Code	Title	Sem.	Credits	Lec	PS	Lab
	NLP Essentials	M1S2	5	14	20	16

Description

This course offers a foundational introduction to Natural Language Processing (NLP), focusing on the essential techniques and methodologies used to analyze, understand, and process human language data. Students will explore the key tasks of NLP, including text preprocessing, tokenization, part-of-speech tagging, named entity recognition, and syntactic parsing. The course also introduces core concepts in text representation (such as bag-of-words and TF-IDF), basic sentiment analysis, and language modeling using traditional approaches. Through a combination of theoretical instruction and practical exercises, students will develop hands-on skills in applying classical NLP methods to real-world text data. This course serves as a prerequisite for more advanced studies in NLP and machine learning-based language models.

Content

1. Text Preprocessing and Tokenization
    a. Text normalization: lowercasing, punctuation removal, stemming vs. lemmatization.
    b. Tokenization strategies: rule-based and statistical.
    c. Sentence segmentation and word boundary detection.
2. Part-of-Speech Tagging and Named Entity Recognition
    a. POS tagging: definition, applications, and common models (e.g., HMM, rule-based).
    b. Named Entity Recognition (NER): common entity types and use cases.
    c. Overview of standard taggers and NER tools (e.g., spaCy, NLTK).
3. Syntactic Parsing
    a. Constituency and dependency parsing: concepts and use cases.
    b. Introduction to parsing algorithms and tools (e.g., Stanford Parser).
4. Text Representation Techniques
    a. Bag-of-Words and TF-IDF: motivation, limitations, and implementations.
    b. Introduction to word embeddings (e.g., word2vec, GloVe - a basic overview).
5. Basic Sentiment Analysis
    a. Rule-based vs. supervised approaches to sentiment classification.
    b. Lexicons (e.g., VADER, TextBlob) and simple classifiers.
6. Language Modeling
    a. Concept of n-gram language models and their applications.
    b. Smoothing techniques (e.g., Laplace).
    c. Introduction to perplexity and evaluation metrics.

Software Tools & Environments:

- NLP Libraries: NLTK, spaCy, scikit-learn, TextBlob
- Programming Language: Python

- Development Environment: Jupyter Notebooks or similar IDEs
- Datasets: IMDB reviews, news articles, social media data.

Recommended References

- Speech and Language Processing by Daniel Jurafsky and James H. Martin
- Natural Language Processing with Python by Steven Bird, Ewan Klein, and Edward Loper
- Official documentation of spaCy, NLTK, and scikit-learn
- Online tutorials and NLP walkthroughs from open-source datasets and repositories.