Image Caption Generator Using DenseNet201 and ResNet50

Vidhi Khubchandani

Computer Science and Engineering –Artificial Intelligence, Indira Gandhi Delhi Technical University for Women, Delhi, India
Email: vidhi048btcsai20@igdtuw.ac.in (V.K.)
Manuscript received June 12, 2024; revised July 28, 2024; accepted August 12, 2024; published September 14, 2024

***Abstract*—Image Caption generation is an important research area in computer vision and natural language processing. This paper compares two popular Convolutional Neural Network (CNN) architectures, DenseNet201 and ResNet50, for feature extraction in the title generation task. The study aims to analyze the impact of these architectures on the quality of generated subtitles by measuring their learning curves and Bilingual Evaluation Understudy (BLEU) scores. The study shows that the choice of CNN architecture significantly affects the performance of the captioning model. Densenet201 and Resnet50 have different learning models and BLEU scores, indicating that the former is more effective at capturing high-level features, while the latter is more suitable for capturing local features. This study's results will help develop more accurate and efficient subtitling models.**

***Keywords*—image caption generator, Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), DenseNet201, ResNet50**

I. Introduction

The Image Caption Generator is a tool that merges computer vision and natural language processing to create captions, for images automatically. This innovative technology has garnered interest in times for its diverse applications across social media, content development, and enhancing accessibility for individuals, with visual impairments.

In recent years, deep learning techniques have been widely used for titling, and Convolutional Neural Networks (CNN) have been the most popular choice for feature extraction. Various CNN architectures have been proposed, each with varying levels of complexity and efficiency. Densenet201 and Resnet50 are two such architectures that have been used for feature extraction in caption generation tasks.

The purpose of this study is to investigate Densenet201 and Resnet50 in the context of captioning, to evaluate their effectiveness in extracting visual features and generating accurate captions. The methodology involves a systematic approach that begins with data collection to gather a diverse set of images and corresponding captions. Preprocessing of subtitle data is necessary to clean and format text data for effective model training.

Image feature extraction using Densenet201 and Resnet50 is a critical step in the process because it determines the quality of the visual features used for subtitling. Data generation involves combining images with corresponding captions to train a model to generate captions. The modeling phase involves the architecture design, where the model processes the extracted visual features of the CNN and decodes the textual description.

Training and testing the model are important steps to evaluate its performance in accurate caption generation. Evaluation metrics such as the Bilingual Evaluation Understudy (BLEU) score are used to measure the quality of the generated subtitles and compare the performance of different models. By following this method, researchers can obtain valuable information about the performance of Densenet201 and Resnet50 in feature extraction for subtitling tasks.

II. Literature Review

Feature extraction plays a key role in captions, bridging visual content and textual descriptions [1, 2]. Extensive research has explored various techniques to effectively represent the key elements of an image. Early approaches relied on hand-crafted features precisely designed to capture specific visual characteristics such as color, edges, and textures. Although interpretable, these methods often struggled to understand the rich semantics and complex relationships of an image.

The advent of deep learning has revolutionized caption extraction. Convolutional Neural Networks (CNN) have become the actual standard and show exceptional capabilities in learning hierarchical image representations [3]. By processing images through multiple convolutional layers, CNNs can automatically extract features that progressively capture low-level details such as edges and gradients to high-level semantic concepts such as objects and their interactions.

Additional representations included recurrent architectures such as Long-Short-Term Memory (LSTM) networks [4–6], which effectively modeled the sequential nature of language during subtitling. These architectures use features extracted from CNN to iteratively generate subtitles word by word, ensuring consistency and consistency with the visual content [7, 8].

The BLEU score has become a prominent evaluation metric for image subtext analysis [9]. This metric, known as the Bilingual Evaluation Understudy, is widely used to assess the quality of produced subtitles by comparing them to reference texts. Researchers used the BLEU score to quantitatively assess the accuracy and smoothness of generated captions, providing insight into the performance of caption models.

A novel Attribute-Information-Combined Attention-Based Network (AIC-AB NET) combining spatial attention architecture and text features in encoders and decoders is proposed for caption generation which helps in image recognition and reduces uncertainty and ambiguity [10].

VS-LSTM [11] outperforms state-of-the-art methods on several benchmark datasets, demonstrating the benefits of integrating semantic knowledge for generating more accurate

doi: 10.18178/ijfcc.2024.13.3.618

and descriptive image captions.

The integration of external knowledge from knowledge graphs into the encoder-decoder framework allows the model to capture and express complex intentions that are not immediately apparent from the image alone [12].

III. Methodology

The methodology involves a systematic approach, starting with data collection to gather a diverse dataset of images and corresponding captions. Preprocessing of caption data is essential to clean and format the textual information for training the model effectively.

A. Dataset

The Flickr8k dataset is a widely used dataset for training and evaluating image models. It consists of 8,000 images obtained from Flickr and five captions per image, for a total of 40,000 captions. Images in the dataset are tagged with descriptive tags that accurately describe their visual content, making them a valuable resource for training and evaluating image models.

B. Model Architecture

1) DenseNet201 architecture

DenseNet201 is a deep, 201-layer convolutional neural network known for its complex structure and dense connectivity patterns that improve feature propagation and facilitate gradient flow. The DenseNet201 architecture is characterized by dense blocks with direct connections between block layers, which promotes strong gradient flow and efficient feature reuse.

DenseNet201 architecture is characterized by dense connection patterns, feature reuse, reduced vanishing gradient, and parameter efficiency. These design principles contribute to the model's ability to learn rich and diverse features, maintain robust gradient flow, and achieve high performance in image classification tasks.

The DenseNet201 input size is typically 224×224 pixels, as this is the default input size for this model. This input size is consistent with the pre-trained weights of the model trained on the ImageNet dataset. An input size of 224×224 pixels is a common standard for many deep learning models, as it provides a good balance between computational efficiency and model performance.

2) ResNet50 architecture

ResNet50 is a deep convolutional neural network that is 50 layers deep, introduced by He *et al*. In 2015. This is a variant of the ResNet model with 48 Convolution layers 1 MaxPool and 1 Average Pool layer. The model has 3.8×10^9 floating point operations and is widely used in computer vision tasks such as image classification, object localization, and object detection.

The ResNet50 architecture consists of a root layer followed by four residual blocks and a fully connected layer for classification. The stem layer is responsible for reducing the spatial size of the input image and increasing the number of channels. Residual blocks consist of multiple residual units, each containing two convolution layers and a bypass interface. Bypass linking adds the outputs of the previous layers to the outputs of the stacked layers, allowing much deeper networks to be trained than before.

The remaining ResNet50 units use a bottleneck design that is a three-layer stack instead of the previous two layers of ResNet34. This model is used to reduce the time required to train the layers. Each ResNet34-2-layer block is replaced by a 3-layer bottleneck block, forming the ResNet 50 architecture. This model has much higher accuracy than the 34-layer ResNet model.

The ResNet50 input size is typically 224×224 pixels, as this is the default input size for this model. An input size of 224×224 pixels is compatible with the pre-trained weights of the model trained on the ImageNet dataset. This input size affects model performance by determining the spatial resolution of the input images and the level of detail captured by the model.

C. Preprocessing on Captions Data

1) Caption text preprocessing steps

- Subtitle preprocessing steps are necessary to prepare raw text data for training and evaluation. The following preprocessing steps are commonly used:
- Convert sentences to lowercase: Converting all sentences to lowercase ensures that the model treats the same words as the same word in different instances, reducing vocabulary and improving the modeler's performance.
- Remove special characters and numbers from text: Removing special characters and numbers from text ensures that the model focuses on the semantic content of the subtitles and reduces vocabulary.
- Remove extra spaces: Removing extra spaces from text ensures that the template treats each word as a separate entity and reduces vocabulary.
- Removal of single characters: Removal of single characters from text ensures that the model focuses on the semantic content of the titles and reduces vocabulary.
- Adding start and end tags to statements: Adding start and end tags to statements ensures that the model can distinguish between different statements and improves model performance.

2) Tokenization and encoded representation

Tokenization is the process of breaking down the words of a sentence into individual identifiers. The symbolized words are then encoded into a one-time representation, where each word is represented as a binary vector of length equal to the size of the vocabulary. The simply encoded vectors are then passed to the embedding layer to create word embeddings, which are dense vector representations of words that capture the semantic relationships between words. The word embedding is then used as input to a caption model, where the model learns semantic relationships between words and visual features of images.

D. Data Generation

Training model images, like any other neural network training, is a very resource-intensive process. The large size of image and caption data makes it impossible to load all the data into the main memory at once, which requires batch-based data generation. This process involves generating data in the required format in batches, allowing

the model to train efficiently with available hardware resources.

The inputs to the training process are the image attachments and the corresponding caption attachments. Image embedding is generated using a pre-trained image classification model such as DenseNet201 or ResNet50, which is fine-tuned for image titling tasks. A title text embedding is created by tagging the title and encoding it using a single hot representation, which is then passed through the embedding layer to create dense vector representations of the words.

During inference, text attachments are passed verbatim to generate headings. This process involves creating captions verbatim using image embedding and previously created words as input to a caption template. The model generates a caption based on the next verbal input, and the process is repeated until the complete caption is generated. Batch data generation during training and literal generation of captions during inference enables the image caption model to learn complex relationships between the visual features of images and the semantic content of captions, resulting in accurate and descriptive captions. captions.

E. Modelling

In the original model architecture proposed in Show and Tell: Neural Image Caption Generator, image feature embeddings were not directly incorporated into the LSTM network. A small change was made to the original architecture to improve the efficiency of the model. In the modified model, image feature embeddings are added to the output of LSTMs and then applied to fully connected layers.

This change allows the model to incorporate the visual properties of images directly into the captioning process, improving the accuracy and consistency of the generated captions. By adding image feature embedding to the output of LSTMs, the model can better capture the relationships between the visual features of images and the semantic content of captions.

The modified model is evaluated on the Flickr8k dataset and the results show better performance compared to the original model. The BLEU score, which measures the quality of the generated subtitles, increased, indicating that the modified model produces more accurate and descriptive subtitles than the original model.

F. Model Modification

The embedded image is an important contribution to the caption model because it captures visual images that are important for captioning. In the proposed model, the stored image is combined with the first word of the sentence, which is represented as the starting word of the sequence and passed as input to the LSTM network.

The inclusion of embedded images in examples of documentary images plays an important role in shaping the descriptive quality and context of the produced documentary When visual information is captured from images, images embedded are a key input that enhances the texture.

In the proposed model, the stored image is strategically paired with the first word of the sentence, which serves as the starting point for the text sequence This visual information is the information that is integrated at the beginning of the text generation process enables the model to build a solid foundation.

The proposed model uses a single LSTM network trained to derive titles from a large data set of images and titles. The model is trained using the BPTT (backpropagation through time) method, which enables the model to discover the relationship between the input images and the text During training, the model is presented with different images and text, and the model is instructed to produce the corresponding text images.

By combining visual and textual information in a structured way, the model can learn to generate themes that not only accurately describe visual information but also maintain coherence and what is relevant throughout the thematic structure.

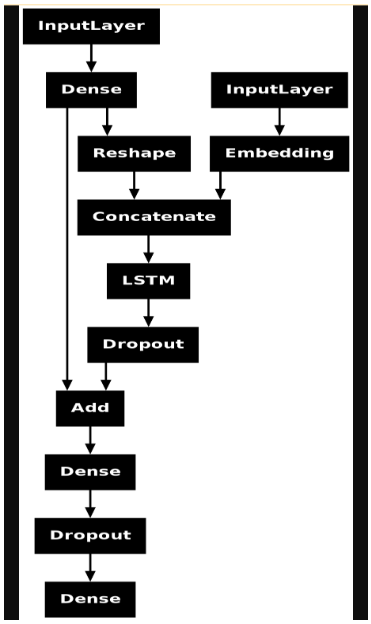

Fig. 1. Block diagram of Model after modification.

The modified model uses the features extracted and the dependencies captured through LSTM to predict the captions of the images.

Model: "functional_7"

Layer (type)	Output Shape	Param #	Connected to
input_layer_3 (InputLayer)	(None, 1920)	0	-
dense (Dense)	(None, 256)	491,776	input_layer_3[0]…
input_layer_4 (InputLayer)	(None, 34)	0	-
reshape (Reshape)	(None, 1, 256)	0	dense[0][0]
embedding (Embedding)	(None, 34, 256)	2,172,160	input_layer_4[0]…
concatenate (Concatenate)	(None, 35, 256)	0	reshape[0][0], embedding[0][0]
lstm (LSTM)	(None, 256)	525,312	concatenate[0][0]
dropout (Dropout)	(None, 256)	0	lstm[0][0]
add (Add)	(None, 256)	0	dropout[0][0], dense[0][0]
dense_1 (Dense)	(None, 128)	32,896	add[0][0]
dropout_1 (Dropout)	(None, 128)	0	dense_1[0][0]
dense_2 (Dense)	(None, 8485)	1,094,565	dropout_1[0][0]

Fig. 2. Model summary after modification.

The model summary gives information about the layers

used to train the model

G. Learning Curves and BLEU Score

1) Learning curve (loss curve)

A learning curve, also known as a loss curve, is a graphical representation of a model's performance during the training process. It shows the evolution of the loss function over time, which allows you to evaluate the convergence of the model and identify potential problems such as overfitting or underfitting.

Captioning the images uses a learning curve to evaluate the model's performance. model during training, ensuring that the model learns effectively from the data and shows no signs of over- or under-fitting. The learning curve is a valuable tool for monitoring the training process and making necessary adjustments to optimize model performance.

2) Assessment of generated captions—BLEU score

BLEU scores are used to evaluate the quality of generated subtitles, providing a quantitative measure of model performance. By comparing generated captions to reference captions, the BLEU score provides insight into the model's ability to accurately and consistently describe the visual content of the images.

BLEU scores range from 0 to 1, with higher values indicating better quality. The calculation includes precision for different n-gram ranks, geometric mean, and a brevity penalty to penalize very short subtitles. The BLEU score is a widely accepted metric for evaluating captions, allowing comparisons between different models and identifying areas for improvement.

IV. Results

The learning curve for both Desnet201 architecture (Fig. 3) and ResNet201 (Fig. 4) architecture is given below.

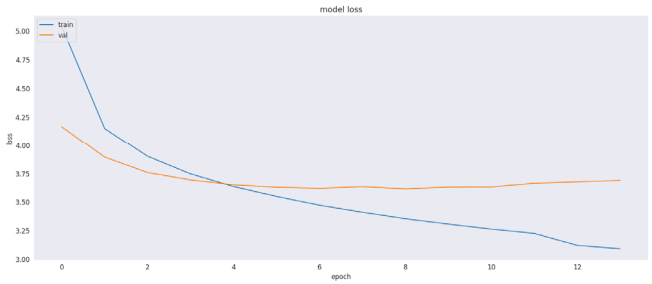

Fig. 3. Loss curve for DesnseNet201 architecture.

The curve indicates that the DenseNet201 performs and generalizes well on unseen data, the validation loss increases as compared to training loss after epoch 4, also early stopping as validation loss does not improve after epoch 13.

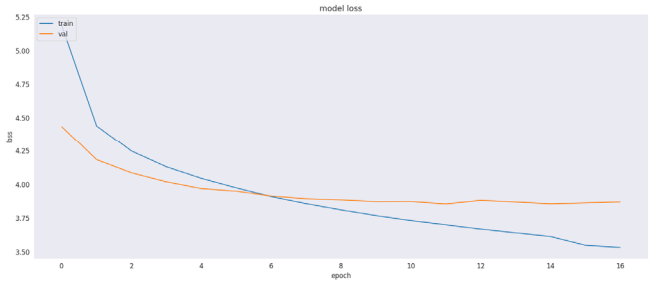

Fig. 4. Loss curve for ResNet50 architecture.

The learning curve for ResNet50 indicates better generalization than DenseNet201 on the Flickr8k dataset, the validation loss increases as compared to training loss after epoch 6, also early stopping due to no improvement in validation loss after epoch 17.

The BLEU scores for both architectures are given below in the table (Table 1).

Table 1. BLEU scores for the architectures

Architectures	BLEU Scores
DenseNet201	0.7075072949586665
ResNet50	0.7111048660585256

V. Discussion

The discussion based on the results, learning curves, and BLEU scores is discussed below.

A. Comparison of Densenet201 and Resnet50 in Feature Extraction

The results presented in the paper indicate that both Densenet201 and Resnet50 models perform well in feature extraction for image captioning tasks. However, the Resnet50 model has a slightly higher BLEU score (0.7111048660585256) compared to the Densenet201 model (0.7075072949586665), suggesting that Resnet50 may be more effective in capturing the visual features of images relevant to caption generation.

B. Implications of Learning Curves on Model Training

The learning curves presented in the article provide valuable insight into the model training process. The curves show that the training loss decreases over time, while the validation loss initially decreases and then plateaus, indicating that the model no longer improves the validation data. This indicates that early stopping may be useful to avoid technical overfitting and improve model generalization.

C. Discussion on the Relevance of BLEU Scores in Evaluating Caption Quality

The results presented in the paper show that both the Densenet201 and Resnet50 models have a high BLEU score, which indicates that the generated captions are of high quality and accurately reflect the visual content of the images.

The results show that both the Densenet201 and Resnet50 models are effective in feature extraction for captions, and the BLEU score of Resnet50 is slightly higher. The learning curves provide valuable insight into the model's training process and highlight the importance of stopping early to avoid overfitting. A high BLEU score indicates that the generated subtitles are of high quality.

VI. Conclusion

DenseNet201 and ResNet50 are two popular deep-learning architectures used for image classification tasks. Both models have their advantages and disadvantages.

A. Advantages of DenseNet201

1) Strong gradient flow: DenseNet201 has a strong gradient flow, which allows the error signal to easily pass to the previous layers, improving the model performance.

2) Implicit depth tracking: DenseNet201 has implicit depth tracking to help model and improve model consistency.
3) Fewer parameters and higher accuracy: DenseNet201 has fewer parameters compared to ResNet50 and pre-activation ResNet and achieves higher accuracy.
4) Feature reuse: DenseNet201 reuses features more efficiently, reducing the risk of overfitting and improving model generalization.
5) High computing and memory efficiency: DenseNet201 is thinner and more compact, which improves computing power and memory efficiency.

B. Disadvantages of DenseNet201

1) Complexity: DenseNet201 is more complex than ResNet50, which can make it harder to implement and optimize.
2) Computationally expensive: DenseNet201 is computationally expensive due to aggregation, which can require more computing resources.

C. Advantages of ResNet50

1) Residual connections: ResNet50 uses residual connections, which help promote gradient propagation and avoid the vanishing gradient problem.
2) High accuracy: ResNet50 achieves high accuracy in image classification tasks.
3) Pre-activation: ResNet50 uses pre-activation to help improve model uniformity.

D. Disadvantages of ResNet50

1) Overfitting: If there is insufficient training data, ResNet50 may suffer from overfitting.
2) More parameters: ResNet50 has more parameters compared to DenseNet201, which can result in longer training and higher memory requirements.

E. Summary of Key Findings

The key findings from the comparison of DenseNet201 and ResNet50 models in feature extraction for image captioning reveal that both models perform well, with ResNet50 achieving a slightly higher BLEU score of 0.7111 compared to DenseNet201's score of 0.7075. This indicates that ResNet50 may more effectively capture visual features relevant to generating accurate and descriptive captions for images.

F. Suggestions for Future Research and Improvements

To further advance image captioning, future research could focus on hybrid models that combine the strengths of different feature extraction architectures to improve caption quality. In addition to BLEU scores, exploring new evaluation metrics such as semantic similarity measures or human evaluation studies can provide a more comprehensive evaluation of subtitle quality. In addition, the research could be aimed at optimizing training strategies, data augmentation techniques, and model architectures to improve the reliability and generalizability of image caption generators.

CONFLICT OF INTEREST

The author declares no conflict of interest.

ACKNOWLEDGMENT

I would like to acknowledge the organization for providing me the opportunity to conceptualize, ideate, and perform this research and reach an outcome for betterment and advancement.

REFERENCES

[1] Sharma, M. Agrahari, S. K. Singh, M. Firoj and R. K. Mishra, "Image captioning: A comprehensive survey," presented at 2020 International Conference on Power Electronics & IoT Applications in Renewable Energy, 2020.
[2] G. Sharma, P. Kalena, N. Malde, A. Nair, and S. Parkar, "Visual image caption generator using deep learning," presented at 2nd International Conference on Advances in Science & Technology (ICAST), 2019.
[3] R. Calvin and S. Suresh, "Image Captioning using convolutional neural networks and recurrent neural network," presented at 2021 6th International Conference for Convergence in Technology (I2CT), 2021.
[4] Y. Bengio *et al.*, *Show, Attend, and Tell: Neural Image Caption Generation with Visual Attention*, 2015.
[5] G. Sairam, M. Mandha, P. Prashanth, and P. Swetha, "Image captioning using CNN and LSTM," in *Proc. 4th Smart Cities Symposium (SCS 2021)*, Bahrain, 2021, pp. 274–277. doi: 10.1049/icp.2022.0356
[6] N. Indumathi, R. J. Divyalakshmi, J. Stalin, V. Ramachandran, and P. Rajaram, "Apply deep learning-based CNN and LSTM for visual image caption generator," in *Proc. 2023 3rd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, Greater Noida, India, 2023, pp. 1586–1591. doi: 10.1109/ICACITE57410.2023.10183097.
[7] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.
[8] J. Deng, W. Dong *et al*., "ImageNet: A large-scale hierarchical image database," in *Proc. 2009 IEEE Conference on Computer Vision and Pattern Recognition*, Piscataway: IEEE, 2009, pp. 248–55.
[9] X. Yin and V. Ordonez, "Obj2text: Generating visually descriptive language from object layouts," arXiv preprint arXiv:1707.07102, 2017.
[10] Guoyun Tu, Ying Liu, Vladimir Vlassov. "AIC-AB NET: A Neural Network for Image Captioning with Spatial Attention and Text Attributes"
[11] N. Li and Z. Chen, "Image captioning with visual-semantic LSTM," in *Proc. the 2018 Twenty-Seventh International Joint Conference on Artificial Intelligence*, IJCAI, Stockholm, Sweden, July 13–19, 2018, pp. 793, 799.
[12] F. Huang *et al*., "Image captioning with internal and external knowledge," in *Proc. the CIKM '20: The 29th ACM International Conference on Information and Knowledge Management*, Virtual Event, Ireland, October 19–23, 2020, pp. 535–544.