Word2Vec Exercise

Goal: Training and utilizing word embedding models

Write a Python program (and some comments) in the form of ipynb, following the details below.

- Due: 2024-04-08 (Mon) 23:59
- Submit on eTL

[1] Dataset

- Texts from Gutenberg project (several novels merged)

[2] To do list

1. Load the dataset file and pre-process it
   - Proper tokenization
   - Removal of useless or unimportant characters/words/phrases (optional)
   - You can use external libraries. (e.g., nltk, SpaCy)
2. Train a word2vec model based on the preprocessed corpus.
   - Use **gensim** library
   - Use any of the algorithm: skip-gram or cbow
   - **Explain the algorithm you selected briefly and provide a rationale for your choice.**
3. With your trained word2vec model (of step 2), print the following things:
   - The embedding for the word “rain”
   - Top 5 most similar words of the terms “Emma” (or “emma” if needed) and “Hamlet” (or “hamlet” if needed), and compare the results: **explain how the most similar words are different.**
4. Download and load a pre-trained word embedding model from gensim library.
   - Use the given methods of gensim.
   - Choose ‘**glove-wiki-gigaword-100**’: a 100-dimensional word embedding model (GloVe) trained based on Wikipedia and news articles.
   - Print the top 5 similar words of the terms “emma” and “hamlet”. Then, compare these results to those obtained in step 3. **Provide an analysis highlighting any similarities or differences, and explain the reasons behind any observed disparities.**
5. Utilizing the model from step 4, calculate the cosine similarities between two words from each of the following pairs.
   - “apple” and “boots”
   - “apple” and “steve”
   - “apple” and “plum”

- Then, compare these results. **Provide an analysis highlighting any similarities or differences, and explain the reasons behind any observed disparities.**