Part A: CNN and RNN for image captioning

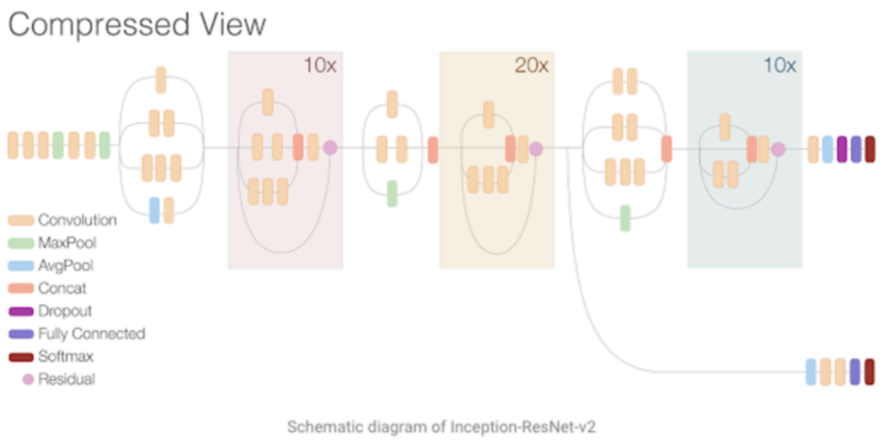
Methodology:

CNN:

For CNN, we used a pretrained inception-resnet-v2 model with a custom linear layer in place of the classification layer. It has 164 layers and is trained on over a million images from the ImageNet database. It is a convolutional neural architecture that builds on the Inception family of architectures but incorporates residual connections (replacing the filter concatenation stage of the Inception architecture).

Preprocessing :

For image data, normalization was applied to match the input requirements of the Inception-ResNet-v2 model. Images were resized to 299×299 pixels. For text data, captions were tokenized and padded to create uniform sequence lengths, and a vocabulary was built from the training set of captions.



Schematic representation of the Inception-ResNet-v2 architecture, depicting the arrangement of convolutional, pooling, and fully connected layers alongside residual connections. This compressed view highlights the model's repeated modules, which are key to its deep learning capabilities for image recognition tasks.

RNN:

For RNN, we used the original LSTM cell with 3 layers and no self-attention. Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN) that can process and analyze sequential data. LSTMs are used in deep learning and are designed to solve the vanishing gradient problem that traditional RNNs and machine learning algorithms face.

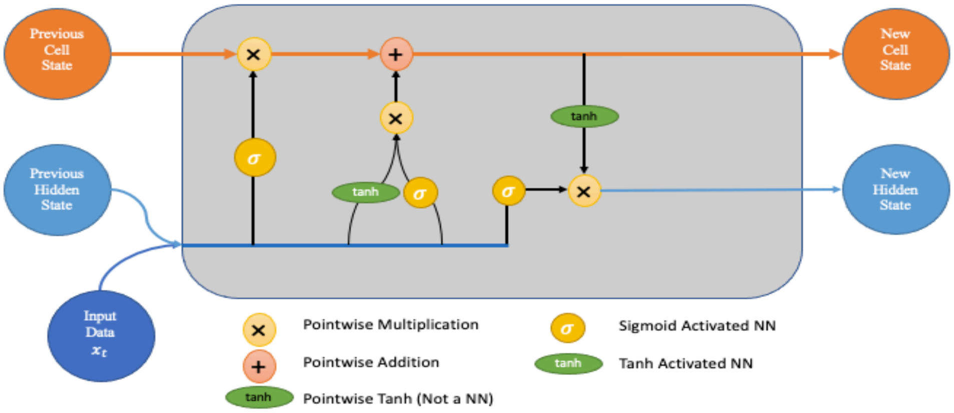


Diagram illustrating the internal structure of a Long Short-Term Memory (LSTM) cell, showcasing the flow of information through various gates (input, output, and forget), which enable the cell to maintain and modify its state over time. The interplay of operations like pointwise multiplication and addition ensures the cell's capability to remember long-term dependencies and forget irrelevant data.

Captioning Model:

Our captioning model passes the image through the CNN mentioned above, and the output of the CNN is passed through the RNN to autoregressive generate the captions for the image. Teacher forcing is used during training. Caption length during inference has been limited to 50 words.

Results:

The ROUGE_L score indicates the model's ability to capture phrase-level accuracy, which is relatively low, suggesting difficulty in forming longer, coherent phrases. SPICE and CIDEr scores, which are low, reflect challenges in capturing the finer semantic details of the images. Comparisons with baseline models (not included here) could provide further insights into these metrics.

Error Analysis: Examination of errors revealed common issues with specific object recognition and contextual relevance in captions. These misinterpretations often occur in complex scenes with multiple objects.

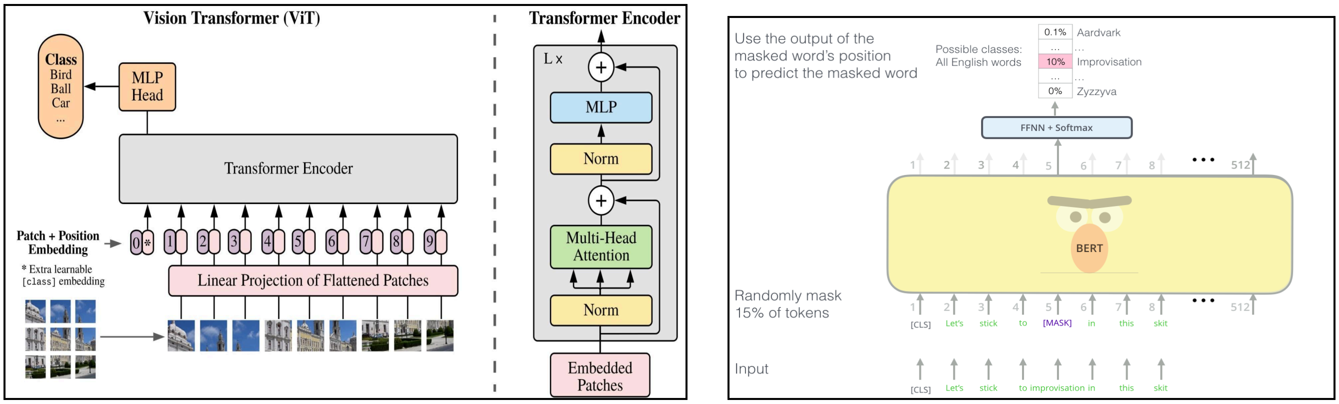
Metric	Score
ROUGE_L	0.17982118138532383
SPICE	0.030993521969669237
CIDEr	0.00709378732477089

Part B: Vision Transformer and Bert Model for Image Captioning

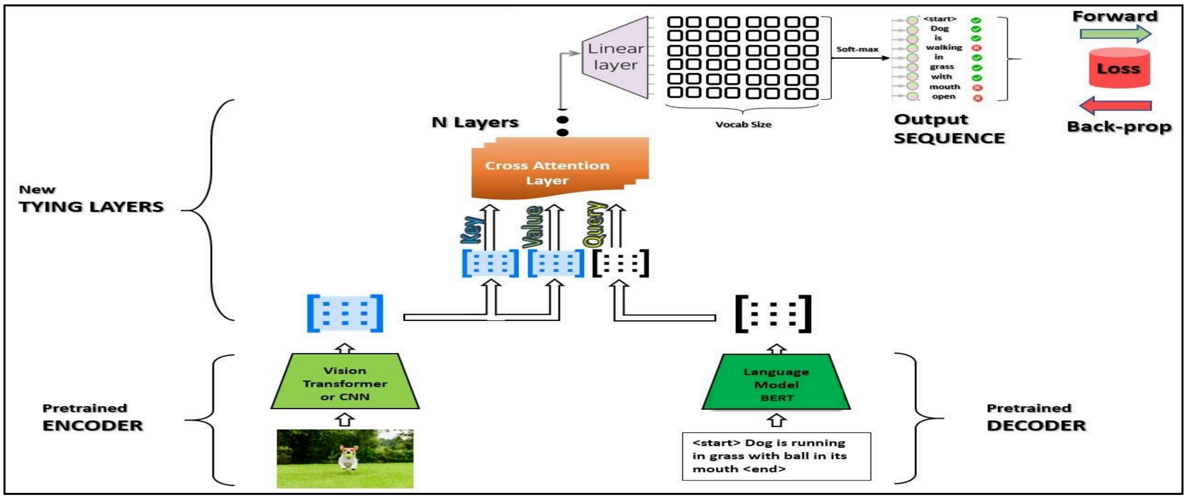
Methodology:

- **Vision Transformer (ViT) and BERT Tokenizer**: For the encoder, we utilized Google's ViT-base-patch16-224 model, which is designed to process images into patches and then encode these patches similarly to words in NLP. The decoder uses a BERT-based model from Google's Bert-base-uncased, which is fine-tuned to generate captions based on the features extracted by ViT.

- **Training Data and Preprocessing**: The dataset consists of images with associated captions. Images are processed through the ViTImageProcessor, which prepares them for the transformer model. Captions are tokenized using BertTokenizer, ensuring uniform sequence lengths for training stability.


Vision Transformer (ViT)
Class
Bird
Ball
Car
...
MLP Head
Transformer Encoder
Patch + Position Embedding
* Extra learnable [class] embedding
0 *
1
2
3
4
5
6
7
8
9
Linear Projection of Flattened Patches
Transformer Encoder
L x
+
MLP
Norm
+
Multi-Head Attention
Norm
Embedded Patches
Use the output of the masked word's position to predict the masked word
Possible classes: All English words
0.1% Aardvark
...
10% Improvisation
...
0% Zyzzyva
FFNN + Softmax
1 2 3 4 5 6 7 8 ... 512
BERT
Randomly mask 15% of tokens
[CLS] Let's stick to [MASK] in this skit
Input
[CLS] Let's stick to improvisation in this skit

Left: The Vision Transformer (ViT) architecture dissects input images into patches and enriches them with positional information before linear projection. These processed patches then journey through a Transformer Encoder, where multi-head attention mechanisms refine their contribution to the model's potent image recognition functions. **Right:** The BERT model employs masked language learning, where a random 15% of input tokens are concealed, prompting BERT to predict these words by analyzing the context. A feedforward neural network and a softmax layer collaborate to deduce probable word choices from the vast English lexicon.



Architectural blueprint of an encoder-decoder model for image captioning. This model uses a pre-trained Vision Transformer (ViT) or CNN as an encoder to analyze image data, with subsequent cross-attention layers that integrate visual and textual information. The decoder, a pretrained language model such as BERT, generates descriptive captions. The diagram outlines the complete process from image input to caption output, including the softmax layer that converts logits to probabilities for each word in the vocabulary.

Results and analysis:

Metric	Score
ROUGE_L	0.20605297669022243
SPICE	0.12814811400282447
CIDEr	0.09611562590177923

Analysis:

The performance metrics for the Vision Transformer and BERT model indicate a substantial improvement over the traditional CNN-RNN architecture used in Part A. The use of a transformer-based approach, particularly the integration of the Vision Transformer (ViT) for image processing and BERT for text generation, suggests that the model benefits from advanced self-attention mechanisms that better capture the complexities of both visual perception and language semantics.

- **CIDEr Score Analysis:** The CIDEr score of 0.0961, although modest, is significantly higher than that achieved in Part A. This suggests that the captions generated are more aligned with those created by humans, particularly in terms of capturing the salient features that are relevant for describing the images.
- **SPICE Score Analysis**: The SPICE score of 0.1281 reflects improvements in the semantic understanding of the images. This is indicative of the model's ability to construct more meaningful and contextually appropriate descriptions, a critical aspect of natural language generation in image captioning.
- **ROUGE Score Analysis**: The ROUGE score of 0.2060 shows that there is a reasonable overlap between the words in the generated captions and those in the reference data. This score highlights the model's capability in replicating human-like syntactic structures and keyword usage.

Comparative Analysis:

Comparing these results to those of Part A, the advanced architecture of Part B leverages deeper contextual embeddings and a sophisticated attention mechanism, providing a more nuanced understanding of both the visual content and its appropriate linguistic description. These attributes are crucial for the generation of coherent and contextually rich captions, which are essential for applications in automated media content generation and assistive technologies.

Recommendations for Future Work:

To further enhance the performance:

- **Model Ensemble**: Combining the strengths of different architectural approaches, such as a hybrid model that integrates elements from both CNNs and transformers, could yield better results.
- **Hyperparameter Tuning**: More extensive experiments with training parameters and model configurations may lead to optimizations that improve the metrics further.
- **Dataset Expansion:** Utilizing a more diverse and larger dataset could help the model learn more generalized features and reduce overfitting.