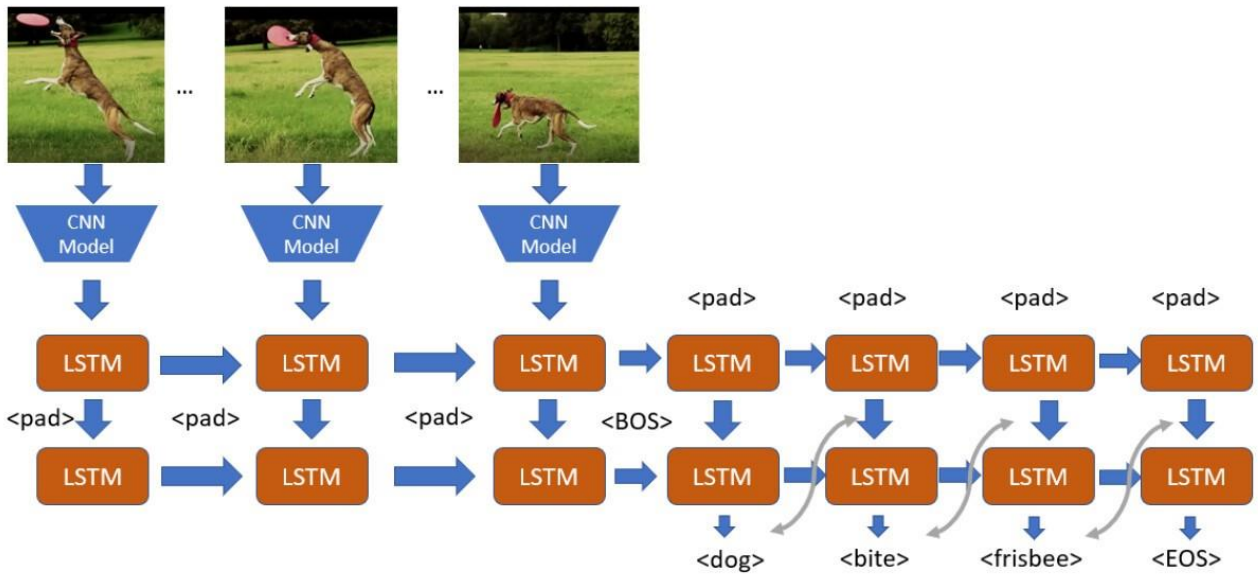
Video to Text with LSTM models

Real-world videos often have complex dynamics, methods for generating open-domain video descriptions should be sensitive to temporal structure and allow both input (sequence of frames) and output (sequence of words) of variable length. To approach this problem we propose a novel end-to-end sequence-to-sequence model to generate captions for videos. For this we exploit recurrent neural networks, specifically LSTMs, which have demonstrated state-of-the-art performance in image caption generation. Our LSTM model is trained on video-sentence pairs and learns to associate a sequence of video frames to a sequence of words in order to generate a description of the event in the video clip. Our model naturally is able to learn the temporal structure of the sequence of frames as well as the sequence model of the generated sentences, i.e. a language model. We evaluate several variants of our model that exploit different visual features on a standard set of YouTube videos and two movie description datasets (M-VAD and MPII-MD).



Problem with Long-Term Dependencies in RNN

Recurrent Neural Networks (RNNs) are designed to handle sequential data by maintaining a hidden state that captures information from previous time steps. However they often face

challenges in learning long-term dependencies where information from distant time steps becomes crucial for making accurate predictions for current state. This problem is known as the vanishing gradient or exploding gradient problem.

- Vanishing Gradient: When training a model over time, the gradients (which help the model learn) can shrink as they pass through many steps. This makes it hard for the model to learn long-term patterns since earlier information becomes almost irrelevant.
- Exploding Gradient: Sometimes, gradients can grow too large, causing instability. This makes it difficult for the model to learn properly, as the updates to the model become erratic and unpredictable.

Both of these issues make it challenging for standard RNNs to effectively capture long-term dependencies in sequential data.

LSTM Architecture

LSTM architectures involves the memory cell which is controlled by three gates: the input gate, the forget gate and the output gate. These gates decide what information to add to, remove from and output from the memory cell.

- Input gate: Controls what information is added to the memory cell.
- Forget gate: Determines what information is removed from the memory cell.
- Output gate: Controls what information is output from the memory cell.

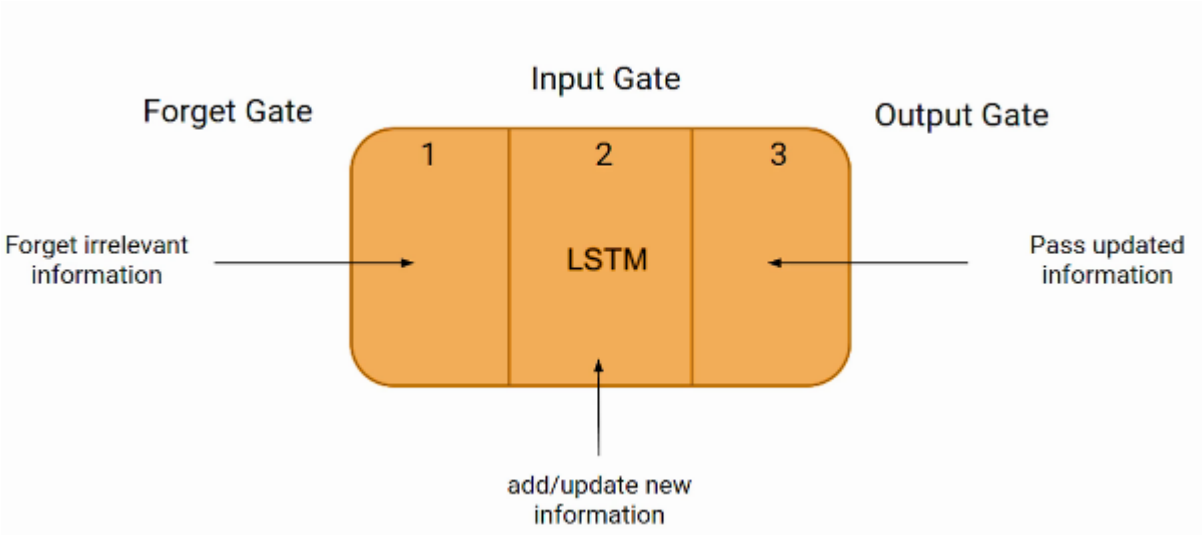
This allows LSTM networks to selectively retain or discard information as it flows through the network which allows them to learn long-term dependencies. The network has a hidden state which is like its short-term memory. This memory is updated using the current input, the previous hidden state and the current state of the memory cell.

Working of LSTM

LSTM architecture has a chain structure that contains four neural networks and different memory blocks called cells.

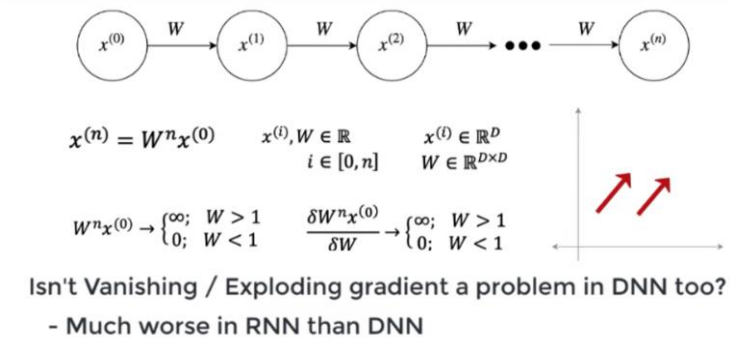
What makes LSTM suitable for Video Captioning:

- Sequential Data Handling:LSTMs designed for sequential data, suitable for video frames.
- Temporal Relationships:Captures temporal dependencies crucial for video understanding.
- Memory Cells:LSTMs use memory cells for storing and retrieving information over sequences.
- Vanishing Gradient Problem:Mitigates vanishing gradient problem for effective training.
- Gating Mechanisms:Input, forget, and output gates control information flow.



LSTM Model

- Variable Sequence Lengths:Handles videos with varying lengths, providing flexibility.



LSTM for Video Captioning

- Feature Extraction:Combines with convolutional layers for spatial feature extraction.

Working:

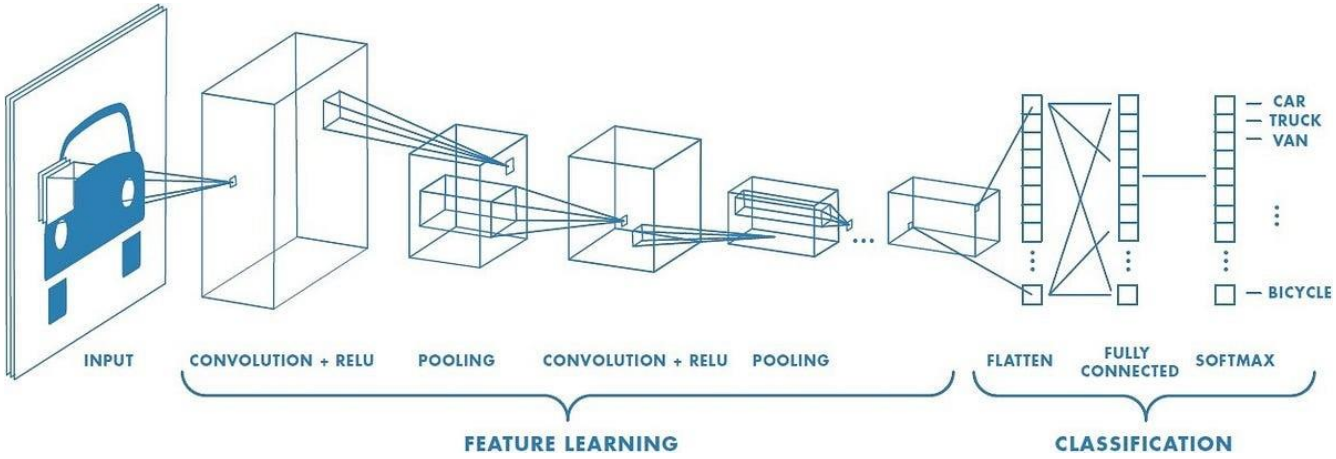
1) Preprocessing:

Remember that a Video is made of Severl number of frames. This step involves extracting frames from the video and converting them into a format suitable for input into the LSTM network. Additionally, textual data for training and evaluation is prepared.

Selecting frames from Video

2) Feature Extraction:

Video frames are often transformed into feature vectors using techniques like convolutional neural networks (CNNs). Convolutional neural networks are distinguished from other neural networks by their superior performance with image, speech, or audio signal inputs. These features capture visual information in a more abstract form, which is then fed into the LSTM network.


CAR
TRUCK
VAN
BICYCLE
INPUT
CONVOLUTION + RELU
POOLING
CONVOLUTION + RELU
POOLING
FLATTEN
FULLY CONNECTED
SOFTMAX
FEATURE LEARNING
CLASSIFICATION