Indian Institute of Technology Hyderabad
Deep Learning (AI2100/AI5100/EE6380): Assignment-3

Topic: Word2Vec and Vision Transformers
Assigned on: 22^{nd} **March, 2025**
Deadline: 31^{st} **March, 2025**
Maximum Marks: 50

1 Instructions

- Answer all questions. We encourage best coding practices by not penalizing (i.e., you may not get full marks if you make it difficult for us to understand your submission. Hence, use intuitive names for the variables, functions, etc., and comment your code liberally. You may use the text cells in the notebook to briefly explain the objective of a code cell.)
- It is **expected** that you work on these problems individually. If you have any doubts please contact the TA or the instructor no later than 2 days before the deadline.
- You may use built-in implementations only for the basic functions such as `sqrt, log`, etc. from libraries such as `numpy` or `PyTorch`. Other high-level functionalities are expected to be implemented by the students. (Individual problem statements will make this clear). For plots, you may use `matplotlib` and generate clear plots that are complete and easy to understand.
- You are expected to submit the Python Notebooks saved as A3_<your-roll-number>.ipynb. If you are asked to report your observations, use the mark down text cells in the notebook.

2 Problems

1. **Word Representation using Word2Vec:** Implement the skip-gram model with negative-sampling loss function for word embedding generation. Your implementation should include: [20 Marks]
   (a) Download and preprocess the text8 dataset, including tokenization, cleaning (e.g., lowercasing, punctuation removal), vocabulary creation. [3]
   (b) Implement the skip-gram model from scratch with negative sampling loss. [4]
   (c) Derive and implement the gradients for backpropagation. [4]
   (d) Train your model on the text8 dataset with appropriate hyperparameters (specify your choices and justify them). [3]
   (e) Evaluate the quality of your embeddings through: [4]
      - Visualization using SVD to project the embeddings to 2D space.
      - Word similarity analysis for semantically related words (e.g., "king" - "man" + "woman" ≈ "queen").
   (f) Discuss the impact of key hyperparameters (e.g., embedding dimension, context window size, number of negative samples) on the quality of the learned representations. [2]
2. **Vision Transformer: Analyzing Attention and Performance on CIFAR-10**: Implement a Vision Transformer on CIFAR-10, analyze its attention maps and performance, and experiment with hyperparameters to evaluate their impact.
   (a) Implement an Encoder only Vision Transformer (ViT) using PyTorch on the CIFAR-10 dataset, including key components such as patch embedding, positional embeddings, the Transformer encoder with multi-head self-attention, and a classification head using the CLS token. [10 marks]
   (b) Set up the training procedure for your ViT model on the CIFAR-10 dataset, and analyze its performance (e.g., accuracy, loss) to evaluate how well your model learns. [6 marks]

(c) Visualize and carefully analyze attention maps from different layers and attention heads of your transformer to identify patterns and insights. Evaluate how effectively your ViT captures global context and spatial relationships within images through attention maps. [8 marks]

(d) Experiment by varying hyper parameters such as patch size, number of transformer layers, and attention heads to observe their impact on attention patterns and model performance. Discuss your findings clearly with supporting visualizations and examples. [6 marks]