**HW5Chapter 4: Transformers**

**Introduction:**

In this assignment, you will implement Transformers model step-by-step by referencing the original paper, (https://arxiv.org/pdf/1706.03762.pdf). You will also use a toy dataset to solve a vector-to-vector problem which is a subset of sequence-to-sequence problem.

**Assignment Overview:**

This assignment consists of four parts, with detailed instructions and testing functions provided in `Transformers.ipynb`. The implementation should be done by modifying the necessary functions and classes in `transformers.py`. Follow the notebook's guidelines carefully to ensure correctness.

In this notebook, you will learn how to implement an Encoder-Decoder based Transformers in a step-by-step manner. We will implement a simpler version here, where the simplicity arise from the task that we are solving, which is a vector-to-vector task. This essentially means that the length of input and output sequence is \*\*fixed\*\* and we dont have to worry about variable length of sequences. This makes the implementation simpler.

1. Part I (Preparation): We will preprocess a toy dataset that consists of input arithmetic expression and an output result of the expression

2. Part II (Implement Transformer blocks): we will look how to implement building blocks of a Transformer. It will consist of following blocks

1. MultiHeadAttention

2. FeedForward

3. LayerNorm

4. Encoder Block

5. Decoder Block

3. Part III (Data Loading): We will use the preprocessing functions in part I and the positional encoding module to construct the Dataloader.

4. Part IV (Train a model): In the last part we will look at how to fit the implemented Transformer model to the toy dataset.

You can run all things on CPU. But, it is recommended to run on GPU for faster training.

**Submission:**

* Submit the Jupyter Notebook as your assignment. Remember to keep the output of cells of the notebook.
* Ensure all required tasks are completed and all code is functional.

**Notes:**

* Use the course forum or reach out to the instructor and TAs for any questions.
* Ensure your code is well-documented and readable.