5. Artificial Intelligence



| **Module** | **Time** (**Mins**) |
| --- | --- |
| **Module 1: Introduction to Text Mining and NLP** |  |
| **Module 2: Extracting, Cleaning and Preprocessing Text** |  |
| **Module 3: Analysing Sentence Structure** |  |
| **Module 4: Text Classification-I** |  |
| **Module 5: Introduction to Deep Learning** |  |
| **Module 6: Getting Started with TensorFlow 2.0** |  |
| **Module 7: Convolution Neural Network** |  |
| **Module 8: ANN** |  |
| **Module 9: RNN, LSTM** |  |
| **Module 10: Transformers** |  |

# 

# Module 1: Introduction to Text Mining and NLP

# 

Topics:

## Overview of Text Mining

## Need of Text Mining

## Natural Language Processing (NLP) in Text Mining

## Applications of Text Mining

## OS Module

## Reading, Writing to text and word files

## Setting the NLTK Environment

## Accessing the NLTK Corpora

Hands-on:

* Install NLTK Packages using NLTK Downloader
* Accessing your operating system using the OS Module in Python
* How to read json format, understand key-value pairs, and for that matter, understand uses of pkl files

Skills You will Learn:

* Reading & Writing .txt Files from/to your Local
* Reading & Writing .docx Files from/to your Local
* Working with the NLTK Corpora

# Module 2: Text Analytics

****

**Topics:**

## Tokenization

## Frequency Distribution

## Different Types of Tokenizers

## Bigrams, Trigrams & Ngrams

## Stemming

## Lemmatization

## Stopwords

## POS Tagging

## Named Entity Recognition

**Hands-on:**

* Regex, Word, Blankline, Sentence Tokenizers
* Bigrams, Trigrams & Ngrams
* Stopword Removal
* UTF encoding, dealing with URLs, hashtags
* POS Tagging
* Named Entity Recognition (NER)

**Skills You will Learn:**

* Tokenization
* Stopword Removal
* UTF encoding
* POS Tagging
* Named Entity Recognition (NER)

# Module 3: Analysing Sentence Structure

# 

**Topics:**

## Syntax Trees

## Chunking

## Chinking

## Context Free Grammars (CFG)

## Automating Text Paraphrasing

**Hands-on:**

* Parsing Syntax Trees
* Chunking
* Chinking
* Automate Text Paraphrasing using CFG’s

**Skills You will Learn**

* Chunking
* Chinking
* Automate Text Paraphrasing

# Module 4: Text Classification-I

# 

**Topics:**

## Machine Learning: Brush Up

## Bag of Words

## Count Vectorizer

## Term Frequency (TF)

## Inverse Document Frequency (IDF)

**Hands-on:**

* Demonstrate Bag of Words Approach
* Working with CountVectorizer()
* Using TF & IDF

**Skills You will Learn**

* Bag of Words
* CountVectorizer()
* TF-IDF

# Module 5: Introduction to Deep Learning

# 

**Topics:**

## What is Deep Learning?

## Curse of Dimensionality

## Machine Learning vs. Deep Learning

## Use cases of Deep Learning

## Human Brain vs. Neural Network

## What is Perceptron?

## Learning Rate

## Epoch

## Batch Size

## Activation Function

## Single Layer Perceptron

**Hands-on:**

* Single Layer Perceptron

**Skills You will Learn**

* Curse of Dimensionality
* Single Layer Perceptron

# Module 6: Getting Started with TensorFlow 2.0

# 

**Topics:**

## Introduction to TensorFlow 2.x

## Installing TensorFlow 2.x

## Defining Sequence model layers

## Activation Function

## Layer Types

## Model Compilation

## Model Optimizer

## Model Loss Function

## Model Training

## Digit Classification using Simple Neural Network in TensorFlow 2.x

## Improving the model

## Adding Hidden Layer

## Adding Dropout

## Using Adam Optimizer

**Hands-on:**

* Classifying handwritten digits using TensorFlow 2.0

**Skills You will Learn**

* Installing and Working with TensorFlow 2.0

# Module 7: Convolution Neural Network (Optional)

# 

**Topics:**

## Image Classification Example

## What is Convolution

## Convolutional Layer Network

## Convolutional Layer

## Filtering

## ReLU Layer

## Pooling

## Data Flattening

## Fully Connected Layer

## Predicting a cat or a dog

## Saving and Loading a Model

## Face Detection using OpenCV

**Hands-on:**

* Saving and Loading a Model
* Face Detection using OpenCV

**Skills You will Learn**

* Image Classification using CNN
* Face Detection using OpenCV