CS 559A: Machine Learning

Homework Assignment 5 (Optional)

Due Date: Saturday 6:30 PM, May 7, 2022 **Total: 100 points**

Problem 1. Neural Networks (50 points)

Develop a Neural Network (NN) model to predict class labels for the Iris data set. Split the data into training, validation and testing set with the ratio of 0.8, 0.1 and 0.1. Train the model on the training set, select the best model based on the validation set, and test your model on the testing set. Report your training, validation and testing accuracy. You can use packages such as Tensorflow or Pytorch.

Link to the data: https://archive.ics.uci.edu/ml/datasets/Iris

Problem 2. Word2vec (50 points)

In this assignment, you will use word2vec embeddings to find the nearest words for a given word. The pretrained word embeddings file (vectors.txt) is already given, so you do not need to run word2vec model. The file contains word2vec embeddings for 400K words, and the dimension of each vector is 50. Each line contains the word and its corresponding vector. The first word in each line is the word, followed by 50 numbers, where each number is a dimension of the vector.

1) Semantics: Use the pre-trained embeddings file to compute the 20 most similar words using cosine similarity for the following words, and show your work.
   a. life
   b. market
   c. Stanford

2) Visualization
   a. Create a t-sne visualization, displaying all the words in the file.
   b. Use t-sne visualization to display the nearest 20 words for a given word. Create a separate visualization of all the 3 words given in question 1), where each visualization displays the nearest 20 words for a word.

Note: t-SNE is a tool to visualize high-dimensional data. You can find more details with the link provided here and use the implementation in sklearn: https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html. To visualize the data, you can use matplotlib library: https://matplotlib.org/.

Please submit the plots along with the code to generate the visualization.

More information about solving this problem:

1. How to load the data: please refer to the following links. Basically, you can add '400000 50' to the first line and used the function “KeyedVectors.load_word2vec_format” to load the data.

   https://github.com/3Top/word2vec-api/issues/6;
   https://radimrehurek.com/gensim/models/keyedvectors.html

2. If your computer can not handle 400k word data, please try to run it on a subset but keep the size of the subset as large as you can.
3. If tsne is too slow, you can run it on google colab, or if you use NVIDIA graphic card you could search cudatsne which will run the model on GPU.