

# Natural language processing report lab 5

## Introduction

The main purpose of this assignment was to implement a neural language model. The assignment required us to download code from a tutorial labelled `word_embeddings_tutorial.py` and reads its relevant documentation attached to it.

## Model description

The language model that was presented before us in this assignment in which it was relatively small and each sentence required us to have a start and end token for the purpose of the neural network. Regarding the epoch loop it seems to me that it has been placed into the model because they are turned into indices which are wrapped into variables. Furthermore, upon a closer inspection it seems like to me that the words are transformed into the word embeddings and the dimensionality of that regards is 10. Due to this these words are defined into a vector size of 1,10. In respect to the input layer of the neural network it could be regarded as context for words, which has the last two words are stored which is kept in a vector dimensionality. The results are placed onto a hidden layer because it is a multiplication of the results vector by the weight vector of dimensionality of 1,128 as they go through the relu function making the outcome non-linear. Moreover, the first result is then multiplied by the second weight vector with the dimensionality 128, 17. The mathematical equation of this particular model is given below:

$$I \cdot W_1 \leftrightarrow (1, 20) \cdot (20, 128)$$

$$O_1 = \text{Relu}(I \cdot W_1)$$

$$O_1 \cdot W_2 \leftrightarrow (1, 128) \cdot (128, 17)$$

$$O_2 = \log \text{softmax}(O_1 \cdot W_2) \leftrightarrow (1, 17)$$

$$P(X_n | X_{n-2}, X_{n-1}) = \text{argmax}(O_2)$$

## Sanity checking

As per requirements of this particular lab sheet I have implemented a sanity check in regards of the model in which I have implemented a loop which examines the sentence of when the mathematician ran to the store. The purpose of this was to determine whether the model predicts the words accurately and with consistency. Examining the hyper parameters, I tried numerous epoch numbers and learning rates. When examining this I found out that an epoch of 85 in conjunction with a learning rate of 0.075 gave me a very good accuracy when trying to run it numerous times. The main reason why the model the model foresees the word mathematician rather than physicist is because regarding the word

mathematician it is typically followed by the word the so therefore, it will have higher log probability.

### **Testing**

Coming towards the last stages of the assignment I was asked to predict which word physicist or philosopher will fill the gap for The \_\_\_\_\_ solved the open problem, I have used a cosine similarity. The cosine similarity was used for measuring if physicist or philosopher is nearer to the word mathematician to fill in the relevant gap. One would predict that physicist should be chosen to fill in the gap because its nearer to the word mathematician. When I tested this numerous of times the word physicist came up majority and nearly all the time but the word philosopher only came up a few times. It will not be possible with the aid of bigram ML model from lab 2, the reason for this is that the bigram ML will not check and go through the words that come quite frequently in terms of the vocabulary so therefore, it will not measure semantic similarity.