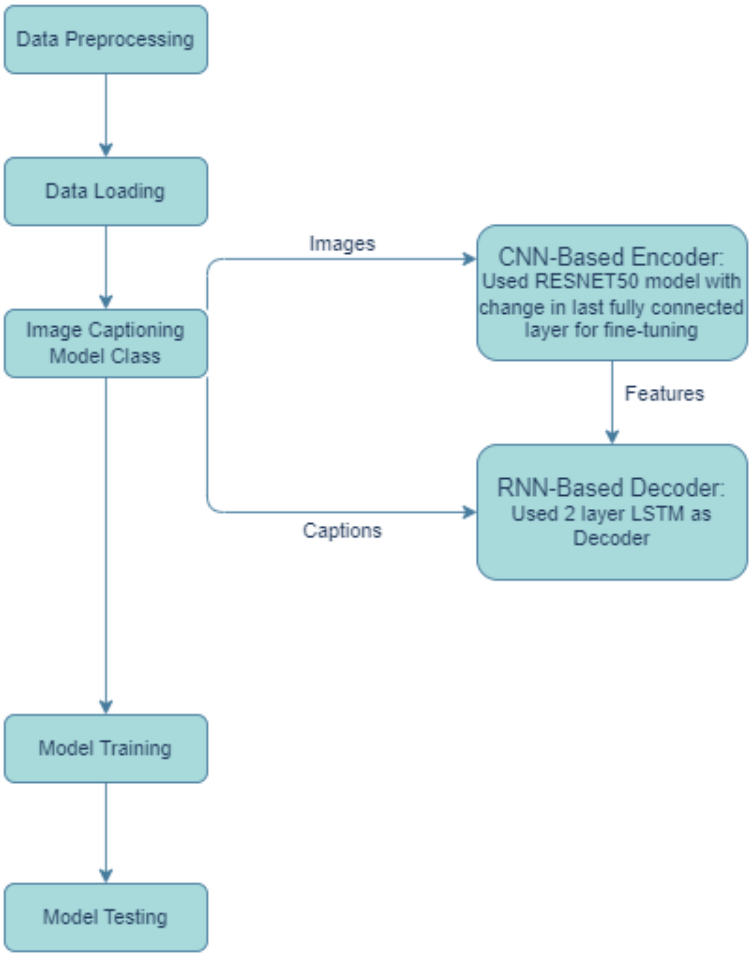
Deep Learning (CS60010)

Term Project Report

Methodology:

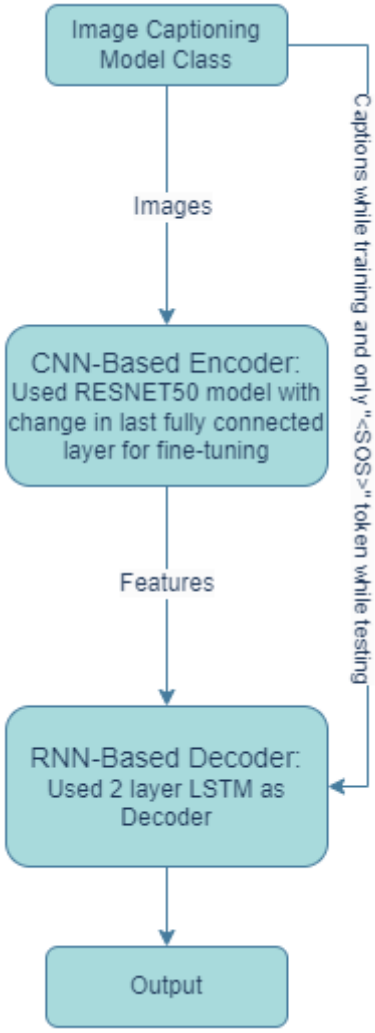
Part A

For the Part A, For CNN Based Encoder we have used ResNet50 model with change in the last fully connected layer and fine-tuned it on the given train dataset. For RNN Based Decoder we have used 2 layer LSTM. Here is the flow diagram representing the flow in which Part A is implemented.



Data Preprocessing:
For Training images, we have resized the images to (300, 300,3) and randomly cropped the image to (224,224,3), then converted the pixel values in image to (0,1) and normalized the images using mean = [0.485, 0.456, 0.406], standard deviation = [0.229, 0.224, 0.225].

For Testing and Validation images we have resized the images to (224,224,3), then converted the pixel values in image to (0,1) and normalized the images using mean = [0.485, 0.456, 0.406], standard deviation = [0.229, 0.224, 0.225].
For captions, the vocabulary has been built by tokenizing each sentence in training data and counting the frequency of each word. If a word's frequency reaches the threshold of 5, it is added to the vocabulary. Then converted the caption into a list of numerical values based on the vocabulary. If a word is not in the vocabulary, it is replaced with the <UNK> token.
Here is the Flow Diagram of Model Class:



Part B

For preprocessing of data, we have applied all techniques used in Part A along with ViTImageProcessor from hugging face library to preprocess the images to the desired format for Vision Transformer.

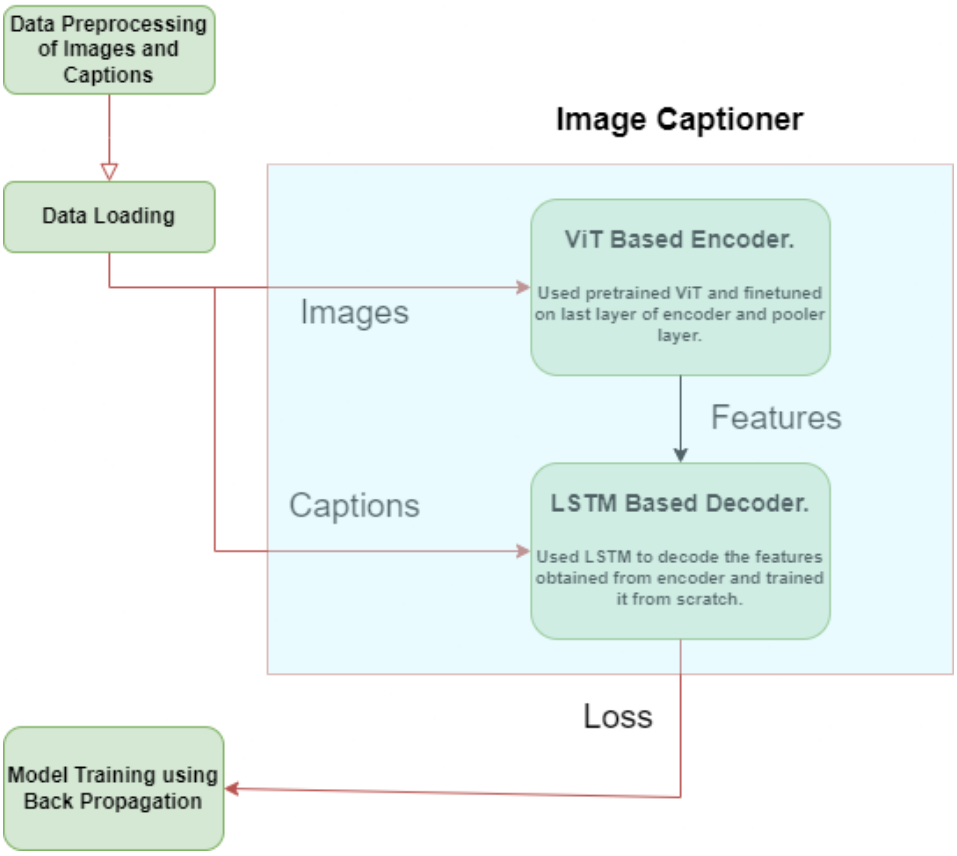
We have used pretrained Vision Transformer as encoder by obtaining the pretrained model and weights from hugging face library and used the pretrained weights of 'google/vit-

base-patch16-224' model which is trained on ImageNet dataset. We have also used. To finetune the encoder, we have set the parameters of last layer of ViT encoder and ViT pooler layers to require_grad = True and all other parameters are pretrained. From the outputs of the ViTModel, we have considered to take only CLS TOKEN and used it to decode the caption. We have used an embedding layer to embed the CLS TOKEN to the embedding size used in decoder.

For Decoder, we have considered using two-layer LSTM. All parameters are trained from scratch using the dataset given and then we have used the Linear layer to project hidden size in LSTM to the size of vocabulary which is further used to obtain logits and find the probability of the next word.

We have used Adam optimizer with learning rate = 3e-4 and CrossEntropyLoss as loss function and trained it for 300 epochs. Then using the predicted words, we have generated predicted captions and then used actual captions and predicted captions to evaluate our metrics on the test dataset.

The following diagram gives an insight of the process of the training of our model.



Both models are trained using Cross Entropy as loss function and Adam as optimizer. The following are the results obtained on the Test set.

Results:

Part A

Evaluation metric	Score
CIDEr	0.06497638178441448
BLEU	0.22724780555641888
ROUGE-L	0.20923209441809068

Part B

Evaluation metric	Score
CIDEr	0.06277697878634352
BLEU	0.23319887014854587
ROUGE-L	0.21086168278323710

Evaluation Metrics and Analysis:

- CIDEr (Consensus-based Image Description Evaluation) focuses on the saliency, correctness, and relevance of generated captions by comparing n-grams.
- BLEU (Bilingual Evaluation Understudy) measures the precision of predicted captions by comparing them with reference captions at various n-gram levels.
- ROUGE-L (Recall-Oriented Understudy for Gisting Evaluation) assesses the longest common subsequence between the predicted and reference captions, reflecting the fluency and the order of the content.

1. CIDEr Performance:

- A slightly outperforms B. This could indicate that the ResNet50 encoder is better at capturing the more salient and relevant features needed for generating captions that align well with human judgments of quality.

- The small difference suggests that both models are somewhat effective but still struggle with fully capturing the gist in a way that humans would agree strongly on.

2. BLEU Performance:

- B has a higher BLEU score than A. This suggests that the Vision Transformer model might be better at matching specific n-grams with the reference captions, possibly due to its ability to capture broader and more varied context across the whole image via its self-attention mechanisms.

3. ROUGE-L Performance:

- B also slightly outperforms A in ROUGE-L, suggesting it might be generating captions that are more fluent and maintain more of the sentence structure seen in the reference captions. This could be attributed to the global contextual understanding provided by the ViT.