Homework 1

Overview:

This assignment will provide hands-on practice with text processing techniques in Python, including tokenization, lemmatization, and stemming. You will also gain experience loading, analyzing, and scraping textual data from different sources.

Setup (10 points)

1. Install a Python programming environment (e.g. PyCharm, Jupyter Notebook)

2. Install these Python libraries: nltk, spaCy, BeautifulSoup

3. Create a new Python file to complete this assignment in

Data Loading & Basic Analysis (10 points)

4. Load the spam.csv dataset (source: KAGGLE)

5. Print basic statistics on the data:

- Total number of SMS messages
- Number of spam/ham messages
- Average number of words per message
- 5 most frequent words
- Number of words that only appear once

Text Processing (40 points)

6. Tokenize the SMS text using both nltk and spaCy. Analyze the time complexity of the tokenization algorithm

7. Lemmatize the SMS text using nltk and spaCy. Analyze the time complexity of the lemmatization algorithm

8. Stem the SMS text using nltk and spaCy. Analyze the time complexity of the stemming algorithm.

9. For each technique, write 2-3 sentences comparing the nltk and spaCy implementation. Consider things like output format, processing speed, language support etc.

10. Print updated statistics on word count and frequent words after applying each technique.

Web Scraping (20 points)

11. Use BeautifulSoup to scrape text data from a public page on one of your social media profiles.

12. Perform tokenization, lemmatization, and stemming on the scraped text.

13. Print word statistics on the scraped data before and after text processing.

WhatsApp Analysis (20 points)

14. Import a .txt file of at least 50 WhatsApp messages in Hebrew.

15. Tokenize, lemmatize, and stem the WhatsApp data.

16. Print comparisons of word statistics before and after processing.

Submission

- Submit your completed .py or .ipynb file to the course portal

- Include written responses in code comments or markdown cells.