Assignment 1

In-Depth Word Vectors Analysis (Total: 50 Points)

Objectives

This assignment focuses on a comprehensive understanding of word vector technologies, specifically Word2Vec and GloVe. You will explore their applications, visualize the results, and analyze the semantic and syntactic relationships they capture.

1. Building and Analyzing Word Vectors with Word2Vec (20 Points)

- **Task:** Create word vectors using the Word2Vec model on a selected corpus.
- **Visualization:** Utilize PCA or t-SNE for visualizing these vectors in 2D.
- **Analysis:** Discuss the word relationships and clusters formed.

2. GloVe Vectors Advanced Analysis (20 Points)

- **Task:** Analyze word relationships using pre-trained GloVe vectors.
- **Activity:** Conduct an analogy task (e.g., *king - man + woman = ?*) with examples.
- **Explanation:** Explain the results of the analogy tasks.

3. Semantic and Syntactic Word Relationships (10 Points)

- **Comparison:** Evaluate Word2Vec and GloVe for capturing semantic and syntactic relationships.
- **Illustration:** Use specific word pairs or groups to demonstrate differences in representation by each model.

Resources for Word2Vec and GloVe:

- **Word2Vec:** Pre-trained Word2Vec embeddings can be found on repositories such as HuggingFace or directly using Gensim.
- **GloVe:** Pre-trained GloVe embeddings are available at the GloVe website or through HuggingFace.

Submission Guidelines

- Submit all code as a Google Colab notebook, ensuring the link is shared in the assignment description with **Anyone with the link** access. Additionally, upload the notebook file and include detailed explanations and visualizations, either within the notebook as markdown cells or in a separate PDF report. Avoid sharing .py or ZIP files for this assignment.
- Clearly label each part and question in your submissions.
- **Deadline:** February 14th, 2025

Rubric and Expectations

- **Code Quality and Functionality (40%):** Code should be well-organized, commented, and functioning as intended. The use of Python and relevant libraries (e.g., Gensim for Word2Vec) should demonstrate a good grasp of the tools.
- **Analysis and Interpretation (30%):** Written explanations should be insightful, demonstrating a deep understanding of the word vector models. Analysis of visualizations, analogy tasks, and model comparisons should be thorough and reflective.
- **Visualization (20%):** Visualizations should be clear, accurately labeled, and effectively convey the relationships or patterns discovered in the data.
- **Adherence to Guidelines (10%):** Submissions should follow the provided guidelines, including format, labeling, and adherence to the deadline.