Video Captioning Bot

A PROJECT REPORT

submitted in fulfilment of requirements for the award of

the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

SUBMITTED BY

KARTIK KARIRA (18103051)

ROHIT MITTAL (18103081)

under the supervision of

DR. AVTAR SINGH

ASSISTANT PROFESSOR

CSE DEPARTMENT

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING DR. B.R. AMBEDKAR NATIONAL INSTITUTE OF TECHNOLOGY, JALANDHAR-144011, PUNJAB (INDIA)

MAY 2022

ACKNOWLEDGEMENT

It is true that hundreds of people work behind the scenes to make a Play a success. We'd want to thank everyone who helped us finish our final project-**Video Captioning Bot** with our sincere gratitude. We ran into a several problems during the project due to a lack of information and competence, but these people helped us overcome these difficulties and develop our idea for shaping sculpture.

We'd like to thank thankful to Professor A.L Sangal, Head, Department of Computer Science & Engineering, for giving his leadership, guidance and all his direct and indirect support, which allowed the entire team to grasp every facet of the project.

We are thankful to the In-charge, Major Project Final Year, for providing us mentor and all other support.

We'd also want to thank Dr Avtar Singh, Assistant Professor, our mentor, who believed in our idea and offered fresh suggestions as needed and also provided his continuous support and monitoring throughout the project.

We are extremely thankful to have got constant encouragement and guidance from all the Faculties of Department of Computer Science & Engineering (CSE) who gave us their time and suggestions and thence helped us in successfully completing our project work. Also, we would like to extend our sincere esteems to all staff in laboratory for their timely support.

Thank you

ABSTRACT

Video understanding has become increasingly important as surveillance, social, and informational videos weave themselves into our everyday lives. Video captioning offers a simple way to summarize, index, and search the data. Video Captioning is a task of automatic captioning a video by understanding the action and event in the video which can help in the retrieval of the video efficiently through text. In our work we introduce a model which first extract features from the videos using Convolutional Neural Network and then using Long Short-term memory (LSTM) try to provide suitable caption to the video clips.

DECLARATION

*We herewith certify that the work that is being conferred within the project report entitled "**Video Captioning Bot**" in partial fulfilment of necessities for the award of degree of B.Tech. (Computer Science and Engineering.) submitted to the **Department of Computer Science and Engineering** of **Dr B R Ambedkar National Institute of Technology, Jalandhar**, is an authentic record of our own work carried out during a period from July, 2021 to May, 2022 under the supervision of **Dr Avtar Singh, Assistant Professor**. The matter presented in this dissertation has not been submitted by me in any other University/Institute for the award of any degree.*

Kartik Karira *(18103051)*

Rohit Mittal *(18103081)*

This is to certify that the above statement made by the candidates is correct and true to the best of my knowledge

Signature of Supervisor
Dr. Avtar Singh
Assistant Professor
(Dept. of CSE)
NIT Jalandhar

Thank you all.
Date: 14th May, 2022

Table Of Contents

1.INTRODUCTION

1.1. Background of Problem

The task of video captioning has become very popular in recent years. With all these platforms like YouTube, Twitch and short video like Instagram Reels, videos have become a very important means of communication in our daily life. According to Forbes, over 500 million people watch video on Facebook every day. 72 hours of videos are uploaded to YouTube every minute. With videos gaining such high popularity AI products for videos have become an all time necessity.

1.2. Computer Vision

Computer visualization is a machine learning environment dedicated to the interpretation and comprehension of images and videos. It is used to teach computers to “see” and to use visual information to perform tangible tasks that people can do.

Computer vision models are designed to translate visual data based on features and knowledge of a situation identified during training. This enables models to interpret images and videos and to use them in interpreting and decision-making tasks.

Although both are related to visual data, image processing is not the same as computer view. Image processing involves adjusting or enhancing images to produce a new effect. It may include improving light or brightness, enhancing correction, blurring sensitive information, or blurring. The difference between image processing and computer recognition is that the original does not require the identification of content.

1.3. Machine Learning

Machine learning is important because it gives businesses an idea of customer behavior trends and business performance patterns, as well as supporting the development of new products. Many of today's leading companies, such as Facebook, Google and Uber, make machine learning an integral part of their operations. Machine learning has become an important competitive advantage in many companies.

1.4. Artificial Neural Network

Computer Visualization is a program for viewing images and videos available in digital formats. In Machine Learning (ML) and AI - Computer vision is used to train a model to identify specific patterns and store data in its artificial memory so that it can use the same to predict results in real-world applications.

The ultimate goal is to use computer visualization technology in ML and AI to create a model that can work on its own without human intervention. The whole process involves methods of data acquisition, processing, analysis, and understanding of digital images to apply the same in a real-world context.

1.5. Deep Learning

In-depth learning (also known as in-depth formal learning) is part of a wider family of mechanical learning methods based on neural networks with learning representation. Learning can be supervised, partially monitored or supervised.

In-depth learning structures such as deep neural networks, deep belief networks, deep reinforcement learning, general neural networks and convolutional neural networks used in fields including computer vision, speech recognition, natural language processing, machine translation, bionomics of drugs, treatment. photographic analysis, climate science, material testing and board game systems, where it produces comparable results and in some cases surpasses the performance of human professionals.

Artificial neural networks (ANNs) are developed to process information and communication nodes distributed in biological systems. ANNs vary from brain to biological. In particular, artificial sensory networks tend to stand out and symbolize, while the biological brains of many organisms are flexible (plastic) and analogue.

1.6. Convolutional Neural Network

Convolutional neural networks are a special type of artificial intelligence networks that use mathematical operations called convolution instead of repeating a normal matrix in at least one of them. They are specifically designed to process pixel data and are used for image recognition and processing.

VGG16 is a structure of the convolution neural net (CNN). It is considered to be one of the best vision model structures to date. The great thing about VGG16 is that instead of having a large hyper parameter they focus on having conviction layers of 3x3 filter with stride 1 and always use the same padding and maxpool layer of 2x2 stride filter 2. It follows this stride program 2. convolution and layout of large lakes consistently across the architecture. Finally it has 2 FC (fully connected layers) followed by softmax output. 16 in VGG16 refers to it has 16 layers of weight. This network is the largest network and has about 138 million frames (approx).

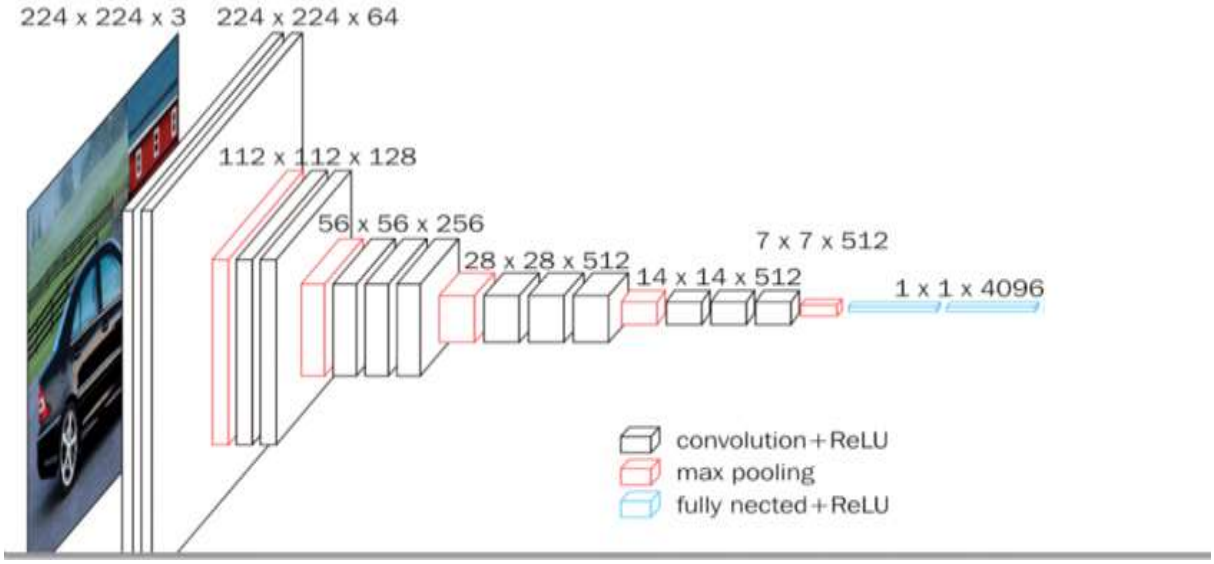


Fig: Diagram showing working of CNN

1.7. Recurrent Neural Network

Recurrent neural network (RNN) is a class of sensory networks that are formed when connections between nodes form a direct or indirect graph. This allows it to display temporary dynamic behavior. Based on feedforward neural networks, RNNs can use their internal memory (memory) to process a sequence of different lengths of inputs. This enables them to work on tasks such as uninterrupted, handwritten attention or speech recognition. Common neural networks in theory are perfect and can use malicious programs to process random input sequences.

The term “repetitive neural network” is used to refer to the category of networks with a continuous dynamic response, whereas the “convolutional neural network” refers to the stage of limited accessible response. Both categories of networks exhibit temporary dynamic behavior. A limited recurring network is a direct acyclic graph that can be opened and replaced by a solid neural feedforward network, while an infinite repetitive network is a graph of the directional cycle that can disassemble.

Both limited duplicate networks and non-permanent networks may have additional stored conditions, and storage can be directly controlled by a neural network. Storage can be replaced with another network or graph if that includes time delays or feedback loopholes. Such controlled situations are called gate state states or gate memory, and are part of the short-term memory networks (LSTMs) and duplicate units with gates. This is also called the Feedback Neural Network (FNN).

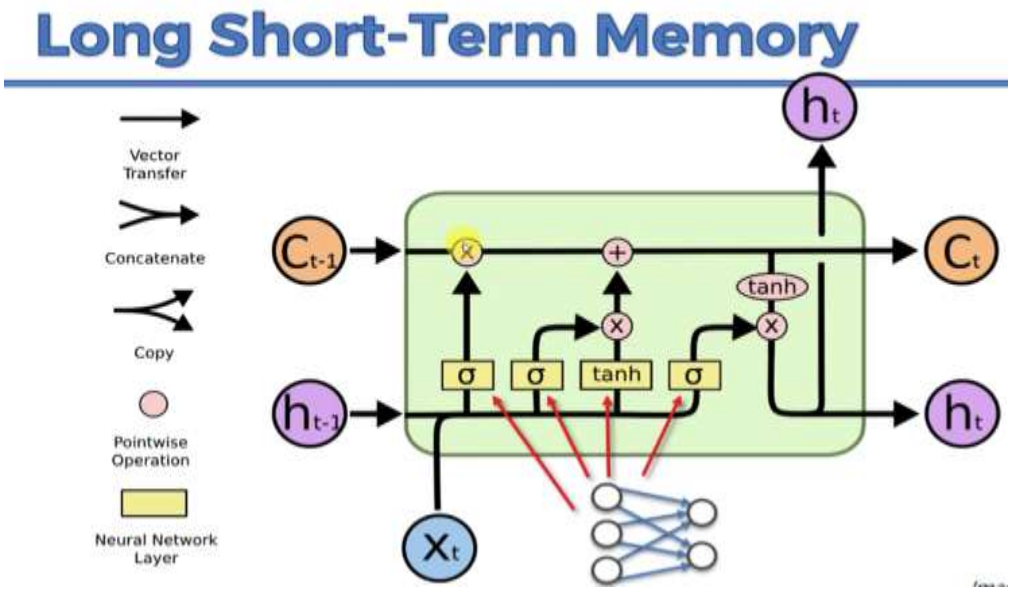
LSTM

Long Short-term memory (LSTM) networks were established by Hochreiter and Schmidhuber in 1997 and set accuracy records for multiple application domains.

About 2007, LSTM began transforming speech recognition, making traditional models work well in other speech applications. In 2009, the LSTM network trained by Connectionist Temporal Classification (CTC) became the first RNN to win pattern recognition competitions when it won several competitions in integrated handwriting recognition. In 2014, the Chinese company Baidu used CTC-trained RNNs to break the 2S09 Switchboard Hub5'00 speech recognition data set benchmark without using any common speech processing methods.

LSTM also improved vocabulary speech recognition and text-to-speech integration and was implemented in Google Android. In 2015, Google speech recognition reported a shocking 49% performance over CTC-trained LSTM.

LSTM breaks records for advanced machine translation, Language Model and Multilingual Processing. LSTM integrated with convolutional neural networks



1.8. Data Representation

For the purpose of this study, we are using the MSVD dataset created by Microsoft.
This dataset has about 1550 videos in which 1450 videos are used for training and validation and remaining 100 for the testing purpose.One example can be seen as

"caption": [
"A boy is playing a key-board between the people.",
"A boy is playing a piano in front of a crowd.",
"A boy is playing a piano.",
"A boy plays a piano for a group of kids.",
"A boy plays the piano.",
"A kid is playing a piano.",
"A young boy is playing a piano in front of a crowd of other young people.",
"A young boy is playing the piano before an audience.",
"A young boy is playing the piano.",
"A young boy seated on stage is playing a piano as the audience watches him.",
"The boy is playing the piano.",
"The boy performed on the piano for an audience.",
"The boy performed on the piano for the audience."
]

2. LITERATURE SURVEY

2.1. Literature Survey

What do you see in the below video?

Can you write a caption?

Well some of you might say "**A white dog in a grassy area**", some may say "**White dog with brown spots**" and yet some others might say "**A dog on grass and some pink flowers**".

Definitely all of these captions are relevant for this image and there may be some others also. But the point I want to make is; it's so easy for us, as human beings, to just have a glance at a picture and describe it in an appropriate language. Even a 5-year-old could do this with utmost ease. But, can you write a computer program that takes an image as input and produces a relevant caption as output?

If, we can produce a relevant caption for image, we could also comprehend what is being happening in a small video, by dividing the video in number of frames. This problem was well researched by **Andrej Karapathy** in his PhD thesis at Stanford, who is also now the **Director of AI at Tesla.**

3. PROBLEM STATEMENT AND FEASABILITY

3.1. Problem Statement and its Necessity

We must first understand how important this problem is to real world scenarios. Let's see few applications where a solution to this problem can be very useful.

1. **Aid to the blind** — We can create a product for the blind which will guide them travelling on the roads without the support of anyone else. We can do this by first converting the scene into text and then the text to voice. Both are now famous applications of Deep Learning. Refer this link where it shown how Nvidia research is trying to create such a product.
2. **Self-driving cars** — Automatic driving is one of the biggest challenges and if we can properly caption the scene around the car, it can give a boost to the self-driving system.
3. **Better search algorithms** : If each video can be automatically described search algorithms will have finer more accurate results.
4. Automatic Captioning can help, make Video Search as good as Google Search, as then every video could be first converted into a caption and then search can be performed based on the caption.
5. **Recommendation Systems**: We could easily be able to cluster videos based on their similarity if the contents of the video can be automatically described.

3.2. Feasibility- Technical , Non-Technical

Before starting a project, its crucial to have a know-how of it's feasibility. The Various Kinds of Feasibilities can be summed up as follows:-

- **TECHNICAL: -**
  - System with high computing and processing power.
  - Camera with good quality precision.

- Internet connectivity is required for the system.

- **NON- TECHNICAL: -**
  - This project doesn’t require much cost in development.
  - Scope of the project is everywhere in the upcoming digital world

4.METHODOLOGY

4.1. DATASET DESCRIPTION

For the purpose of this study, we are using the MSVD(Microsoft Research Video Description Corpus) and MSR-VTT(Microsoft Research Video to Text) dataset created by Microsoft.

MSVD:

This dataset has about 1550 videos in which 1450 videos are used for training and validation and remaining 100 for the testing purpose.

MSR-VTT:

MSR-VTT is a large-scale dataset for the open domain video captioning, which consists of 10,000 video clips from 20 categories, and each video clip is annotated with 20 English sentences.

4.2. DETAILED SOLUTION

Video captions are a text description of video content production. Compared to picture captions, the location is much more flexible and contains more information than a still image. Therefore, in order to make a description of the text, video captions need to extract a lot of features, which is much harder than image captions. The most common methods of video caption work are made up of two parts, part of the video element and part of the production of the video description. The standard format of video captions is shown in Figure below.

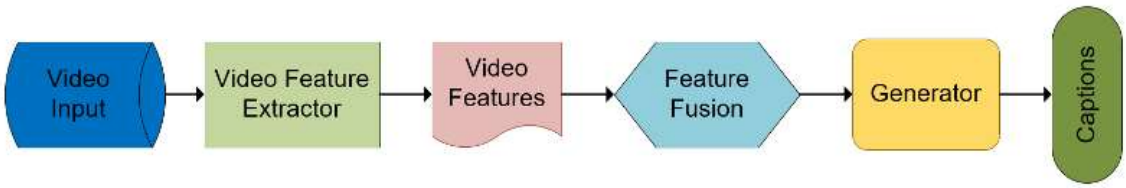


Fig: Flow chart for Model

Video Captioning has two major Parts:

1.Extracting features from video using CNN

2.Generating valid caption for a video frame or image using RNN.

4.3. Model Architecture

Introduction: In the video caption model, the generated statements should have the following characteristics. First is the authenticity. The statements made should really reflect the content of the video. The second is nature. The sentences formed should be close to the sentences expressed by the people in the same situation and in accordance with the rules of grammar. Third is the diversity. The statements made must be very different, and different statements can be used to describe the content of the same video. In order for our video caption model to have all three features, a redesigned video description network is proposed based on multimodal feature integration. The network structure is shown in Figure below.

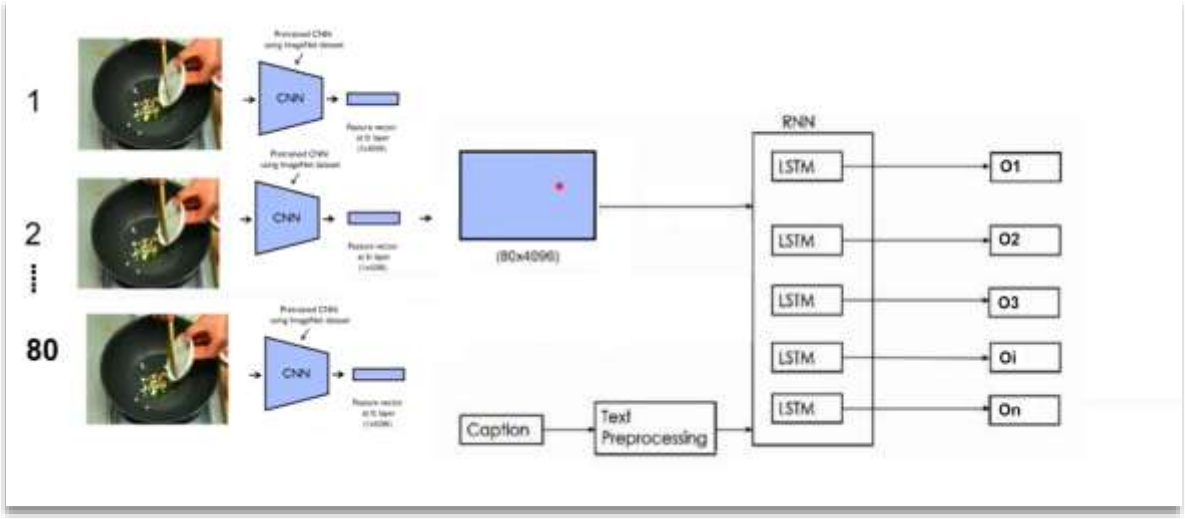


Fig: Model Working Example

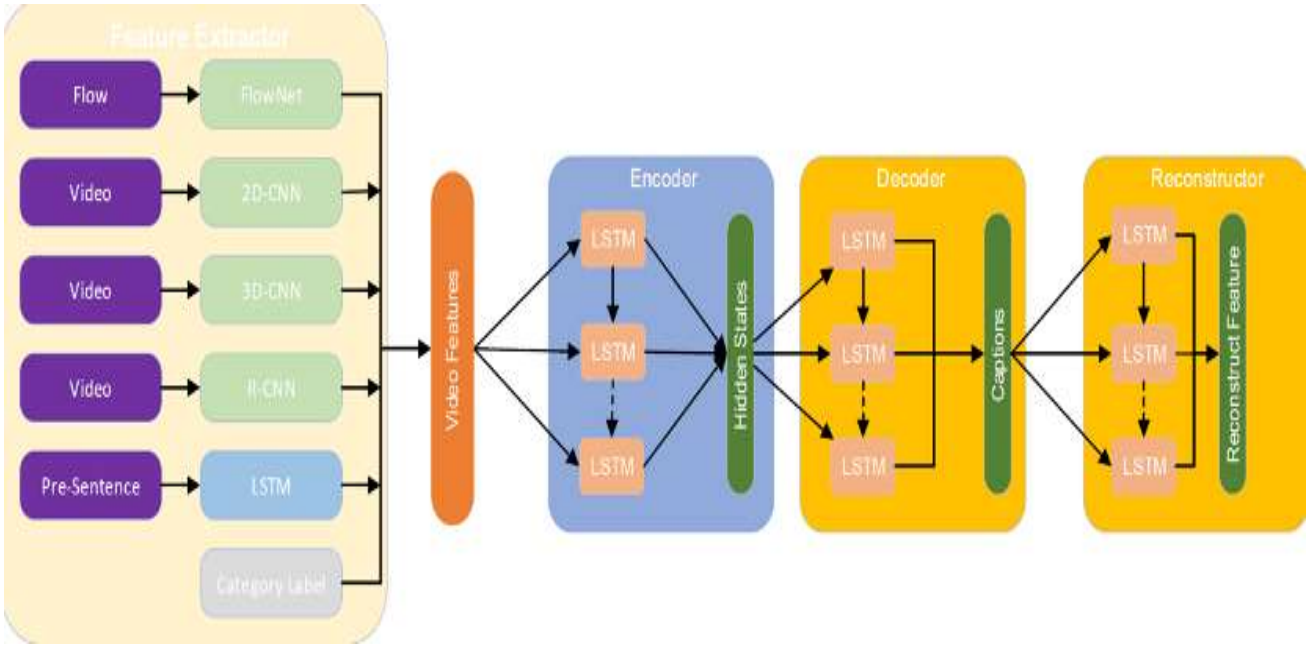


Fig: Model Flow

Architecture: Especially for problems related to text production, the preferred model is encoder-decoder architecture. Here in our statement of problem as the text should be done, we will use this sequence-to-sequence structure.

One thing you should know about this structure is the last state of the encoder cell always acts as the original state of the encoder cell. In our case we will use the encoder to insert video features and the video will be provided with captions.

Now that we have found out we are going to use a decoder model let's take a look at how to use it. What does the video say again? We can call it a sequence of image right? For anything related to sequence we choose to use RNNs or LSTM

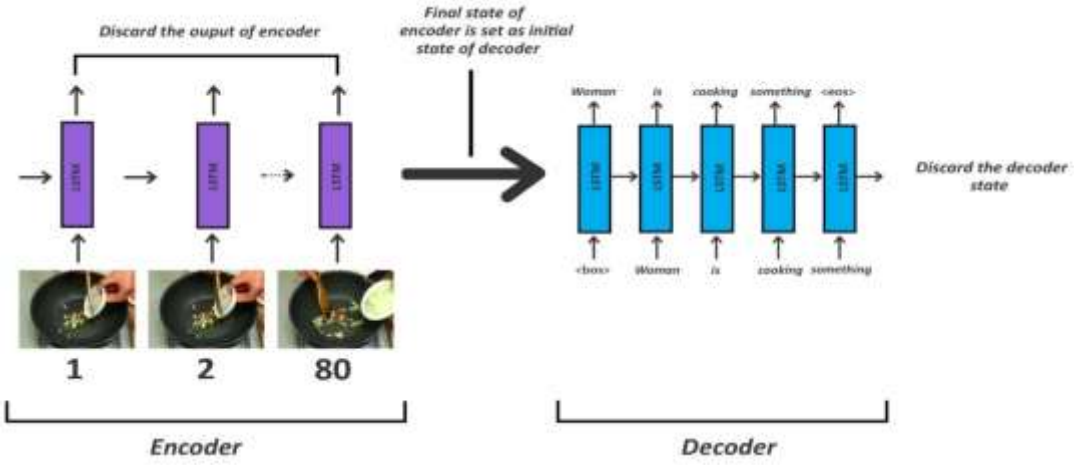


Fig: Decoder and encoder

Now that we're going to use LSTM encoder let's look at the decoder. The decoder will generate captions. Captions are actually word order so we will use LSTM to scan it again.

Here in the picture the elements of the first frame are inserted into the LSTM cell 1 of the encoder. This is followed by features of the second framework and this continues until the 80th framework. In this case we are only interested in the last instance of the encoder so that all other effects from the encoder are discarded. Now the final state of the LSM encoder acts as the first state of the LSTM decoder. Here in the first decoder LSTM <bos> serves as the starting point for a sentence. Each caption from the training data is fed one by one until <eos>.

So, in the example above, when the real caption says a woman is cooking something the decoder starts with <bos> in the first LSTM decoder. In the next cell the next word comes from the real meaning that a woman is fed and then cooked. This ends with a <eos> token.

Codec timer steps for the number of LSTM cells that we will use for the 80-bit encoder. Coding tokens number of features from 4096 video to us. Steps time decoder number of LSTM 10 decoder cells and token number of 12594 vocabulary length.

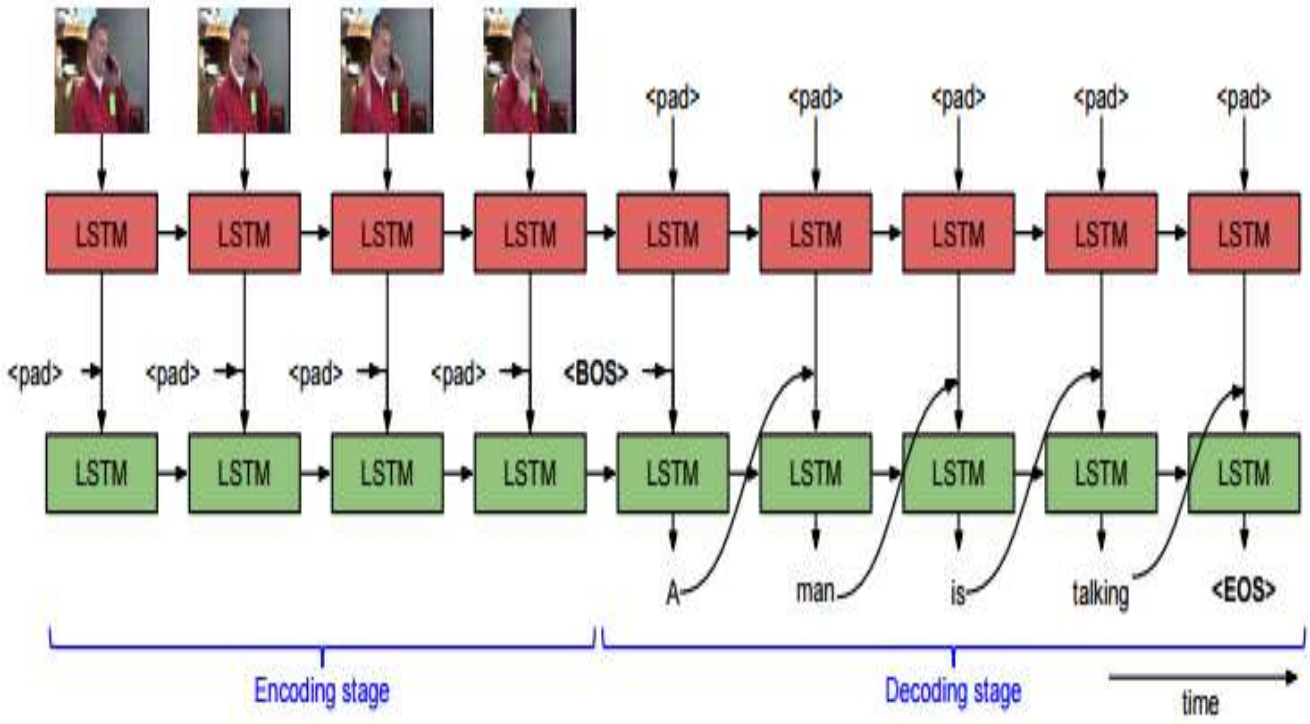


Fig: Encoding Decoding LSTM

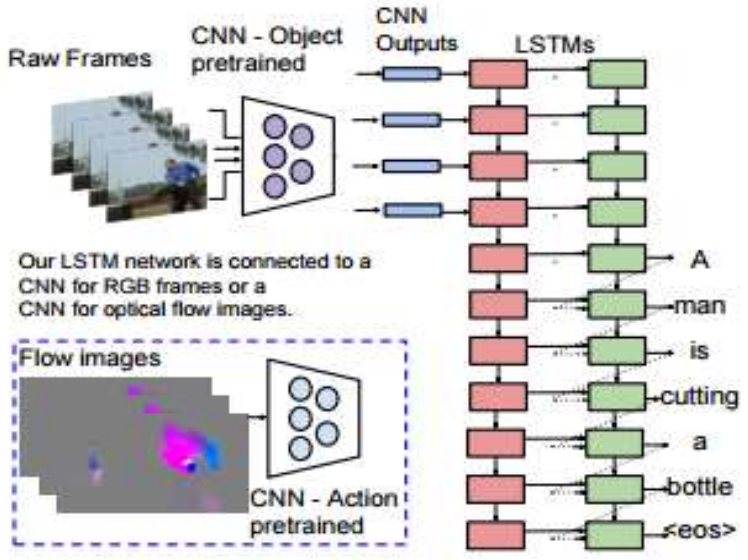


Fig: Dividing Video in frames, those are fed in CNN for feature extraction

4.4 Tech Stack Analysis

In order to achieve various solutions, we have used a variety of Tech Stacks. All these technologies have been chosen on the basis of the following few criteria: 1. Ease of Usage and Ease of Learning

2. Time Required to build
3. Efficiency
4. Security

The various technologies used are:

1. **Python**

   Python is a programming language that supports multiple application builds. Engineers consider it a good choice for Artificial Intelligence (AI), Machine Learning, and In-Depth Learning projects.

   Large number of libraries and frameworks: Python Language comes with many libraries and frameworks that make coding easier. This also saves valuable time.

   The popular NumPy libraries, used for scientific statistics; SciPy to get the most advanced statistics; and scikit, for data mining and data analysis.

   These libraries work closely with powerful frameworks such as TensorFlow, CNTK, and Apache Spark. These libraries and frameworks are important when it comes to machine learning and in-depth learning projects.

   Python code is short and readable even for new developers, which benefits machine and in-depth learning projects. Due to its simple syntax, the development of Python applications is faster compared to most programming languages. In addition, it allows the engineer to test algorithms without using them.

2. **Tkinter (GUI)**

   Tkinter is a standard Python GUI library. Python when combined with Tkinter provides a quick and easy way to build GUI applications. Tkinter provides a powerful visual-based interface to the Tk GUI tool kit.

3. Keras

Keras is an in-depth learning API written in Python, running on the TensorFlow machine learning platform. Built with a focus on compliance and rapid testing. Being able to move from perspective to results as quickly as possible is the key to doing good research.

Keras says:

1.It's easy - but not easy. Keras lowers the engineer's mental load to free you from focusing on the problem areas that are really important.

2.Flexible - Keras adopts the principle of continuous complexity disclosure: easy workflow should be fast and easy, while improperly improved workflow should occur in a clear way that builds on what you have already learned.

3.Powerful - Keras provides industry firm performance and measurement: used by organizations and companies including NASA, YouTube, or Waymo.

4. Tensorflow

TensorFlow is a framework created by Google for creating Deep Learning models. Deep Learning is a category of machine learning models (= algorithms) that use sensory networks with multiple layers.

Machine Learning has enabled us to build complex applications with great accuracy. Whether related to photography, videos, text or audio, Machine Learning can solve problems from a wide range. Tensorflow can be used to execute all these applications.

The reason for its popularity is that developers can easily create and use applications. The GitHub projects we will be looking at closely because the following sections are very powerful but also very easy to work with. In addition, Tensorflow is created with a limited processing capacity in mind. The library can be used on all types of computers, even smartphones (yes, even for something with more than half an apple on it). I can assure you, you are working on an Intel Core i3 with 8 GB of RAM, you will not have any performance problems.

5. OpenCV2

We all know OpenCV as one of the leading computer libraries out there. Additionally, it also has the functionality of using in-depth reading inference. The best part is to support the loading of different models from different frameworks using which we can perform several in-depth learning activities. The feature of supporting models from different frames has become part of OpenCV since version 3.3. However, many newcomers are unaware of this excellent OpenCV feature.

6. Numpy

NumPy or Numerical Python is an open source Python library that makes it easy to perform numerical operations. Working with machine learning and in-depth learning applications involves complex mathematical operations with large databases. NumPy makes using these functions easier and more efficient compared to its pure Python application.

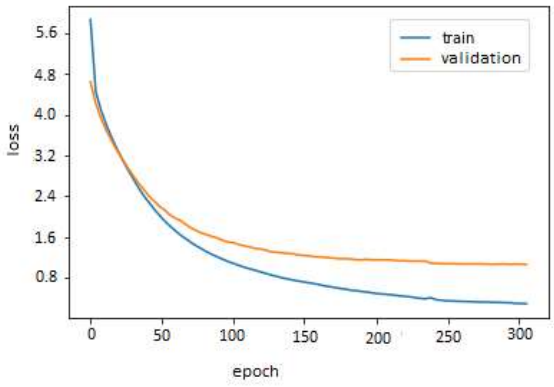
At its core, NumPy uses its own data structure (n-dimensional array), similar to the standard Python array. Many programming languages have the concept of fines only. Python uses lists, which act as lists, but there are differences.In deep learning, you encounter situations where you need the Ziro & Ones matrix. NumPy has useful uteros () and () functions that you can use to generate a matrix of 0 or 1s.

5.Result and Discussion

5.1 Performance of our models:

We trained and tested our machine learning models on a dataset of 2000 videos each having 10 captions so total our dataset has 20000 labels We used different combinations of various machine learning and feature extraction algorithms.

We tracked the model's loss on the training set, as well as its performance on the validation set, during training. The validation set is solely used to assess the trained model's generalisation ability, not to train it.



Quantitative evaluation of the models are performed using the METEOR and BLEU Score metric.

BLEU Score: BLEU (BiLingual Evaluation Understudy) is a metric for automatically evaluating machine-translated text. The BLEU score is a number between zero and one that measures the similarity of the machine-translated text to a set of high quality reference translations.

METEOR Score: METEOR (Metric for Evaluation of Translation with Explicit ORdering) is a metric for the evaluation of machine translation output. The metric is based on the harmonic mean of unigram precision and recall, with recall weighted higher than precision.

```
Bleu Score is 0.8883465596797255
Meteor Score is 0.9294392357928086
```

Test Metric	Value
BLEU Score	0.88
METEOR Score	0.92

.

5.3 Deployment Status And Testing

To understand how good our bot is, let's try to generate captions on videos from the test dataset (of MSVD2).

Important Point:

We must understand that the videos used for testing must be semantically related to those used for training the model. For example, if we train our model on videos mentioning only men in clips. we must not test it on videos of women in clips, as model is not trained to identify a women. This is an example where the distribution of the train and test sets will be very different and in such cases no Machine Learning model in the world will give good performance.

Caption: .
A MAN IS CUTTING A PAPER .

Caption: .
A Man is cutting an onion

Caption
A man is talking

Caption: .
A baby is eating spaghetti .

Caption:
Men are playing in the beach

Caption:
a man is playing a guitar

Caption
A cat is playing the piano

Caption: .
A GROUP OF PEOPLE ARE DANCING

5.4 Social and Environmental Impact

- One of the most significant advantages of adding subtitles/ captions to your videos is boosting engagement across social media platforms.
- Condensation of huge, long ,lengthy videos into useful information.
- In advent of todays social media world, where videos are biggest form of media , and by , captioning them we can help multi tasking users, to have seamless experience.
- It will open doors to innovations in computer vision which can be implemented in self driving cars, self controllable machinery, in recommendation system, search engines ,and many more.

6.CONCLUSION AND FUTURE SCOPE

Please refer Google Drive link **here** to access the full code. We learnt a lot of new concepts, gained technical expertise in various tech-stacks and applied them in this project. On the non-technical side, we learnt project planning, time management and working in a team remotely.

Future Scope:

A lot of modifications can be made to improve this solution like:

- Using a **larger** dataset for more refined output or caption result.
- Changing the model architecture, e.g. include an **attention** module.
- Doing more **hyper parameter tuning** (learning rate, batch size, number of layers, number of units, dropout rate, batch normalization etc.).
- The response time are heavily rely on the hardware of the machine, i.e. the processing speed of the processor, the size of the available RAM, and the available features of the webcam, its resolution. Therefore, the program may have better performance when it's running on a decent machines.
- This system can be implemented in many applications that can access government websites whereby video captioning in audio for blind is available or filling out forms online whereby no interpreter may be present to help.

----------END OF REPORT----------

References

1. H. Aradhye, G. Toderici, and J. Yagnik. Video2text: Learning to annotate video content. In ICDMW, 2009.
2. X. Chen and C. L. Zitnick. Learning a recurrent visual representation for image caption generation. CVPR, 2015
3. Sequence to Sequence – Video to Text by Subhashini Venugopalan, Marcus Rohrbach,Jeff Donahue, Raymond Mooney, Trevor Darrell,Kate Saenko
4. M. Denkowski and A. Lavie. Meteor universal: Language specific translation evaluation for any target language. In EACL, 2014
5. S. Hochreiter and J. Schmidhuber. Long short-term memory. Neural computation, 9(8), 1997
6. J. Y. Ng, M. J. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. Beyond short snippets: Deep networks for video classification
7. Very Deep Convolutional Networks for Large-Scale Image Recognition by Karen Simonyan, Andrew Zisserman
8. Transfer learning using VGG-16 with Deep Convolutional Neural Network for Classifying Images by Srikanth Tammina
9. BLEU: a Method for Automatic Evaluation of Machine Translation by Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu
10. Image Captioning Using R-CNN & LSTM Deep Learning Model by Aditya Kumar Yadav and Prakash.J