Part-of-speech problem:

Introduction:

Part-of-speech (POS) tagging is a process of marking each word in a text with its corresponding part of speech, such as noun, verb, adjective, etc. POS tagging is a critical step in natural language processing (NLP). One of the main challenges in POS tagging is that many words in a language can have multiple possible POS tags depending on the context in which they are used. In addition, POS tagging can be affected by language-specific issues, such as the presence of inflectional morphology, which can make it difficult to determine the correct POS tag for a word. For example, in Arabic, the same root word can have multiple inflections depending on the context, which can lead to ambiguity in POS tagging.

NetworkX provides various tools for creating and analyzing graphs, as well as functions for visualizing graphs with node and edge attributes. Networkx can be used in NLP tasks to create graphs that represent the syntactic structure of a text. However, when combining POS tagging with networkx, there can be a problem with how the POS tags are displayed in the graph. The POS tags are typically used as edge labels in the graph to indicate the syntactic relationship between adjacent words in a sentence. However, if the font size of the edge labels is too small or if there are too many edges in the graph, the POS tags may become difficult to read or may not be visible at all.

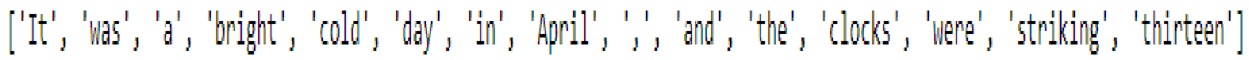
Data description:

There is no dataset used for this task.

Baseline experiments:

The baseline experiments can be broken down into 3 stages:

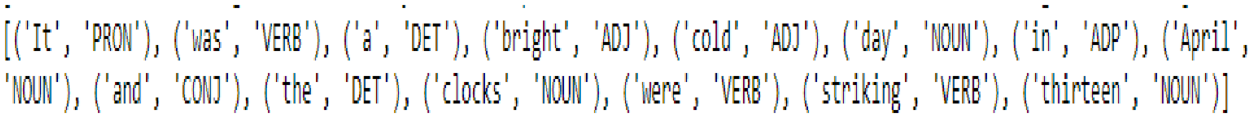
- Tokenization:
  Convert input text into tokens using the word_tokenize function from the nltk to make it easy to extract each word and it's pos tagger.



```
['It', 'was', 'a', 'bright', 'cold', 'day', 'in', 'April', ',', 'and', 'the', 'clocks', 'were', 'striking', 'thirteen']
```

Fig.1. The result from the word_tokenize function from the nltk

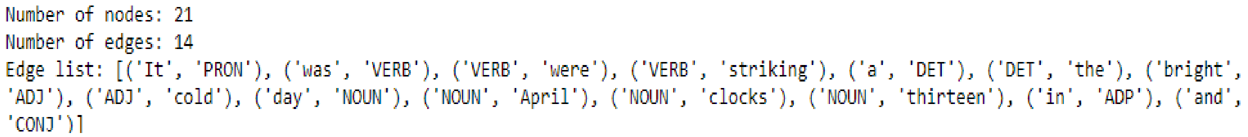
- Part-of-speech (POS) tagging:
  Get the tagging for each token using nltk and use universal tag to make pos tagging readable to anyone.



```
[('It', 'PRON'), ('was', 'VERB'), ('a', 'DET'), ('bright', 'ADJ'), ('cold', 'ADJ'), ('day', 'NOUN'), ('in', 'ADP'), ('April',
'NOUN'), ('and', 'CONJ'), ('the', 'DET'), ('clocks', 'NOUN'), ('were', 'VERB'), ('striking', 'VERB'), ('thirteen', 'NOUN')]
```

Fig .2. The pos tagging for each token using nltk

- Visualization:
  - Graph creation , a graph is created by using the NetworkX library. The nodes of the graph correspond to the POS tags, and the edges correspond to the co-occurrence of tags in the text. The Two nodes are connected by an edge if they are adjacent or separated by one or two words in the text and get the number of edges and nodes.



```
Number of nodes: 21
Number of edges: 14
Edge list: [('It', 'PRON'), ('was', 'VERB'), ('VERB', 'were'), ('VERB', 'striking'), ('a', 'DET'), ('DET', 'the'), ('bright',
'ADJ'), ('ADJ', 'cold'), ('day', 'NOUN'), ('NOUN', 'April'), ('NOUN', 'clocks'), ('NOUN', 'thirteen'), ('in', 'ADP'), ('and',
'CONJ')]
```

Fig.3. Discriptian of the POS graph.

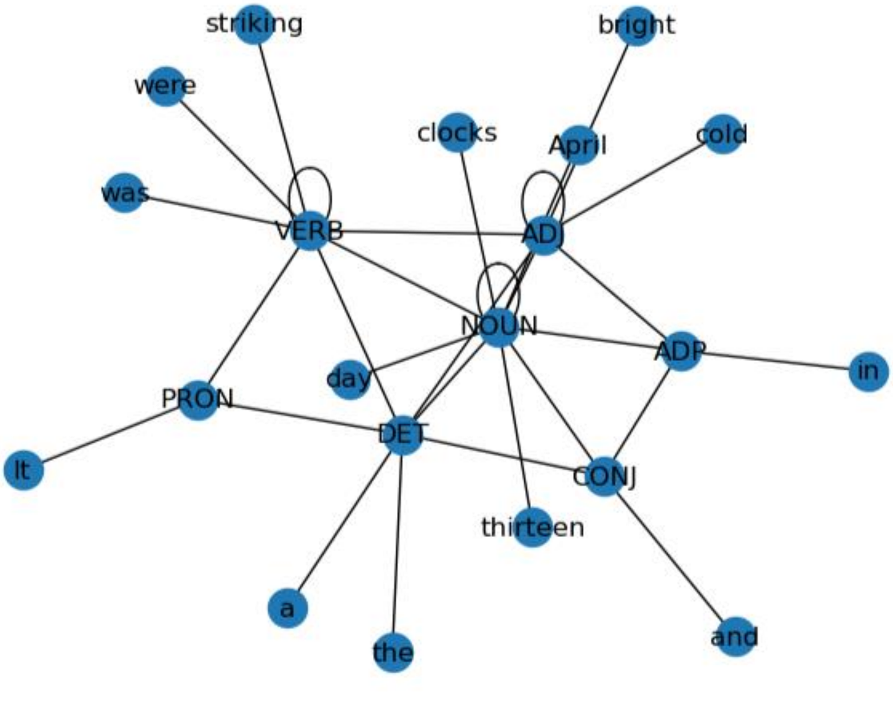


Fig.4. The POS tags graph

- Graph for the POS tags and syntactic dependencies using spacy render.

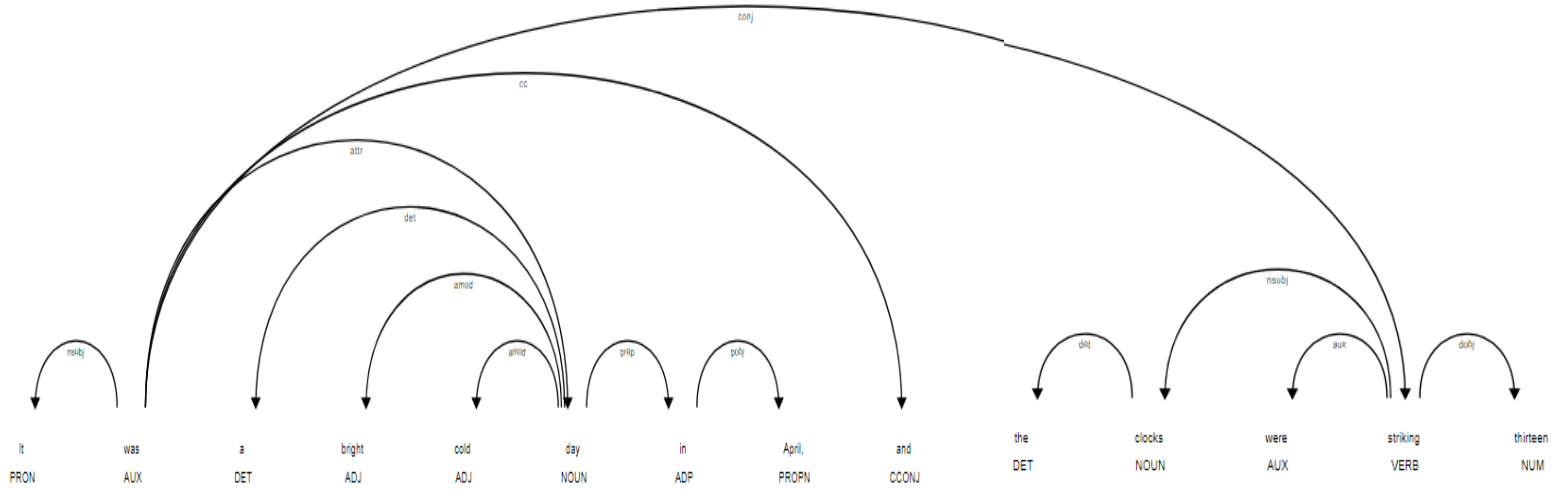
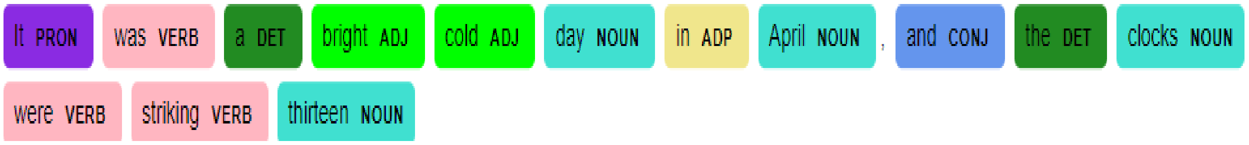


Fig.5. The POS tags and syntactic dependencies using spacy.

- Visualize POS tags with NLTK and SpaCy



It PRON was VERB a DET bright ADJ cold ADJ day NOUN in ADP April NOUN , and CONJ the DET clocks NOUN were VERB striking VERB thirteen NOUN

Fig.6. ThePOS tags with NLTK and SpaCy.

The questions Answer:

1) **What was the biggest challenge you faced when carrying out this project?**
   The biggest problem was to select the best tokenizer and pos_taggging because there are many but not perform well with different sentences.

2) **What do you think you have learned from the project?**

   The project demonstrates how to use Python libraries to perform text analysis and create visualizations of the results. It can help in tasks such as text classification, sentiment analysis, and topic modeling. And how to use SpaCy with NLTK to visualize data and the depending on POS how to get syntactic dependencies

- **Conclusion:**

  Part-of-speech (POS) is one of the most important techniques used to solve most problems in NLP. Part-of-speech (POS) analysis on a text is performed by using Python libraries such as NLTK and NetworkX. The code tokenizes the text, performs POS tagging, and creates a graph that visualizes the co-occurrence of POS tags in the text. The graph can be useful in identifying patterns and relationships between words and parts of speech in a text. The code also adds additional edges to the graph based on the distance between nodes in the text. Overall, this project showcases the power and flexibility of Python for natural language processing tasks, and how it can be used to gain insights into the structure of texts.

- **Resources:**
  - https://towardsdatascience.com/visualizing-part-of-speech-tags-with-nltk-and-spacy-42056fcd777e