CS310 Natural Language Processing - Assignment 2: Word2vec Implementation
Total points: 50

Task: Train a word2vec model using the skip-gram architecture and negative sampling.

- The corpus data being trained on is the full text of 《论语》.
- Use the code from **Lab 4** to help you.

Submit:

- A modified notebook file `A2_w2v.ipynb`
- A text file of word embeddings (.txt format)
- Any other Python files your notebook depends on.
- Write up the results for requirement 3 and 5 in a Word/PDF document.

Requirements:

1) (10 points) Implement the data loading and processing pipeline. You should re-use and augment the code for `generate_data` and `batchify` functions.

2) (15 points) Implement the `SkipGram` class. The key is to implement the computation for `loss` in `forward` function. Make sure the inputs to this function are tensors in correct dimensions.

3) (10 points) Implement the `train` function that runs correctly.
   a) Print the loss every # intervals (determine the # by your own observation).
      Include a screenshot of loss change in your write-up.
   b) **Briefly describe** how you determine the training epochs needed by observing when the loss stops decreasing significantly.

4) (10 points) Run training with different hyper-parameters.
   a) Train with `emb_size` = 50, 100 respectively
   b) Train with negative sample number `k` = 2, 5 respectively
   c) Train with `window_size` = 1, 3 respectively
      Therefore, there are in total $2 \times 2 \times 2 = 8$ experiment groups, resulting in 8 embedding files. Record the computation time.

5) (5 points) Plot and compare the embeddings.
   a) Use Truncated SVD to reduce the dimension of embeddings from the target words provided (`['学', '习', '曰', '子', '人', '仁']`).
      Plot all 8 embedding results and include the plots in your report.
      You may use other words instead, if they you find interesting patterns.
   b) Pick one embedding plot, and compare it from the plot from Lab 4 (LSA). Observe if there are differences and **briefly describe**.
   c) **Save** one version of word embeddings to .txt format