IMAGE CAPTION GENERATOR USING CNN AND LSTM

PROBLEM STATEMENT

The image caption generator using CNN and LSTM aims to automatically generate meaningful textual descriptions for images. CNN extracts visual features, while LSTM generates natural language captions based on these features. This system bridges computer vision and natural language processing, improving accessibility, content organization, and human-computer interaction across various applications.

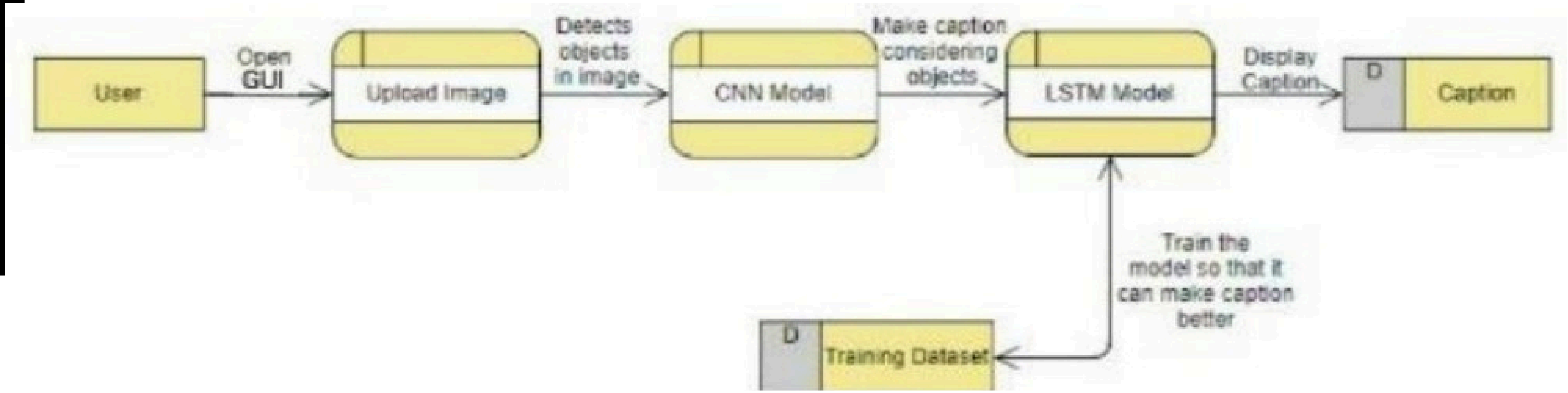
DATASET

Flickr8k

- Description: Contains 8,000 images, each annotated with 5 captions.
- Strengths: Smaller and simpler than COCO, good for initial experiments or prototyping.

MODEL ARCHITECTURE (CNN + LSTM)

Workflow:



PROPOSED METHODOLOGY

1. DataCollection&Preprocessing:

Dataset Selection: Use large-scale annotated datasets like MS COCO, Flickr30k, or Flickr8k with images and corresponding captions.

A. Image Preprocessing :

Resize images to a fixed size (e.g., 224x224 pixels).

Normalize pixel values to the range expected by the CNN model (e.g., mean subtraction based on ImageNet stats).

B. Caption Preprocessing:

- Tokenize captions (split sentences into words/tokens).
- Build a vocabulary of most frequent words.
- Convert captions into sequences of integers using the vocabulary.
- Pad sequences to a fixed length for batch processing.
- Optionally add start and end tokens to indicate caption boundaries.

2. Feature Extraction Using CNN (Encoder)

- Use a pre-trained CNN (e.g., ResNet, InceptionV3, VGG16) as a feature extractor by removing the final classification layer.Extract feature vectors from the last
- convolutional orfullyconnectedlayersforeach inputimage.
- Extract feature vectors from the last convolutional or fully connected layers for each input image.
- These extracted features will serve as visual context vectors for the caption generation.

3. Caption Generation Using LSTM (Decoder)

- The LSTM receives the CNN-extractedimagefeaturesasitsinitialhiddenstateorasinput at the first time step.

- Use an embedding layer to convert word indices to dense vectors for the LSTM input.
- At each subsequent time step, the LSTM takes the previous word (starting with the start token) as input and predicts the next word in the caption sequence.

- The output is a probability distribution over the vocabulary for each time step, from which the next word is selected (via greedy search or beam search during inference).

4. Training the Model

Use teacher forcing during training, feeding the ground truth word at each time step into the LSTM. Define the loss function as the categorical cross-entropy between the predicted word probabilities and the true next word. Optimize using an optimizer like Adam. Regularization techniques such as dropout and early stopping to prevent overfitting.

5. Inference / Caption Generation

Given a new image: Extract features using the CNN encoder. Feed features to the LSTM decoder. Generate captions by predicting the next word until the end token is produced or maximum length is reached. Use beam search or greedy decoding to improve caption quality.

6. Evaluation

Evaluate generated captions against ground truth captions using metrics such as BLEU, METEOR, CIDEr, and SPICE. Perform qualitative analysis by inspecting sample captions and comparing them to human-generated captions.

APPLICATIONS

- Assistivetechnology for visuallyimpaired
- Automatic metadata generation
- Enhanced image search engines
- Social media automation
- Content management systems

CONCLUSION

An Image Caption GeneratorusingCNNand LSTM provides a powerful way to bridge vision and language. It has broad applications in accessibility, information retrieval, and AI-powered content creation

THANK YOU

BABITHA C
PALLAVI P PAWALE
TEJASWINI K P