# Fine-tuning Assignment 7

Several fine-tuning adjustments have been made to enhance the accuracy of the transformer-based language model. Let's explore these adjustments:

## 1. Label Mapping and Frequency Filtering:

The code introduces a mechanism for mapping labels into a new set and filters out labels based on their frequency. By using the label_map dictionary, it maps labels to integers and filters out less frequent labels, effectively reducing noise in the training data. This preprocessing step helps focus the model on the most relevant and frequently occurring classes, potentially improving accuracy.

## 2. Tokenization:

Tokenization is a critical step in preparing data for a transformer-based model. The code uses the Hugging Face AutoTokenizer to tokenize the input data, ensuring it is in a format suitable for training with the distilBERT model (distilbert-base-cased). The tokenization process includes padding sequences to a maximum length and truncating longer sequences.

## 3. Tokenization:

Tokenization is a critical step in preparing data for a transformer-based model. The code uses the Hugging Face AutoTokenizer to tokenize the input data, ensuring it is in a format suitable for training with the distilBERT model (distilbert-base-cased). The tokenization process includes padding sequences to a maximum length and truncating longer sequences.

## 4. Logging and Evaluation:

The Trainer from Hugging Face's transformers library is employed for training and evaluation. The evaluation is performed after each epoch (evaluation_strategy="epoch") to monitor the model's progress. The compute_metrics argument is set to the accuracy function, which computes the accuracy metric. The logs are saved to the "./logs" directory, facilitating analysis of training progress.

## 5. Stability and Model Saving:

The model is trained with a train-test split of 90-10%. After training, the model's performance on the test set is evaluated using the trainer.evaluate() method. The model is then saved to the "finetuned.model" directory for later use or deployment

The observed increase in accuracy from 0.88 after the first epoch to a stabilized 0.9 after the second epoch can be attributed to the above adjustments made to the model.