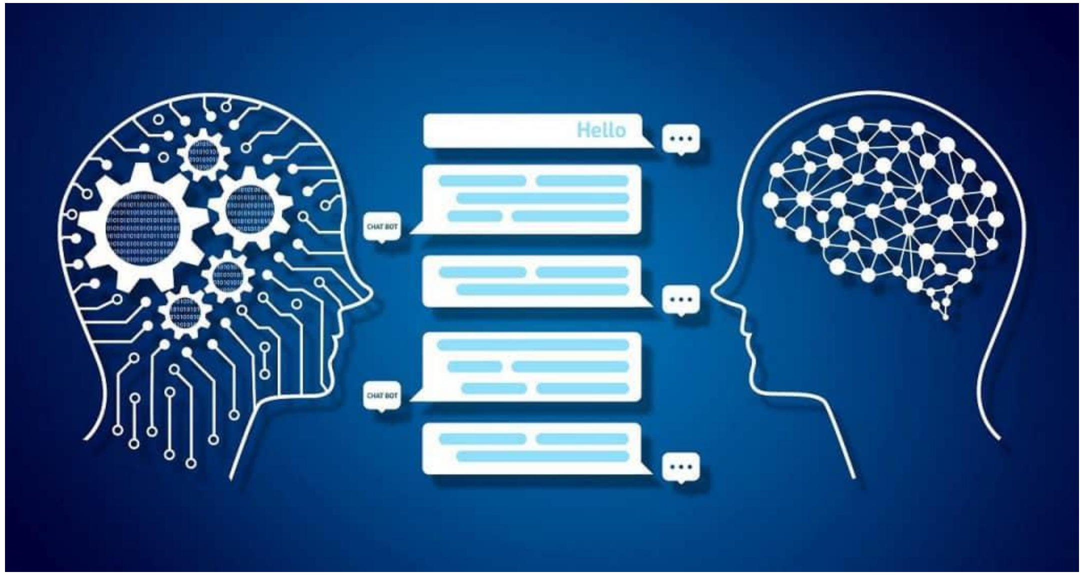
Natural Language Processing

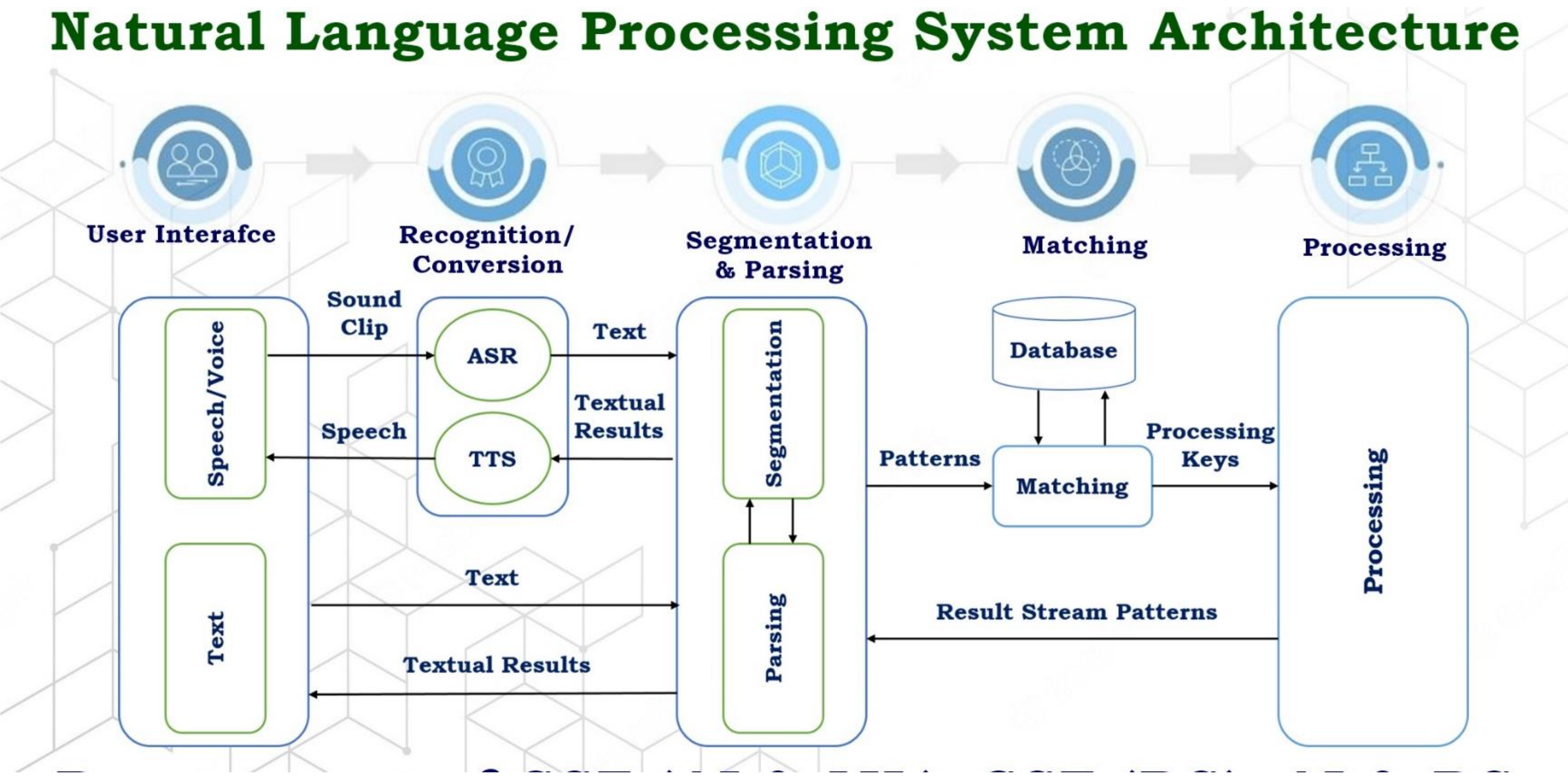
Module 1

Outline

- What is NLP?
- Components of NLP
    - Natural Language Understanding (NLU)
    - Natural Language Generation (NLG)
- Levels of NLP
    - Phonological Analysis
    - Morphological Analysis
    - Lexical Analysis
    - Syntactical Analysis
    - Semantic Analysis
    - Discourse Level
    - Pragmatic Level
- Introduction to Text Processing
    - Working with Text and PDF Files, HTML Files, XML Files, JSON Files, and Regular Expressions



Natural Language Processing Architecture



Components of NLP (Contd..)

NLP can be divided into two basic components.

- Natural Language Understanding
- Natural Language Generation

Natural Language Understanding (NLU) (Contd..)

- LU deals with understanding a given text and interpreting its meaning. It converts human language into a structured format that is usable by a computer.
- NLU is naturally harder than NLG tasks because of the following challenges faced by the machine while understanding, there is a lot of ambiguity while learning or trying to interpret a language.
    - **Lexical Ambiguity** can occur when a word carries different sense, i.e. having more than one meaning and the sentence in which it is contained can be interpreted differently depending on its correct sense. Lexical ambiguity can be resolved to some extent using parts-of-speech tagging techniques.
    - **Syntactical Ambiguity** means when we see more than one meaning in a sequence of words. It is also termed as grammatical ambiguity.
    - **Referential Ambiguity**: Very often a text mentions as entity (something/someone), and then refers to it again, possibly in a different sentence, using another word. Pronoun causing ambiguity when it is not clear which noun it is referring to

Natural Language Generation (NLG) (Contd..)

- It is the process of producing meaningful phrases and sentences in the form of natural language from some internal representation.

- It involves –

  - **Text planning** – It includes retrieving the relevant content from the knowledge base.

  - **Sentence planning** – It includes choosing required words, forming meaningful phrases, and setting the tone of the sentence.

  - **Text Realization** – It is mapping sentence plan into sentence structure.

Levels Of NLP (Contd..)

- The process of Natural Language Processing is divided into 5 major stages or phases, starting from basic word-level processing up to finding complex meanings of sentences.
- The given is a standard workflow, it may, however, differ drastically as we do real-life implementations basis on our problem statement or requirements.
- The source of Natural Language could be speech (sound) or text.
  - Phonological Analysis
  - Morphological Analysis/Lexical Analysis
  - Syntactical Analysis
  - Semantic Analysis
  - Discourse Level
  - Pragmatic Level

  - Often

Phonological Analysis (Contd..)

- This level is applied only if the text origin is a speech. It deals with the interpretation of speech sounds within and across words.
- Speech sounds might give a big hint about the meaning of a word or a sentence.
- Identifies and interprets the sounds that make up words when the machine has to understand the spoken language.

Morphological Analysis (Contd..)

- It is the process of determining the morphenes from which a given word is constructed. Morphenes are the smallest meaningful words that cannot be divided further. Morphenes can be stem or affix. The stem is the root word whereas the affix can be a prefix, suffix, or infix. For example-
- Unsuccessful -->

un	**success**	**ful**
prefix	stem	suffix

- Order of words also decides the morphological parser. To design a morphological parser we require three things- lexicon, morphotactic and orthographic rules.

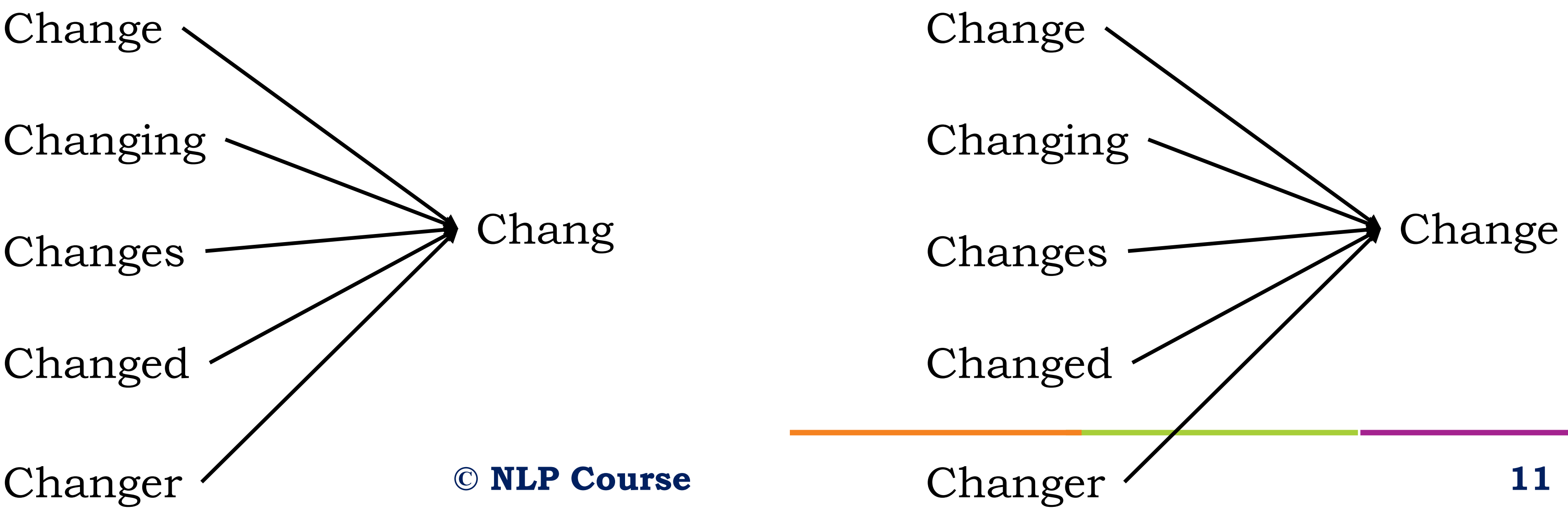
Lexical Analysis (Contd..)

- It involves identifying and analyzing the structure of words. The Lexicon of a language means the collection of words and phrases in a language. Lexical analysis is dividing the whole chunk of text into paragraphs, sentences, and words. I order to deal with lexical analysis, we often need to perform Lexicon Normalization.
- The most common lexicon normalization practices are Stemming:
  - **Stemming**: Stemming is a rudimentary rule-based process of stripping the suffixes (“ing”, “ly”, “es”, “s” etc.) from a word.
  - **Lemmatization**: Lemmatization, on the other hand, is an organized & step by step procedure of obtaining the root form of the word, it makes use of vocabulary (dictionary importance of words) and morphological analysis (word structure and grammar relations).

Lexical Analysis (Contd..)

- Lemmatization and Stemming, both are used to generate root form of derived (inflected) words. However, lemma is an actual language word, whereas stem may not be an actual word.
- For instance, stemming the word ‘Caring‘ would return ‘Car‘ and lemmatizing the word ‘Caring‘ would return ‘Care‘. Stemming is used in case of large dataset where performance is an issue.

Stemming vs Lemmatization



Syntactical Analysis (Contd..)

- Syntactic analysis is defined as analysis that tells us the logical meaning of certainly given sentences or parts of those sentences. We also need to consider rules of grammar in order to define the logical meaning as well as the correctness of the sentences.
- The syntactic analysis basically assigns a semantic structure to text. It is also known as syntax analysis or parsing. The word ‘parsing’ is originated from the Latin word ‘pars’ which means ‘part’. The syntactic analysis deals with the syntax of Natural Language. In syntactic analysis, grammar rules have been used.
- For example:

  Correct Syntax: *Sun rises in the east.*

  Incorrect Syntax: *Rise in sun the east.*

Semantic Analysis (Contd..)

- Semantic Analysis of Natural Language captures the meaning of the given text while taking into account context, logical structuring of sentences, and grammar roles.
- Semantic Analysis of Natural Language can be classified into two broad parts:
  - **Lexical Semantic Analysis**: Lexical Semantic Analysis involves understanding the meaning of each word of the text individually. It basically refers to fetching the dictionary meaning that a word in the text is deputed to carry.
  - **Compositional Semantics Analysis**: Although knowing the meaning of each word of the text is essential, it is not sufficient to completely understand the meaning of the text.
- Consider the sentence: “The apple ate a banana”. Although the sentence is syntactically correct, it doesn’t make sense because apples can’t eat. The semantic analysis looks for meaning in the given sentence. It also deals with combining words into phrases.

Discourse Level (Contd..)

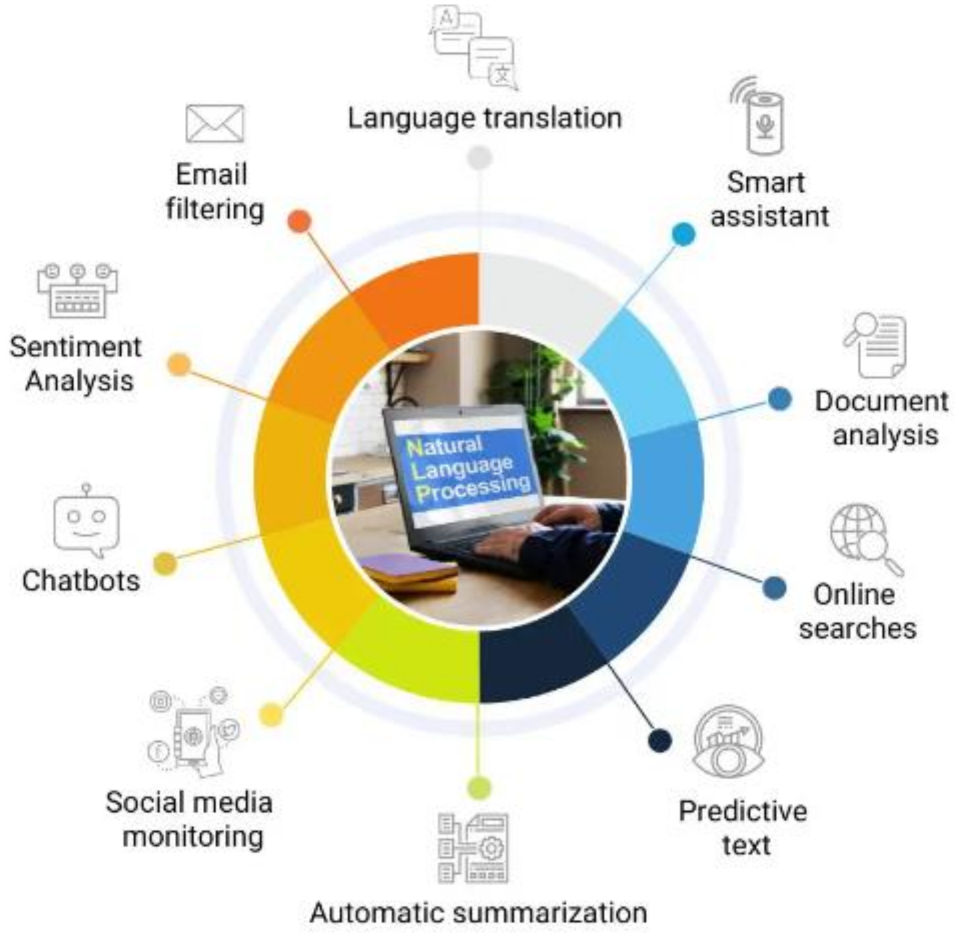
- The discourse level of linguistic processing deals with the analysis of the structure and meaning of text beyond a single sentence, making connections between words and sentences.
- In the text, “Jack is a bright student. He spends most of the time in the library.” Here, discourse assigns “he” to refer to “Jack”.

Pragmatic Level (Contd..)

- The pragmatic level of linguistic processing deals with the use of real-world knowledge and understanding of how this impacts the meaning of what is being communicated.
- By analyzing the contextual dimension of the documents and queries, a more detailed representation is derived.
- Given a sentence, “Turn off the lights” is an order or request to switch off the lights.

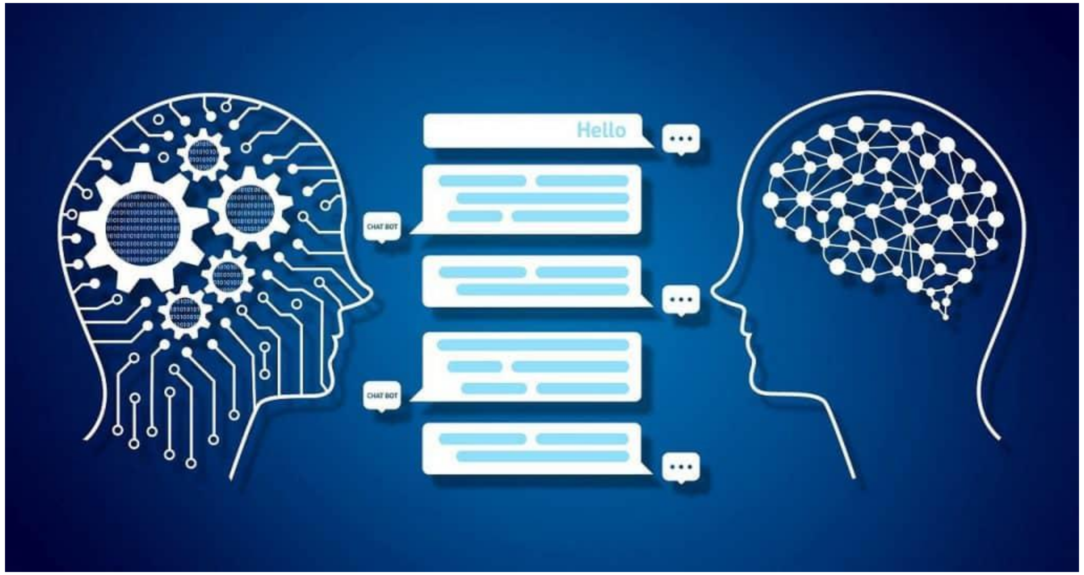
Applications of NLP (Contd..)

- Applications of NLP
  - E-Mail Filtering
  - Language Translation
  - Smart Assistants
  - Document Analysis
  - Online Searches
  - Predictive Text
  - Automatic Summarization
  - Sentiment Analysis
  - Chatbots
  - Social Media Monitoring



Outline – Session II

- Introduction to Text Processing: Working With
- Text Files
- HTML Files
- XML Files
- JSON Files
- PDF Files



Working with Text File (Contd..)

- Text files are probably the most basic types of files that you are going to encounter in your NLP endeavours.

- **Reading a Text File**
- Create a text file with the following text and save it in your local directory with a ".txt" extension.
- For example, I stored the file named "myfile.txt" in my root "D:" directory.

- **Reading All File Contents**
- The first step is to specify the path of the file, as shown below:
- myfile = open(“D:\myfile.txt”)

Working with Text File (Contd..)

- The output looks like this:
- <io.TextIOWrapper name='D:\\myfile.txt' mode='r' encoding='cp1252'>

- The output reads that myfile variable is a wrapper to the myfile.txt file and opens the file in read-only mode.

- Now, let's read the contents of the file. To do so, you need to call the read() function on the myfile variable, as shown below:
- print(myfile.read())

- In the output, you should see the text of the file.

Working with Text File (Contd..)

- A solution to this problem is that after calling the read() method, call the seek() method and pass 0 as the argument.
- This will move the cursor back to the start of the text file.

- myfile = open(“D:\myfile.txt”)
- print(myfile.read())
- myfile.seek(0)
- print(myfile.read())

- In the output, you will see the contents of the text file printed twice.

Working with PDF File (Contd..)

- By default, Python doesn't come with any built-in library that can be used to read or write PDF files. Rather, we can use the PyPDF2 library.

- Before we can use the PyPDF2 library, we need to install it. If you are using pip installer, you can use the following command to install PyPDF2 library:
- **$ pip install PyPDF2**

- Alternatively, if you are using Python from Anaconda environment, you can execute the following command at the conda command prompt:
- **$ conda install -c conda-forge pypdf2**

Working with PDF File (Contd..)

- **Reading a PDF Document**
- To read a PDF document, we first have to open it like any ordinary file. Look at the following script:
- **import PyPDF2**
- **mypdf = open('D:\Lorem-Ipsum.pdf', mode='rb')**

- The mode must be set to ‘rb’, which stands for "read binary" since most of the PDF files are in binary format.

- Once the file is opened, we will need to call the **PdfFileReader()** function of the PyPDF2 library, as shown below.
- **pdf_document = PyPDF2.PdfFileReader(mypdf)**

- Now using the **pdf_document** variable, we can perform a variety of read functions.

Assignment Question

- Take A Paragraph.
- Find out How Many Unigrams, Bigrams and Trigrams are Available in the Paragraph.
- Find out the Occurrence of the Word “The”. Assuming that the Word “The” is Used More than Once in The Paragraph.

THANK YOU