured storage repository, relational databases, and the World Wide Web. The input to an information retrieval system is user’s query. Queries are a representation of user’s need in form of a keyword or phrase. An information retrieval system returns a set of objects such as documents with a degree of relevancy and importance associated with it. The Information Retrieval (IR) domain can be viewed as an applied domain of NLP. Search engines are an application of Information retrieval that process natural language to answer user’s query. “IR deals with the representation, storage, organization of, and access to information items. These information items could be references to real documents, documents themselves, or even single paragraphs, as well as Web pages, spoken documents, images, pictures, music, video, etc.” Information Retrieval systems has to deal with vague and partial descriptions of both user needs and documents queried. Various application areas that require natural language
processing on information retrieval systems are machine aided translation, document clustering and classification, information extraction, question answering, natural language interfaces to databases. The challenges in retrieving relevant documents are due to ambiguity of natural languages. Polysemy and Synonymy of words are essential considerations for precise and efficient information retrieval[41,42].

Natural language processing functions at the following levels
• phonetic or phonological level :It is related with pronunciation.
• morphological level :It deals with the tokens i.e. parts of words that have meaning associated.
• lexical level :It deals with lexical meaning of words and parts of speech.
• syntactic level: It is concerned with grammar and structure of sentences.
• semantic level : It is associated with the meaning of words and sentences.
• discourse level: It is with the structure of different kinds of text using document structures.
• pragmatic level :It is concerned with the knowledge from exterior of document contents[41,42,43].

The various natural language processing levels and their corresponding information retrieval tasks are as follows[42,44,45]:

<table>
<thead>
<tr>
<th>Level of Analysis</th>
<th>IR tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phonological</td>
<td>No major contribution in textual information retrieval</td>
</tr>
<tr>
<td>Morphological</td>
<td>Stemming</td>
</tr>
<tr>
<td>Lexical</td>
<td>Stopword elimination, Part-of-speech tagging</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------------------------------------------</td>
</tr>
<tr>
<td>Syntactic</td>
<td>Identifying phrase units</td>
</tr>
<tr>
<td>Semantic</td>
<td>Word Sense Disambiguation</td>
</tr>
<tr>
<td>Discourse</td>
<td>Not much used in traditional IR</td>
</tr>
<tr>
<td>Pragmatic</td>
<td>No used in IR</td>
</tr>
</tbody>
</table>

**Table 4.1 Essential Tasks in Natural Language Processing**

### 4.4 Natural Language Processing in Python

Python is a simple and powerful programming language. It is an interpreted language. The language provides excellent functionality for processing linguistic data. The syntax and semantics of the language is transparent. It has good string-handling functionality. It is an object-oriented language and allows data and methods to be encapsulated and reused. The dynamic feature of Python permits variables to be typed dynamically and facilitates rapid development. Python is supported by an extensive standard library that includes components for graphical programming, numerical processing, and web connectivity. Python is widely used in scientific research, and education. The features of Python facilitate productivity, quality, and maintainability of software[46].

The interactive interpreter of Python permits the execution of Python programs. The Python interpreter is accessed via graphical interface called the Interactive Development Environment (IDLE). The >>>
symbol indicates that the Python interpreter is ready to take input from the user. For example a simple calculation at the prompt is:

```python
>>> 1+5*3
16.
```

16. The calculation results in output on the console.

Python package for natural language processing is NLTK (natural language processing toolkit). NLTK defines an infrastructure used for building natural language processing programs in Python. NLTK was originally created in 2001. It was created as part of a computational linguistics course in the Department of Computer and Information Science at the University of Pennsylvania. With time it has developed and expanded due to efforts and inputs of various contributors. NLTK was designed to fulfill the following objectives:

- **Simplicity**: The framework supports the essential and complex natural language processing tasks in an uncomplicated and simple way.

- **Consistency**: It provides a uniform framework and consistent interfaces for performing various complex tasks.

- **Extensibility**: New modules can be easily added. The existing framework is enhanced with alternative solutions to the same problem.

- **Modularity**: Components can be used without the knowledge of the complete toolkit.

Some essential NLTK modules are as follows:

- **Corpus**
- tokenize, stem
- Collocations
- Tag
- classify, cluster, tbl
- Chunk
- parse, ccg
- sem, inference
- metrics
- probability
- app, chat
- toolbox

Table 4.2 summarises the various natural language processing tasks supported by NLTK modules:

<table>
<thead>
<tr>
<th>NLTK module</th>
<th>NLP task</th>
</tr>
</thead>
<tbody>
<tr>
<td>Corpus</td>
<td>Accessing corpora by providing interfaces to standard corpus and lexicon.</td>
</tr>
<tr>
<td>tokenize, stem</td>
<td>String processing such as task of stemming</td>
</tr>
<tr>
<td>Collocations</td>
<td>Collocation discovery such functionality of t-test, chi-squared</td>
</tr>
<tr>
<td>Tag</td>
<td>Part-of-speech tagging</td>
</tr>
<tr>
<td>classify, cluster, tbl</td>
<td>Machine learning such as Naive</td>
</tr>
</tbody>
</table>
### Table 4.2 NLTK modules

<table>
<thead>
<tr>
<th>Chunk</th>
<th>Chunking</th>
</tr>
</thead>
<tbody>
<tr>
<td>parse, ccg</td>
<td>Parsing. The various functionality include feature-based, unification,</td>
</tr>
<tr>
<td></td>
<td>probabilistic, dependency</td>
</tr>
<tr>
<td>sem, inference</td>
<td>Semantic interpretation. The functionality include lambda calculus,</td>
</tr>
<tr>
<td></td>
<td>first order logic.</td>
</tr>
<tr>
<td>Metrics</td>
<td>Evaluation metrics such as precision and recall.</td>
</tr>
<tr>
<td>Probability</td>
<td>Probability and estimation</td>
</tr>
<tr>
<td>app, chat</td>
<td>Applications such as Wordnet browser, graphical concondancer</td>
</tr>
<tr>
<td>Toolbox</td>
<td>Linguistic fieldwork</td>
</tr>
</tbody>
</table>

#### 4.5 Implementation of essential NLP tasks in Python

Python and NLTK Module is mandatory for various natural language processing tasks. They are as follows:

I. NLTK module is included as follows
>>>import nltk
tltk.download()

II. To include all the contents of book module the command at the prompt is:

>>>from nltk.book import *

III. To search for text a concordance view displays the occurrence of a give word as follows:

>>>text1.concordance(“Shakespeare”)

IV. To find similar words

>>>text1.similar(“curious”)

V. To find the length of text i.e. number of words and punctuation marks from start till end

>>>len(text2)

VI. To display the dispersion plot i.e. positional information related to words in text

>>>text4.dispersion_plot([“citizens”, “democracy”, “duties”, “freedom”])

VII. Accessing document corpus and viewing the file identifiers
To access Gutenberg text corpora (corpus of about 25,000 free electronic books). The nltk package is included and the fields are displayed

```python
>>> import nltk
>>> nltk.corpus.gutenberg.fileids()
```

VIII. Counting the number of words in a particular file of the Gutenberg corpus:

```python
counte= nltk.corpus.gutenberg.words('shakespeare-hamlet.txt')
>>> len(counte)
```

In the above task one text file of the corpus is assigned name counte and the numbers of words are computed using words and len function.

Several access methods such as: words(), raw(), sents() are useful in finding various statistics related to text data.

Example Code snippet:

```python
>>> for fileid in gutenberg.fileids(): num_chars = len(gutenberg.raw(fileid))
num_words = len(gutenberg.words(fileid))
num_sents = len(gutenberg.sents(fileid))
```

IX. Including wordnet

```python
from nltk.corpus import wordnet as wnd
>>> wn.synsets('vehicle')
```

```python
[Synset('vehicle.n.01'), Synset('vehicle.n.02'), Synset('vehicle.n.03'),
 Synset('fomite.n.01')]
```

X. Accessing Stopwords
XI. Lemmitization

```python
>>> from nltk.stem import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()
print(lemmatizer.lemmatize("python"))
```

4.6 Example Python Program

A program that processes the file and prints all words ending with ing

```python
for line in open("myfile.txt"):
    for word in line.split():
        if word.endswith('ing'):
            print(word)
```