**Week 3 – Lab Assignment**

**NOTES:**

1. **Before attempting this lab assignment, make sure you have completed the exercise in week 3 - lab exercise**
2. **Upon completing this exercise, you’ll need to upload your python (jupyter .ipynb file) to canvas before the end of this week.**

**Assignment details:**

* **Rules:**
  + The code must be **fully commented**.
  + **Print all outputs** so results are clearly visible.
  + **Use appropriate libraries** to complete the tasks. E.g. SpaCy.
  + **Do not use NaN values**—filter or replace missing data where necessary.
  + Your final script should run **without errors** when executed.
* **Datasets:** Use the [Amazon Fine Food Reviews](https://www.kaggle.com/datasets/snap/amazon-fine-food-reviews) dataset from Kaggle:
  + Reviews.csv

**📖 Lab Exercise Book Outline**

**Objective:**

The goal of this assignment is to apply advanced techniques for representing text as numbers and visualising the results. You will process and analyse the *Amazon Fine Food Reviews* dataset using Bag of Words (BoW), TF-IDF, and Word Embeddings (Word2Vec or GloVe), culminating in visualisations to extract insights from the reviews.

**Dataset:**

Use the **Amazon Fine Food Reviews** dataset, which contains over 500,000 reviews. The dataset is available on Kaggle. Focus on the Text and Score columns. [Amazon Fine Food Reviews](https://www.kaggle.com/datasets/snap/amazon-fine-food-reviews) (You may need to truncate the data if your machine is unable to process the entire dataset, e.g. top 10,000 reviews …etc)

**Tasks:**

1. **Text Preprocessing**
   * Clean and prepare the text by applying:
     + Tokenisation
     + Lowercasing
     + Stopword removal
     + Lemmatization
   * Use Python libraries such as spaCy or NLTK for these steps.
2. **Text Representation**
   * Represent the text using:
     + **BoW:** Create a sparse matrix and visualise word frequency distribution.
     + **TF-IDF:** Generate vectors, highlighting the most important words in each document.
     + **Word Embeddings:** Use either Word2Vec or GloVe to capture semantic relationships.
   * Ensure each method outputs numerical representations for further analysis.
3. **Visualization 1: Comparison of Text Representations**

* Compare and contrast visual representations of the text using different methods:
  + **Word clouds for BoW and TF-IDF:**
    - Generate separate word clouds using the Bag of Words and TF-IDF techniques.
    - The word clouds should represent the frequency or importance of words, with larger words indicating higher frequency or importance.
    - Use the WordCloud library to create these visuals.
  + **Heatmaps or bar charts showing the top 20 most frequent or important words:**
    - Extract the 20 most frequent words from the BoW model and the 20 most important words from the TF-IDF model.
    - Create bar charts to display the frequency or importance scores of these words.
    - Alternatively, use heatmaps to show the intensity of word occurrences across different reviews.
  + Use libraries such as Matplotlib and Seaborn to generate these visualizations. Ensure that all visualizations are clear, labelled, and easy to interpret.

1. **Visualization 2: Relationship Between Review Scores and Word Usage**

* Investigate how word usage changes with review scores. Focus on two categories:
  + **Positive Reviews:** Scores of 4 and 5.
  + **Negative Reviews:** Scores of 1 and 2.
* Identify the words that appear frequently in both categories, as well as those that are unique to each category.
* Use word embeddings to visualize clusters of words with similar meanings in each category. For example:
  + Generate word embeddings using Word2Vec or GloVe.
  + Apply dimensionality reduction techniques such as PCA or t-SNE to reduce the embeddings to 2D space.
  + Plot these word clusters using scatter plots, clearly differentiating between positive and negative words.
  + Highlight words that are common to both categories using a distinct colour or marker.
* The goal is to visually demonstrate the similarities and differences in language used in positive and negative reviews.

1. **Visualization 3: Sentiment Analysis and Temporal Trends**

* Perform sentiment analysis on the reviews using either TF-IDF or Word Embeddings.
* Assign sentiment scores to each review based on the words used. For example:
  + Use pre-trained sentiment lexicons or machine learning models to classify each review as positive, neutral, or negative.
* **Visualize sentiment trends over time:**
  + If the dataset includes review dates, group reviews by month or year and calculate the average sentiment score for each time period.
  + Plot these scores using line charts to show how sentiment has changed over time.
* **Visualize sentiment distribution:**
  + Use heatmaps or scatter plots to show the distribution of positive, neutral, and negative reviews within the embedding space.
  + Ensure that visualisations are clear, with appropriate labels and legends.

**✅ Submission Guidelines**

Ensure your Jupyter Notebook **includes all required charts and explanations**.

* Save your notebook as **Week3\_lab\_assignment\_YourID.ipynb**.
* Upload your **.ipynb file** to Canvas before the deadline.