Introduction to Natural Language Processing with Python

What is Natural Language Processing?

- Natural Language Processing (NLP) is a subfield of computer science and artificial intelligence (AI) that focuses on the interaction between computers and human language.
- NLP is concerned with developing algorithms and models that can analyze, understand, and generate natural language text or speech.
- Some common applications of NLP include machine translation, topic modeling, sentiment analysis, speech recognition, chatbots, and text summarization.
- NLP involves a wide range of techniques, such as statistical models, neural networks, rule-based systems, and deep learning.

Why NLP for Computational Social Sciences?

- NLP can be used to analyze social media data, such as tweets, posts, and comments, to understand public opinion, sentiment, and behavior.
- NLP can be used to extract and classify information from large text corpora, such as news articles or online forums, to identify trends, topics, and events.
- NLP can be used to detect and monitor hate speech, fake news, and propaganda in online media, to identify and mitigate harmful speech and behavior.

Why Python?

- Python is a versatile and easy-to-learn programming language that is widely used in the field of NLP.
- Python has a large and active community that has developed many libraries and frameworks for NLP, such as *NLTK*, *spaCy*, and *gensim*.
- Python is well-suited for data analysis and scientific computing, with many powerful tools and libraries for working with data, such as *pandas*, *NumPy*, and *matplotlib*.
- Python is free and open-source, meaning it can be used and modified by anyone without cost, and has a large and active community of developers contributing to its development and improvement.

Structure of the Workshop

- Short Introduction with slides

- Hands on with Python + Slides
  a. Extracting information from texts
  b. Text preprocessing
  c. Text representation
  d. Topic modeling

What you will learn in this session - Information Extraction

Dataset: Tweets

What is Information Extraction?

- Information extraction is a subfield of NLP that focuses on automatically extracting structured information from unstructured or semi-structured text data, such as emails, news articles, or social media posts.
- Information extraction involves identifying and extracting relevant entities, relationships, and attributes from text data, such as people, organizations, locations, dates, and events.
- Information extraction is important because it allows us to transform unstructured text data into structured data that can be easily analyzed and processed by computers.

In this Workshop:

- Extract text patterns using Regular Expressions (URLs, Hashtags, Mentions, Emojis)
- Named Entities Recognition (Person, Locations, Organizations)

What you will learn in this session - Text Preprocessing

Dataset: News articles

What is Text Preprocessing?

- Text preprocessing is a crucial step in NLP that involves cleaning and transforming raw text data into a format that can be easily analyzed by algorithms and models
- Text preprocessing is important because it can significantly impact the accuracy and effectiveness of NLP models and algorithms, and can help to improve the quality of text analysis

In this Workshop:

- Tokenization
- Lemmatization
- Stemming
- Part of speech tagging
- N-grams
- Stopwords

What you will learn in this session - Text Representation

Dataset: News articles

What is Text Representation?

- Text representation is the process of transforming text data into a numerical format that can be easily analyzed and processed by machine learning algorithms.
- Text representation is important in NLP because most machine learning algorithms require numerical data as input, and text data is inherently non-numerical.

In this Workshop:

- Bag of Words (BoW)
- Term Frequency Inverse Document Frequency (TF-IDF)

What you will learn in this session - Topic Modeling

Dataset: News articles

What is Topic Modeling?

- Topic modeling is a technique in NLP that aims to identify and extract the underlying topics or themes in a corpus of documents.
- Topic modeling is an unsupervised learning technique, meaning it does not require labeled data, and can be used to discover patterns and themes in large volumes of unstructured text data.
- Topic modeling involves analyzing the co-occurrence patterns of words in a corpus of documents, and identifying groups of words that tend to appear together frequently.

In this Workshop:

- Latent Dirichlet Allocation
- Evaluating the Topic Model

Text Preprocessing

Text Preprocessing

- Text preprocessing is a crucial step in NLP that involves cleaning and transforming raw text data into a format that can be easily analyzed by algorithms and models
- Text preprocessing is important because it can significantly impact the accuracy and effectiveness of NLP models and algorithms, and can help to improve the quality of text analysis
- Text preprocessing typically involves several steps, such as tokenization, stop word removal, stemming or lemmatization

Text Preprocessing - Tokenization

- Tokenization is the process of breaking down a text into smaller units called tokens.
- The most common tokenization method is to split the text on whitespace characters, such as spaces and tabs.
- A vocabulary is a set of unique tokens that appear in a corpus or collection of texts.

Example:

Today I am learning NLP with Python. Python is good programming language

Tokens:

[‘Today‘, ‘I‘, ‘am‘. ‘learning‘, ‘NLP‘, ‘with‘ , ‘Python‘, ‘.‘, ‘Python‘, ‘is‘, good‘, ‘programming‘, ‘language‘]

Vocabulary

[‘Today‘, ‘I‘, ‘am‘. ‘learning‘, ‘NLP‘, ‘with‘ , ‘Python‘, ‘.‘, ‘is‘, good‘, ‘programming‘, ‘language‘]

Text Preprocessing - Lemmatization

- Lemmatization is the process of reducing words to their base or root form, called a lemma.
- The purpose of lemmatization is to group together different inflected forms of a word so that they can be analyzed as a single item.
- Lemmatization can improve the accuracy of NLP models by reducing the number of unique words in the vocabulary and grouping together words with similar meanings.

Example:

Today I walked a lot, I like walking

Lemmatization:

- the tokens ‘walked’ and ‘walking’ shares the same lemma ‘walk’
- if we apply lemmatization to the tokens we get: *[‘Today’, ‘I’, ‘walk’, ‘a’, ‘lot’, ‘,’, ‘I’, ‘like’, ‘walk’]*
- and the vocabulary will simply be: *[‘Today’, ‘I’, ‘walk’, ‘a’, ‘lot’, ‘,’, ‘like]*

Text Preprocessing - Stemming

- Stemming is the process of reducing words to their base or root form, called a stem.
- Like lemmatization, the purpose of stemming is to group together different inflected forms of a word so that they can be analyzed as a single item.

So what is the difference between stemming and lemmatization?

- Lemmatization uses a dictionary to map each word to its lemma,
- Stemming simply removes the suffix from a word to produce a stem.
- Due to its simplicity, stemming is faster and less computationally expensive than lemmatization.
- However, lemmatization is more accurate than stemming

  Example:

  Word: "wolves"
  Stemming Result: "wolv"
  Lemmatization Result: "wolf"

Text Preprocessing - Stop Words

- Stop words are common words that are often removed from text data during preprocessing in NLP.
- Stop words are typically removed from text data because they do not carry much meaning or semantic value, and can therefore be safely ignored.
- Removing stop words can help reduce the size of the vocabulary and improve the efficiency of NLP tasks.
- Examples of stop words include "the", "a", "an", "and", "in", "of", "to", "is", "are", and many others.

Text Representation

Text Representation

- Text representation is the process of transforming text data into a numerical format that can be easily analyzed and processed by machine learning algorithms.
- Text representation is important in NLP because most machine learning algorithms require numerical data as input, and text data is inherently non-numerical.
- Text representation can be done using several techniques, such as bag-of-words and TF-IDF

Text Representation - Bag of Words

- The bag-of-words model is a common text representation technique in NLP that represents a document as a vector of word counts.
- The bag-of-words model assumes that the order of the words in the document does not matter, and only considers the frequency of each word in the document.
- The bag-of-words model:
    a. creates a vocabulary of all the unique words in the corpus
    b. each document is represented as a vector of word counts
        i. each element of the vector corresponds to a word in the vocabulary
        ii. its value represents the frequency of that word in the document

Text Representation - TF-IDF

- TF-IDF stands for Term Frequency-Inverse Document Frequency.
- TF-IDF is a text representation technique in NLP that assigns weights to words in a document based on their frequency and importance in the corpus.
- The TF-IDF model
    a. represents a document as a vector of weighted word counts
    b. the weight of each word is proportional to its frequency in the document and inversely proportional to its frequency in the corpus.

Topic Modeling

Topic Modeling

- Topic modeling is a technique in NLP that aims to identify and extract the underlying topics or themes in a corpus of documents.
- Topic modeling is an unsupervised learning technique, meaning it does not require labeled data, and can be used to discover patterns and themes in large volumes of unstructured text data.
- Topic modeling involves analyzing the co-occurrence patterns of words in a corpus of documents, and identifying groups of words that tend to appear together frequently.

Topic Modeling - Latent Dirichlet Allocation

- Latent Dirichlet Allocation (LDA) is a probabilistic topic modeling technique in NLP that discovers latent topics in a corpus of text data.
- LDA represents documents as mixes of topics, where each topic is a probability distribution over words in the vocabulary.
- LDA assumes that
    a. each document in the corpus is generated by a mixture of topics
    b. each word in the document is generated by one of the topics with a certain probability.
- LDA learns the topic distributions and word distributions from the corpus using Bayesian inference and optimization algorithms.