

Assignment 4: Word Embedding Network

Description

In this assignment you will practice how to create a Word Embedding Network in Tensorflow 2.0. First, you will finish some functions to parse the data, build the corpus and construct the skip pair. Then, you will construct a word embedding network by follow the specific requirements and architectures. Finally, you will train the network and visualize the result.

Instruction

In this assignment, you need to fill the block of code in the python notebook file. The descriptions of all the functions you need to implement are as follow:

- **build_dictionary (10 points):** Extract the word from the input. Build a non-duplicate word dictionary.
- **one_hot_encoding (10 points):** Every word is represented as a vector containing 1 at its position in the vocabulary.
- **build_word_index_mapping (10 points):** Given a word, the function should return the index of this word in dictionary. Given an index, the function should retrieve the word.
- **build_skip_pair (10 points):** Build the word pair with given window size.
- **MyEmbeddingModel (20 points):** Define all the layers you will use in the embedding network. Define the network layer connectivity.
- **Optimizer (10 points):** Implement five different optimizers (SGD, RMSprop, Adagrad, Adadelata, Adam). Use the Adam optimizer for training.
- **build_embedding_dict (10 points):** Iterate the corpus_dict and generate the embedding for each word. Use the trained model to generate the word embedding with given one-hot embedding word. Store the word and embedding in a dictionary. The key should be the word. The value should be the embedding vector.
- **euclidean_dist_np (10 points):** Calculate the Euclidean distance between two input vectors.
- **find_closest (10 points):** Calculate the Euclidean distance between the given word and all the words in embedding dictionary. Sort the dictionary by value in ascending order. Return the first three closet words.

Note:

- In each file there are comments that walk you through the implementation, and also, there is explanation in each block of code that you have to fill in.
- Points for each block of code is also in the comment.
- Comment your codes.
- Please make sure each block of your code is runnable in Google Cloud environment.

Submission

- Your submission should contain the python notebook file
- **Zip** file named using the following convention:
 <SU-EMAIL>_<FIRST-Name>_AS4.zip
 Ex. zzhao37_ziyi_AS4.zip
- Upload zip file to blackboard before 11:59PM (EST Time) 3/26/2020