**CH5019 Project**

**Group 57**

**Problem statement:** In this project we are supposed to come up with a model that will score the descriptive answers given by students in comparison to the model answer.

**Approach:**

This is a problem of Natural Language Processing (NLP). In this text embedding plays a very major role. Text embedding converts text into numerical vectors.

After the sentences are converted into vectors, we can calculate how close these two vectors our, which in the end will show the similarity of the two vectors thus telling us how much the score should be for the student’s answer. There are a few methods to find out the distance between the vectors such as the Euclidean distance and cosine similarity. In our project we will be using cosine similarity.

In our code we have used Universal Sentence Encoder (USE). It encodes text into higher dimensional vectors that can be used for our semantic similarity task. The pre-trained Universal Sentence Encoder (USE) is publicly available in tensorflow hub.

**Process:**

In the beginning we load all the important libraries such as tensorflow because we are going to use a ML model (USE) and we are also going to use pandas, as we have the input sentences in the form of a dataframe.

Then we import USE as a pretrained model that gives us embedded vectors from the texts, followed by an embedded function to implement them in the code.

After this an important step in the code is to make sure that embedded vectors coming out of the function embedded are of the datatype numpy.ndarray, because we have to use cosine similarity ahead.

After this we run the code for cosine similarity throughout the length of the dataframe to get the scores for every answer written by the student. We will be looking at what cosine similarity is in detail ahead in this report.

After we get a result for all the answers we join those scores to the main dataframe and there you have it, the scores and all the answers in the same dataframe.

**Universal Sentence Encoder (USE):**

The Universal Sentence Encoder encodes text into high-dimensional vectors that can be used for text classification, semantic similarity, clustering and other natural language tasks. The model is trained and optimized for greater-than-word length text, such as sentences, phrases or short paragraphs. It is trained on a variety of data sources and a variety of tasks with the aim of dynamically accommodating a wide variety of natural language understanding tasks. The input is variable length English text and the output is a 512 dimensional vector. The universal-sentence-encoder model is trained with a deep averaging network (DAN) encoder.

We used this model because it gave the numeric vector that we need from the text answers so that we can compare them using cosine similarity.

**Cosine similarity:**

Cosine similarity is a very popular metric to measure text similarity between two text irrespective of their size in Natural Language Processing (NLP). Cosine similarity measures the similarity between two vectors of an inner product space. It is measured by the cosine of the angle between two vectors and determines whether two vectors are pointing in roughly the same direction.

It is given by the formula:

Cosine similarity=

We choose cosine similarity because they are a better metric than the Euclidean distance because if the two-text document far apart by Euclidean distance, there are still chances that they are close to each other in terms of their context.

Thankyou