# Precily Semantic Similarity between two Sentences

The Semantic Similarity Project is an implementation of a model for measuring the semantic similarity between pairs of text sentences. It utilizes advanced Natural Language Processing (NLP) techniques and a state-of-the-art deep learning model to provide accurate semantic similarity scores.

There are various approaches which I tried to do this assessment:-

- Approach 1: -(EX.ipynb)
    - As we can see in the dataset that Precily text Similarity.csv we have approx. 3000 rows dataset in two columns-> text1 and text2 each respectively so ultimately It is a very huge dataset to deal , so as stating the traditional approach to solve this NLP task is as follows:-
        - Read the dataset and store it into a dataframe
        - Convert it into a single format either a lower case or uppercase
        - Check for the null data
        - Preprocess the textual dataset
            1. Remove punctuations
            2. Remove stopwords
            3. lemmetizing
            4. Tokennization
            5. Convert into vectors
            6. Then fetch this vectors into a Custom RNN model to predict the similarity score

Or we can go with another one to feed this vectorized tokens into any Ml model like Logistic Regression to predict the similarity scores

But I could not evaluate these as it tokenizing these is giving the hash table of [3000,9000] so implementing this is a very hard task to do and although I don't have any cuda device to do this.

As we are free to use any kind of approach and any model to implement with, then I tried another approach

- Approach 2:- (Example.ipynb)
    - I have implemented a pretrained model from Hugging face which is

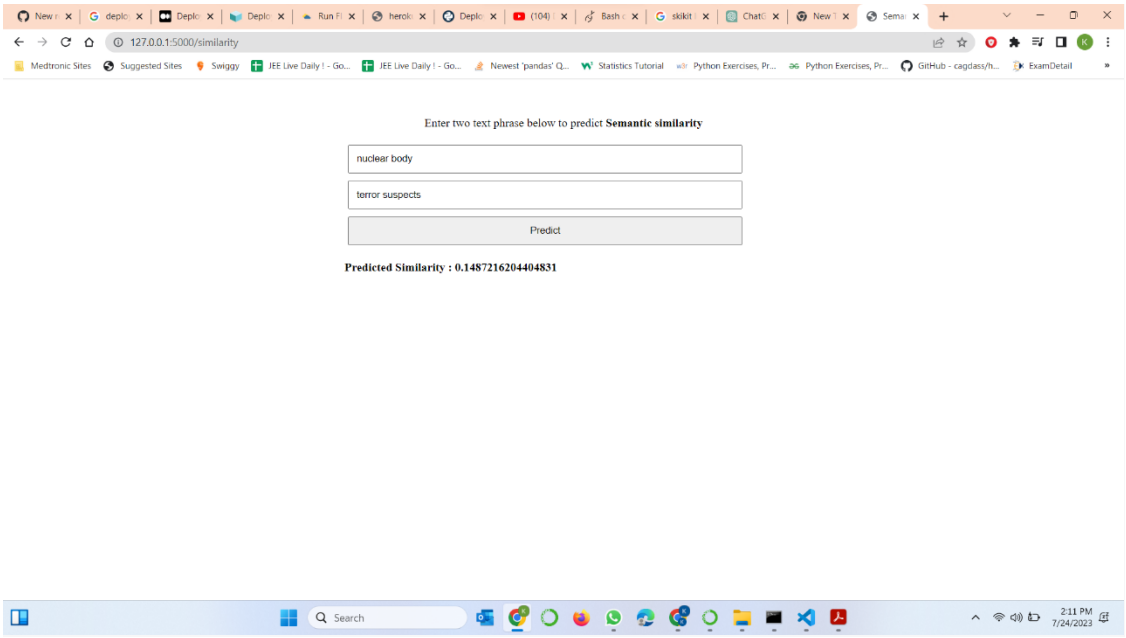
**paraphrase-MiniLM-L6-v2**

The "paraphrase-MiniLM-L6-v2" model is a compact and efficient language model designed specifically for the task of paraphrasing. It is based on the MiniLM architecture, which is a smaller variant of the popular BERT (Bidirectional Encoder Representations

from Transformers) model. This pre-trained model has been fine-tuned on extensive paraphrase datasets to generate high-quality paraphrased sentences.

**Key Features:**

- **Paraphrasing Capability:** The model excels at generating semantically similar sentences, making it ideal for various applications such as text augmentation, data generation, and paraphrase identification tasks.
- **Efficiency:** Being a smaller variant of BERT, the paraphrase-MiniLM-L6-v2 model offers fast inference times and reduced memory requirements without compromising on paraphrasing performance.
- **Easy Integration:** With a user-friendly Python API, integrating the model into existing projects is straightforward, enabling seamless use in various Natural Language Processing (NLP) pipelines.
- According to hugging face this has the Accuracy approximately -> 60 %
- After that feeding this into the dataset to create word embeddings and then predicting by the cosine similarity to get the similarity score
- Then saving this model to pickle file and then from this trained model with this dataset we can predict for the text and custom input data to check the similarity score in (app.py)
- Then hosted this into api endpoint via flask and Jscript



- For the deployment service in an cloud service:-
  - I tried doing in many of the cloud services but I could not implement it like AWS like this is the third time but could not get success.