Image Captioning Model Development

This document details the development of an image captioning model designed to automatically generate descriptive captions for images. The model integrates computer vision and natural language processing techniques, developed as part of a Natural Language Processing course requirement, submitted on May 3, 2025. The project covers dataset preparation, feature extraction, model training, and caption generation using advanced methods such as beam search.

HM by Hossam Mohamed

Dataset Overview and Preparation

The project utilizes the Flickr8k dataset, which contains 8,091 images, each paired with five distinct captions. Due to computational constraints, a subset of 1,000 images was selected for training and evaluation. Images are stored in the data/Images/ directory, and captions are provided in a CSV file named data/captions.txt, containing two columns: image filename and caption text.

Captions were preprocessed by converting all text to lowercase, trimming whitespace, and adding special tokens **starttoken** and **endtoken** to mark the beginning and end of each caption. The dataset was reduced to include only captions corresponding to the selected 1,000 images, ensuring focused training data.

Tokenization was performed using Keras' Tokenizer, initially fitted on dummy text to initialize the vocabulary, then on the reduced captions. Special handling was applied to the start and end tokens to adjust their indices for model compatibility. Vocabulary size and maximum caption length were computed to guide model input dimensions.

Feature Extraction Using ResNet18

Image features were extracted using the ResNet18 convolutional neural network pretrained on the ImageNet dataset. The model was modified by removing the final classification layer, allowing extraction of 512-dimensional feature vectors representing each image's visual content.

Images were preprocessed with resizing to 224x224 pixels, normalization using ImageNet mean and standard deviation, and conversion to tensors. The feature extraction function loads each image, applies the transformations, and passes it through ResNet18 in evaluation mode to obtain feature vectors.

Features for all images in the reduced dataset were extracted and stored in a dictionary mapping image filenames to their corresponding feature vectors. This step is crucial for providing the model with meaningful visual representations to condition caption generation.

Training Data Preparation

Training data was prepared by pairing image features with tokenized caption sequences. For each caption, sequences were created to predict the next word given all previous words and the image features. This involved iterating through each caption's token sequence and generating input-output pairs where the input includes the image features and partial caption sequence, and the output is the next word in the sequence.

Sequences were padded to the maximum caption length to ensure uniform input size. The output words were one-hot encoded to represent the vocabulary classes for categorical cross-entropy loss during training. The resulting arrays for image features, input sequences, and output words were shaped appropriately for model training.

This data preparation enables the model to learn the relationship between image content and language sequences, facilitating accurate caption prediction.

Model Architecture and Training

The model architecture combines image features and caption sequences using a dual-input neural network. Image features are passed through a dense layer with ReLU activation and repeated to match the caption sequence length. Caption sequences are embedded into a dense vector space and processed by an LSTM layer with dropout for regularization.

The outputs of the image and caption processing branches are concatenated and fed into a second LSTM layer, followed by a dense softmax layer that predicts the next word in the caption sequence. The model is compiled with categorical cross-entropy loss and the Adam optimizer.

Training was conducted with a batch size of 32 over up to 50 epochs, using early stopping and model checkpoint callbacks to prevent overfitting and save the best-performing model. Validation split of 20% was used to monitor performance during training.

Caption Generation with Beam Search

Caption generation during inference uses a beam search algorithm to explore multiple possible word sequences and select the most probable caption. The process starts with the start token and iteratively predicts the next word, maintaining a set of candidate sequences ranked by their cumulative probabilities.

Advanced techniques such as temperature scaling and top-k filtering are applied to control randomness and focus on the most likely words, improving caption quality. Additionally, a blacklist of common nonsensical words is used to avoid irrelevant or repetitive terms in generated captions.

The beam width controls the number of candidate sequences considered at each step, balancing computational cost and caption diversity. The algorithm terminates when the end token is generated or a maximum caption length is reached, returning the best sequence as the final caption.

Model Evaluation and Sample Captions

The trained model was evaluated by generating captions for a random sample of five images from the test subset. For each image, features were extracted and passed to the caption generation function. The generated captions were compared to the reference captions from the dataset, with BLEU scores computed to assess similarity.

Sample captions were displayed alongside their corresponding images to qualitatively assess the model's descriptive accuracy and fluency. This evaluation provides insight into the model's ability to generalize and produce meaningful image descriptions.

Conclusions and Future Work

This project successfully developed an image captioning model combining convolutional neural networks and LSTM-based language modeling. The use of ResNet18 for feature extraction and beam search for caption generation contributed to effective performance on the Flickr8k subset.

Future improvements could include expanding the dataset size, experimenting with more advanced architectures like transformers, and refining the beam search parameters. Additionally, incorporating attention mechanisms could enhance the model's ability to focus on relevant image regions during captioning.

Overall, this work demonstrates the feasibility of automated image captioning using deep learning techniques and provides a foundation for further research and application development in this area.