Video Caption Generation In Bangla Based on Attention Model

A thesis

Submitted in partial fulfillment of the requirements for the Degree of
Bachelor of Science in Computer Science and Engineering

Submitted by

Suvom Shaha	**170104050**
Tunazzin Rahman Topu	**170104066**
Almas Shahriar Ador	**170104074**
Saiful Islam	**170104147**

Supervised by

Mr.Faisal Muhammad Shah

Department of Computer Science and Engineering

Ahsanullah University of Science and Technology

Dhaka, Bangladesh

July 2021

ABSTRACT

Captioning can be defined as a manner, which includes dividing massive transcripts into chucks, which is also known as ‘caption frames’. Captioning has continually been vital, but with the converting generation, the upward push of video reputation, and the growth of the deaf and hard of listening to population, it’s become more pertinent than ever earlier than. Artificial intelligence, which is a buzzword of cutting-edge technology, needs a few elements or fashions which could help to apprehend complicated visual frames from a video and describe them in a sentence [3], [5]. It enriches the natural language communication among humans and machines [4]. Good sized steps had been taken by means of scientists and researchers to attain this goal within the previous few years and they can discover the visual context from a video [1], [2], [4], [7]. This form of works makes the computers permit of giving a precis of a video which saves treasured time for people and every so often it extracts such capabilities that a human eye may miss [4]. Within the sector of media, security surveillance, activity tracking [4]. Video captioning can play a critical position in reading the accrued records besides it has some useful impact on society. As it may summarize a video [5], so it may ease their existence by providing instructions from the actual-time scenario. But most of the works has been accomplished within the English language. So for the humans whose mother tongue is Bangla and who have very little understanding of English, they can be positioned to appropriate use of this technology. For them, we have got proposed this model to take away the barrier of language and give them the taste of their mother tongue in the quarter of video captioning.

Contents

List of Figures

Chapter 1

Introduction

1.1 Problem Statement

After seeing this circumstance we have aimed to generate caption in Bengali from a video the use of various machine learning processes. By way of this our society could be benefited and it will upload a star in the area of video captioning. In this approach we must undergo unique stages like making dataset, pre-processing the videos, differentiating spatial and temporal features, word tokenization and so forth.

1.2 Motivation

Bangla is the fifth maximum-spoken (www.dhakatribune.com) native language and the 7th most spoken language by using the full variety of speakers within the international. But no work has been done in the field of video captioning in Bangla. It is one of the motivations of us. Again, it can be used for visually impaired individuals. In the event that they use a camera to report the state of affairs of his outdoor surroundings and feed the video in our model will provide an outline [6] in Bangla which can be converted into speech by means of using any on-line platform [5] and as a consequence they can be benefited. Robotics is a buzzword in contemporary technology. Lately NASA has sent a robot on Mars that's sent to investigate if there are lives on Mars. By this model, a robotic model can be capable of describe any situation [2], [4], [6] in Bangla and might talk with people. It is able to provide an awful lot effect on human-robotic interplay [5]. Even this machine can be perfected at this type of degree in which machines may be capable of generate guidance set for humans. Another motivation at the back of this paintings is to generating Bangla caption from CCTV footage. The use of CCTV cameras within the world in addition to in our country has dramatically improved in latest years for safety functions. But it is not constantly viable to take a seat in the front of the reveal all time. It's far monotonous and additionally useless. Video captioning may be used to clear up this trouble. It may test any pictures and describe what is occurring, in a short time [3], [5]. In our country, many people can't apprehend English sentences. So, they're unable to take the gain of video captioning they may be benefitted if the generated caption is in Bangla.

1.3 Objective

- Firstly, our essential goal is to detect the primary items from a video and describe it in a Bangla sentence. As we are running with video, so we need to supply concern on temporal capabilities in addition to spatial functions [1]. We give interest [3] to this reality and will try to generate real-time captions [4], [5].

- Secondly, as Bangla is a complicated language so it is tough to generate grammatically accurate sentences. This is another objective to generate correct sentences.

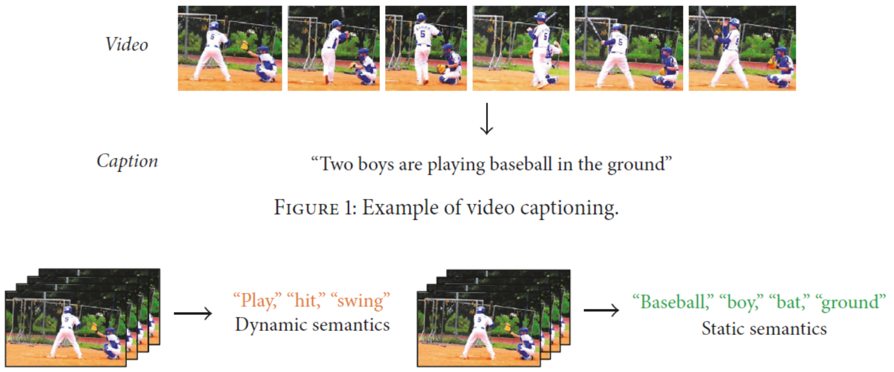

Video
Caption
"Two boys are playing baseball in the ground"
FIGURE 1: Example of video captioning.
"Play," "hit," "swing"
Dynamic semantics
"Baseball," "boy," "bat," "ground"
Static semantics

Figure 1.1: Example of Video Captioning.[6]

1.4 Challenges

For doing the task of video captioning in Bengali we have faced some major challenges. The first challenge is the unavailability of dataset. As we are using deep learning approach, and this deep learning approach is very data hungry so we need a lots of data. But Bengali video dataset is unavailable. Though some machine translated Bengali datasets are available, but they are not up to the mark as Bengali is a very complex language and a machine translated caption is far away then human annotated caption. After all machine translated sentences sometimes syntactically incorrect which decreases the accuracy of the model. The next challenge is unavailability of resources. For processing the video data or to train the model we need huge amount of resources. As our approach is attention-based and our dataset is quite big, so we need high power GPU to train the model. Again, all the available model cannot recognize Bengali alphabet. Some models are pre-trained with English datasets. So, they cannot recognize Bengali words.

Chapter 2

Literature Review

2.1 Video Based Action Recognition using Spatial and Temporal Feature.

This paper proposes a two-stream human action recognition architecture. It is the result of combining both spatial and temporal feature streams. It empowers to give us the two highlights for video-based activity acknowledgment. As the contribution of the spatial transfer, singular video outlines are removed. What's more, the contribution of the profound learning network is the extricated optical stream pictures. Optical stream is a for every pixel expectation and the principle thought is that it's anything but a brilliance steadiness, which means it attempts to assess how the pixels splendor gets across the screen over the long run. The spatial layer is utilizing a RGB outline for catching the appearance (static article), the transient layer is utilizing stacked optical stream for catching the movement data (dynamic item).

Working Approach:

1. First and foremost, video is deteriorated into spatial and fleeting segments where convolutional Organization or, convNet is utilized to catch the appearance data from singular edges. The unique highlights between back to back outlines are likewise extricated.

2. Singular edges are resized into 224*224 and took care of to the convolutional Neural Organization or, CNN. For the better use of the spatial data, fc6 layer of CNN are separated.

3. To get extra movement data optical stream pictures are utilized. It contains dynamic highlights of continuous edges. It can catch the distinction (development) between two casings in pixel level which makes the worldly highlights particular. In this way, it's anything but a superior precision.

4. The determination of the 'graphic' focuses in a video outline is finished utilizing learning visual highlights.

5. Transient data can't be gotten straightforwardly, that is the reason the optical stream is taken as the portrayal of fleeting highlights. Flat and vector segments are utilized here to create the optical stream pictures which is stacked and shipped off the CNN.

6. The contribution of fleeting CNN is this stacked optical stream. AlexNet is utilized as the profound learning organization. The highlights are separated in the element combination grid and can do the assignment of characterization.

2.2 Long-term Recurrent Convolutional Networks for Visual Recognition and Description.

The authors of this research were the first to tackle the video captioning challenge with a deep neural network.They presented three video description architectures.Their model is predicated on the premise that after a full pass of the entire movie, they will have CRF-based predictions of subjects, objects, and verbs.This enables the architecture to view the entire video at every time step.

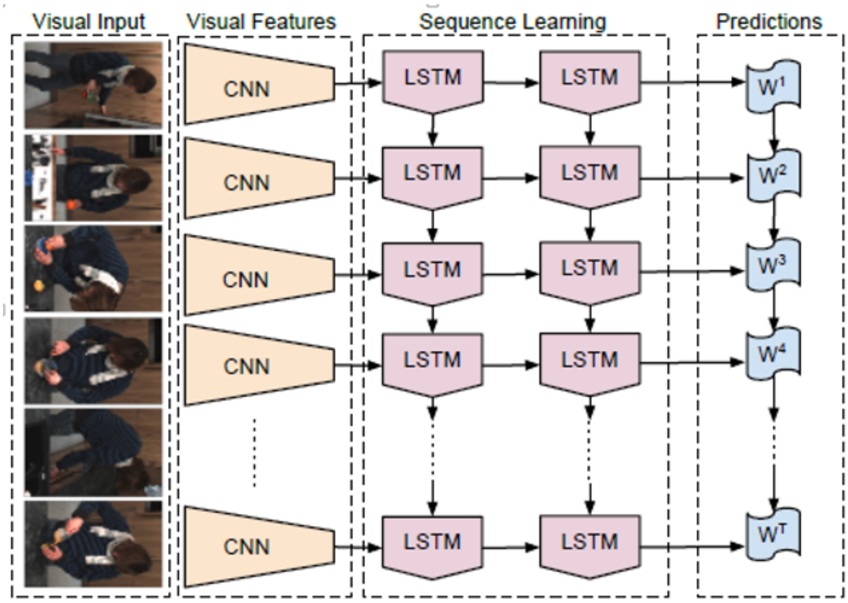


Figure 2.1: Long-Term Recurrent Convectional Network.[2]

Working methodology:

1. In Long Short-Term Memory (LSTM) encoder and decoder with Conditional Random Fields (CRF) max. Model, they replace the Statistical Machine Translation (SMT) with an LSTM model. 2. An encoder LSTM encodes the one-hot vector of the input sentence.

3. The last secret unit of the past advance is the contribution to the decoder stage in which the secret portrayal is decoded into a sentence, single word at each time step.

4. In LSTM decoder with CRF max. Model, they give the whole visual info portrayal at each time step to the LSTM.

5. Third model, LSTM decoder with CRF probabilities, is mostly same as the second one with the replacement of max predictions with probability distributions.

2.3 Video Captioning with Attention-based LSTM and Semantic Consistency.

In this paper, the creators had proposed a system aLSTMs, which is executed by at the same time minimizing the pertinence misfortune and semantic cross-view misfortune. In this paper, they had created to include captioning of recordings obliging semantic implications based on the consideration show. In spite of the trouble of video captioning, there have been a number of endeavors that are primarily propelled by later progresses in deciphering with Long Short-Term Memory (LSTM).LSTM is utilized to avoid the Gradient-Descent issue as well because it is utilized in discourse acknowledgment, Language captioning as well as picture captioning. In this way the creators had chosen to amplify LSTM to produce the video sentence with wealthy semantic contents.

Working methodology:

1. The visual features are generated by a VGGNet and C3D network as well as it proposes a visual-semantic embedding, which enforces the relationship between the entire sentence semantics and the visual content.

2. The creators utilized 2 sorts of misfortune capacities. One misfortune work points to ensure the interpretation from recordings to words, whereas another misfortune work tries to bridge the semantic hole with semantic cross-view correlations.

3. Within the LSTM, there's a context vector which may be a dynamic representation of the pertinent representation of the video input to the LSTM 4. In this paper, first all the videos are gone through LSTM visual encoder to generate CNN features.

5. After getting the features, those features are become the context vectors and pushed as the input of attention based LSTM.

6. At that point the created yields are gone to Multi-model word implanting framework at that point goes to the word highlights which contain semantic highlights. In this way creates a semantic captions of a video.

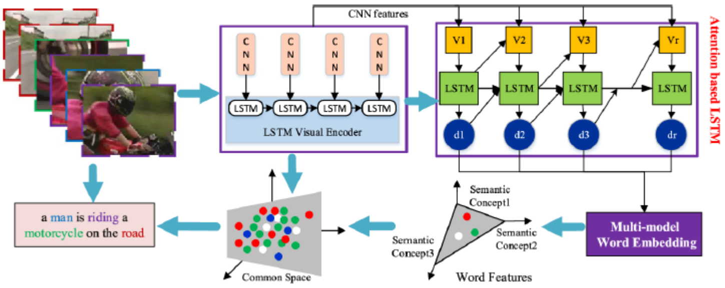


Figure 2.2: Semantic Cross-view Correlation: Semantic Cross-view Correlation.[3]

2.4 Video Captioning with Multi-Faceted Attention.

The authors of this research developed a revolutionary video captioning approach based on an extensible multi-faceted attention mechanism.They also looked at the robustness of our multi-faceted attention and discovered that, despite noise in the characteristics and qualities, its effectiveness remained consistent.For e.g., this model can pay attention to temporal information, motion aspects, and semantic attributes in a variety of ways.

Working methodology:

1. For visual features, they extract one feature vector per frame leading to the series of referring to as temporal vectors.

2. There are 2 types of Input videos: regular features and semantic attributes. Word embedding matrix can be used to transform semantic attributes to semantic embedding vectors.

3. In this paper, their core part is LSTM sentence-generator which input comes from multi-head attention instead of coming from word embedding. This is their exceptionality.

4. They chose a video as semantic attributes and made 16 frames of the video as 5. Motion features and made a single frame from that as temporal features and pass them at multifaceted attention mechanism.

6. Then the output of multifaceted retains as the inputs of LSTM sentence generators.

7. The LSTM outputs are then sent back into the multifaceted process, where the softmax function is employed to generate output captions or vectors that indicate probability distributions for a set of possible outcomes.

8. As a result, subtitles are created from a movie with various faces or objects.

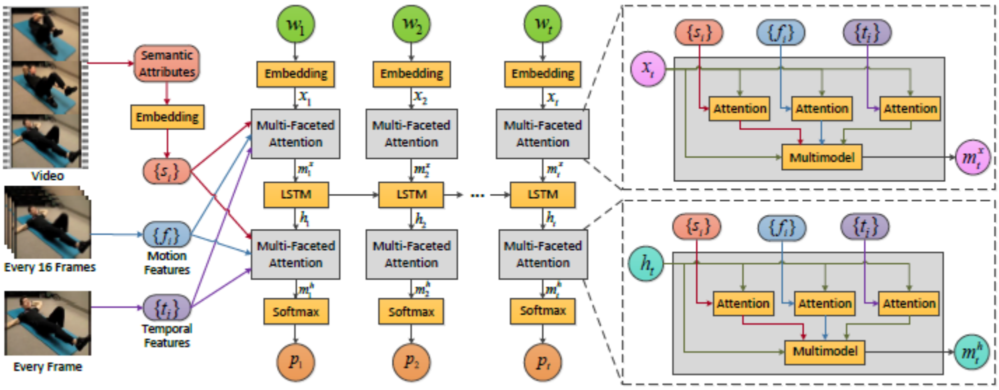


Figure 2.3: Model architecture. Temporal, motion, and semantic features are weighted via an attention mechanism and aggregated, together with an input word, by a multimodal layer. Its output updates the LSTM's hidden state. A similar attention mechanism then determines the next output word via a softmax layer.[4]

2.5 A Better Use of Audio-Visual Cues: Dense Video Captioning with Bi-modal Transformer.

Dense Video Captioning with Bi-modal Transformer for bi-modal input which appears viability for sound and visual modalities on the Dense video captioning assignment. The demonstrate is able of processing any two modalities in a sequence-to-sequence assignment. In this paper, they moreover appear that the pre-trained bi-modal encoder can be utilized as a include extractor for Proposal Generation Module.

Working methodology:

1. This model extracts feature using VGGish (audio), I3D (Visual), and Glove (captions) pre-trained models.

2. ActivityNet Captions dataset with 100k temporally localized sentences for 20k YouTube videos is used for training (50

3. The audio and visual features are processed by the bi-modal encoder layers by utilizing bi-modal multi-headed attention. The encoder(n) layers pass the modalities to the proposal generator.

4. Output of the proposal generator are used to clip the visual input modality. 5. Then the clips from the visual input passes through the bi-modal n layers encoder again.

6. Output from the encoder layers passes in every layers of decoder bi-modal attention block.

7. The decoder also takes previous caption words as input and produces context for language representation.

8. Finally, the decoder output passes into a generator to produce the next word.

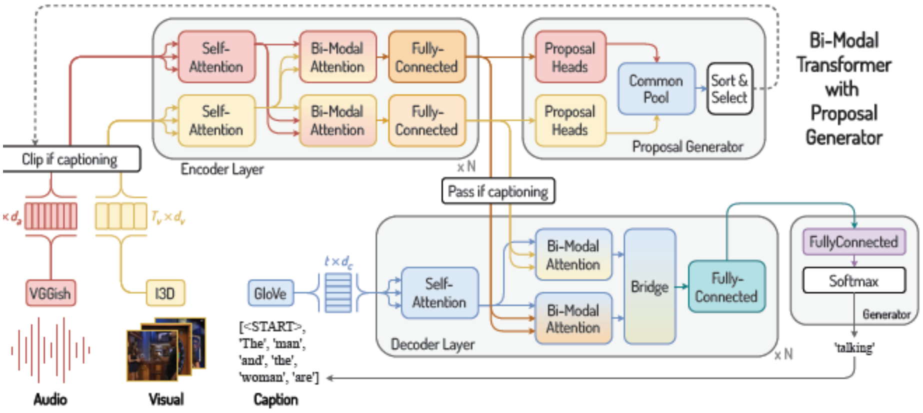


Figure 2.4: The design of Bi-modal Transformer with Multi-headed Proposal Generator.[5]

2.6 Multimodal Feature Learning for Video Captioning.

This article proposes Semantic Feature Learning and Attention-based caption generation or, SeFLA, a deep learning model utilizing both visual and semantic features for the effectiveness of video captioning tasks. They apply both dynamic (action) and static semantic features (object, person, background) detection the video frame.

Working methodology :

1. Proposed model has three parts: a) visual feature extraction, b) semantic feature extraction and c) sentence generation.

2. The MSVD dataset(1970 videos) and a video caption dataset (80,000 captions) from YouTube videos are used for training (1200), validation (100), and test (670). MSR-VTT dataset is also used for categorical videos.

3. Visual features are extracted using pre-trained model Residual Neural network or, ResNet (2D features) and C3D (for 3D features).

4. Then extracted visual features pass through the Dynamic Semantic Network (DSN uses the C3D output) and static semantic network (SSN uses the ResNet output).

5. ResNet output also passes as input in the LSTM to extract the visual features. Output of LSTM is used as initial input for Caption Generation Network or, CGN to determine the specific semantic information and calculate the probability of given word.

6. Both DSN and SSN features are linked to serve as the input of CGN. CGN applies the attention mechanism on the concatenated output feature of the DSN and SSN differently.

7. Then the caption is generated based on the probability distribution from CGN final output.

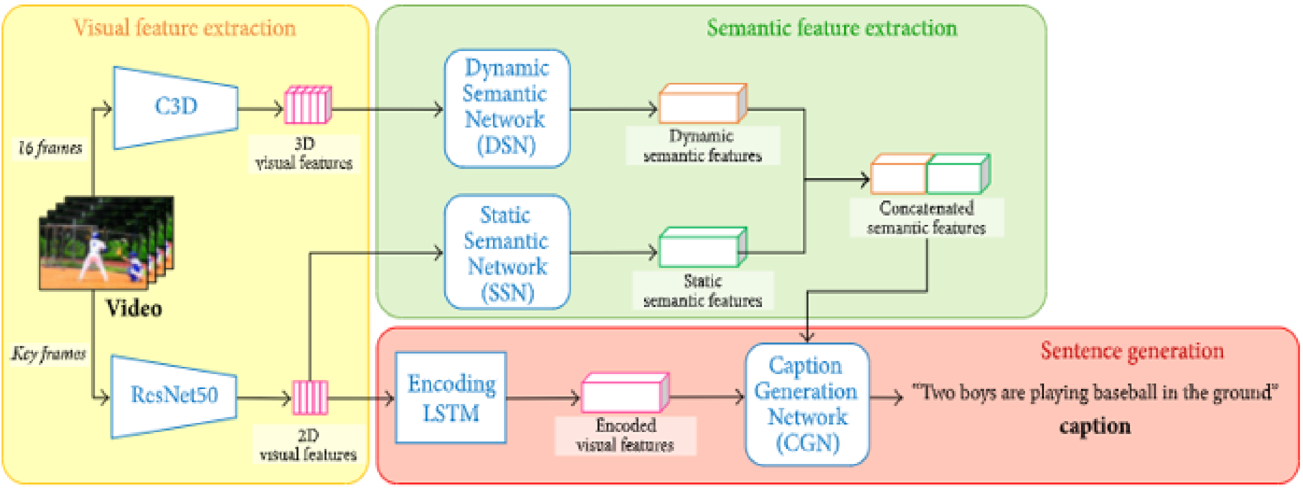

Visual feature extraction
16 frames
C3D
3D visual features
Video
Key frames
ResNet50
2D visual features
Semantic feature extraction
Dynamic Semantic Network (DSN)
Dynamic semantic features
Static Semantic Network (SSN)
Static semantic features
Concatenated semantic features
Sentence generation
Encoding LSTM
Encoded visual features
Caption Generation Network (CGN)
"Two boys are playing baseball in the ground"
caption

Figure 2.5: Overall framework of the proposed video captioning model.[6]

2.7 ViLBERTScore: Evaluating Image Caption Using Vision-and-Language BERT.

In this paper, the authors propose ViLBERTScore, a metric for image captioning tasks by using pre-trained Visio-linguistic representations. By employing VilBERT, which is a task-agnostic pre- trained Visio-linguistic representation. ViLBERTScore computes cosine similarity between token embedding for reference and candidate sentences similar to BERTScore. This model ViLBERTScore correlates better with human judgments than all of the previous metrics.

Working methodology :

1. ViLBERT model consists of a self-attention based embedding layer and co-attention layer for each image and text information. This model employs 2 streams of transformer based architecture. One of each part processes visual and textual inputs, respectively.

2. The image and grounded-text inputs are fed into separate embedding layers; followed by two co-attentional transformer block that allows interaction between the two modalities.

3. ViLBERT is pre-trained with two 36 training objectives, masked multi-modal modeling, and multi-modal alignment.

4. They had used two versions of ViLBERT, one from the pre-trained ViLBERT model from and the other version from that are fine-tuned on 12 downstream tasks.

5. They set N = 100 boxes for each image using image detection model to compute contextual embedding.

6. The co-attentional block in ViLBERT is composed of six layers. To verify the effective-

ness of each layer in computing the contextualized embedding of the data, they compute ViLBERTScore using the outputs of different layer.

7. By gathering candidate caption and reference caption they compute contextual embeddings with ViLBERT respectively. Then, they extract the text embedding and for each output embedding. Finally, they compute the pairwise cosine similarity between the embedding to get max similarity of the image.

8. Then those embedding are pushed through attention based 2 streams of transformers and then putting them on the embed they get results.

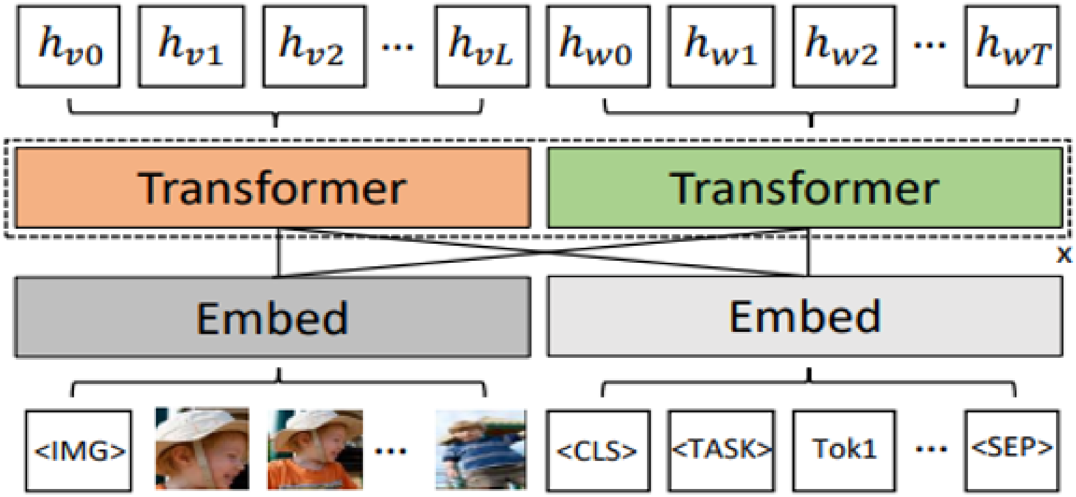


Figure 2.6: The overall architecture of ViLBERT. ViLBERT consists of a self-attention based embedding layer and co-attention layer for each image and text information[7].

2.8 BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.

Bidirectional Encoder Representations from Transformers or BERT's model architecture is a multi-layer bidirectional Transformer encoder based on the original Transformer. It is designed to pertain to deep bi-directional representation from the unlabeled text. Among the two strategies of applying pre-trained language representations, fine-tuning is used here. In this strategy, a masked language model or MLM has been used for pre-training the objective. It can be used in Virtual Question Answering or VQA. For this purpose, it is pre-trained with the SQuAD dataset.

Working methodology:

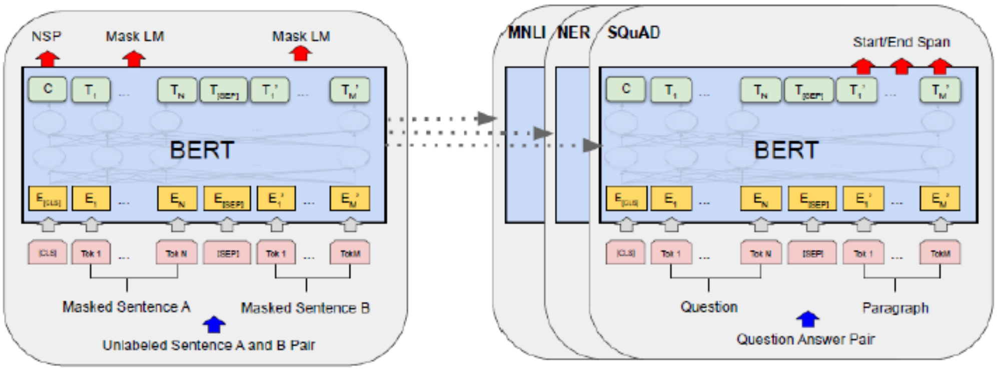
1. Firstly, two steps of training are executed. The first step is to pre-train BERT to understand the language and the second one is fine-tuning to learn specific task.

2. In the process of MLM, a sentence is masked in few positions and the goal is to find the masked token correctly. It improves the power of ordering of the words in a sentence.

3. After that the output is compared with the actual words and cross entropy is taken place. By this the model learns a language.

4. The next process is Next Sentence Prediction or NSP. In this approach, two sentences are given and the model predicts whether the two sentences are connected or not. It gives binary output about the prediction.

5. For the VQA section, a question with corresponding paragraph is given and the model can find the answer.



Pre-training **Fine-tuning**

Figure 2.7: Overall pre-training and fine-tuning process of BERT.[8]

Chapter 3

Background Study

For doing the task of video captioning we should integrate two sections of machine learning. One among them is Natural Language Processing or NLP and some other is computer vision or CV. in the CV segment, we need to extract the features from the video frames, and within the NLP phase is for producing the caption in Bengali. For this, we have studied different techniques within the subject of NLP and CV. They may be stated beneath:

3.1 Natural Language Processing or NLP

Natural Language Processing or NLP is involved with the interactions among computers and human language. Natural language refers to the manner we humans talk with every different. NLP is the technique of processing and analyzing massive amounts of natural language records. It is the automatic manipulation of natural languages like speech and text. It builds a shrewd system able to expertise text and speech. Human language is complicated, ambiguous, and various. There are numerous languages everywhere in the world and every language has one-of-a-kind guidelines. The first involved in assisting machines to understand natural language is to convert records into something that they can interpret. This level is called pre-processing. On the other hand, feature selection is the process of selecting a subset of the terms in the training set and using those subsets as features in text classification. It makes the version more ideal by using reducing the dimensions of the effective vocabulary. For overfitting, the model may be too complicated. It is a method for replacing a complex classifier with a simple one as here the model is trained with less but important features. In machine learning, if we need that our version is of higher accuracy then we must train our model with care. It is not the truth that the simplest massive datasets can make a version better. Honestly, too many useless features increase the threshold value and the accuracy gets lower. It's far known as overfitting. To triumph over this, we need a feature selection technique. There are different techniques for feature selection. They are:

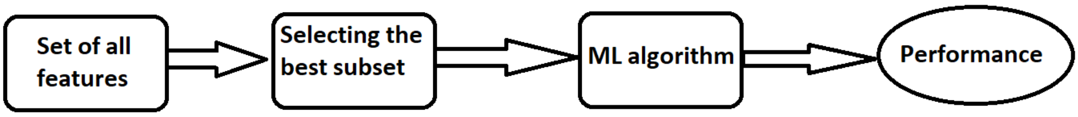


Figure 3.1: Filter Method.

1. Filter method:

In this technique, a set of all features are given. Amongst them, the nice subsets are chosen and they may be given as the input of a machine learning algorithm. And after acting all the steps the accuracy is measured. Inside the filter method, there are different strategies for deciding on the best subsets. Along with: Anova test, Chi-rectangular, co-relation co-efficient, and many others. ANOVA test is used to check the means of two or more groups that are significantly different from each other. In co-relation co-efficient, if X and Y are two features of a model and with the increase of X, Y is also increasing then it is said that they are highly co-relative.

2. Wrapper method:

There are 3 techniques in this method. They are: • Forward selection: It is an iterative technique wherein we can add features in each new iteration and test the accuracy. Allow, we have got five features of a, b, c, d, and e. In the first iteration, we take none of them. Then we take only ‘a’ and take a look at the accuracy. After that, we take ‘a’, ‘b’ and test the accuracy.

• Backward elimination:

Right here, we start with all of the features and removes the least significant feature in every iteration. Together with: if we apply the Chi-square approach then we will get a cost of ‘P’. If $P > 0.05$, then we are able to do away with the function.

• Feature elimination:

It is a greedy optimization algorithm that pursuits to discover the excellent features. It creates a model again and again and continues aside. It constructs the next model with the left features until all the features are exhausted.

3. Embedded method:

Here, all the permutation of the features are made and each of them is tested and among them which accuracy is highest that is chosen.

4. Univariate selection:

Statistical tests can be used to select those features that have the strongest relationship with the output variable.

5. Feature importance:

It gives us a score for each feature of the data, the higher the score more important the feature is to the output variable.

6. Correlation matrix with heatmap:

Co- relation can be positive or negative. Heatmap makes it easy to identify which features are most related to the target variable.

3.2 Computer Vision or CV

In artificial intelligence, computer vision is a area that trains computer systems to interpret, analyze and recognize the visible matters. The usage of digital images from cameras and videos and deep learning models, machines can precisely perceive and determine the objects. On the very beginning computer vision basically come across edges of objects to classify it in a few criteria by using some neural network model. The Optical character recognition (OCR) is the primary commercial use of computer vision which interprets the typed or handwritten textual content. Facial recognition programs released as massive units of images available online for evaluation. Machines can already recognize unique humans in images and videos for these increasing data sets. Computer vision is a technological field that ambitions to use its ideas and models to the development of computer vision systems. A computer vision system's organization is essentially application-dependent. Some systems are standalone applications that tackle a particular measuring or detection problem, whilst others are a part of a much wider design. image processing and image analysis are broadly speaking concerned with 2d images, namely how to turn one image into another using pixel-wise operations like contrast enhancement, local operations like side extraction noise elimination, geometrical transformations. This description implies that image processing and analysis do not necessary to assume or result in interpretations of visual information.

Obtaining Images:

One or more image sensors provide a digital image, together with range sensors, tomography devices, radar, ultrasonic cameras, and so on. One or more image sensors offer a virtual image, including range sensors, tomography gadgets, radar, ultrasonic cameras, and so forth. Pixel values are commonly associated with mild intensity in a single or more spectral bands (grey images or color images), but they also can be related to physical parameters like intensity and absorption.

Image Pre-Processing:

Before a computer vision approach can be used on picture data to extract a specific piece of information, it must first be applied to the data. Some methods are used to pre-process the image. Re-Sampling method is used to verify the coordinate of an image. Noise reduction and contrast enhancing are used to reduce the sensor noise and detect the information respectively. To improve image structures, use a scale space representation. Image Features Extraction:

The image data is used to extract visual features like lines, edges, ridges and Localized interest points of varying levels of complexity.

Image Segmentation:

A judgment is made regarding which picture points or portions of the image are important for further processing at some point during the processing. In image segmentation a specific set of point or specific interests are selected.

Image-understanding systems (IUS) are a massive part of computer vision. They have 3 tiers of abstraction: low level, which incorporates image primitives like edges, texture elements, or regions; intermediate level, which includes boundaries, surfaces, and volumes; and high level, which incorporates objects, scenes, or activities. Many of these requirements are open to additional investigation.

3.3 Convolutional Neural Networks(CNN)

Convolutional neural network or CNN is a class of deep, feed-forward artificial neural networks that are applied to analyzing visual imagery. It is a big example of the real time use of the neural network. Image classification is one of the most common problems where artificial intelligence is applied to solve. CNN is a type of neural network that is most often applied to image processing. Its working procedure is more likely as a human brain. The visual cortex system of human brain is trained such a way that it can recognizes an object, though it is not clearly visible. But it is very tough for machines. CNN has brought this power of recognition to the machines. Here, the word convolution means scanning. The scanning process is done by different types of filtering.

Machines can take input of an image by pixels. It cannot see the whole object. Instead, it only sees numbers in the image. Each pixel in an image is given a value of 0-255. The pixels are filtered in many steps and they detects curve, color, edge in the picture. In figure: 3.2 we can see the steps of CNN. We have to give an image as a input of the system and after calculation it will give us the predicted classification.In the first step, convolution is taken place.

Here, a color image of 2 is selected for classification. As it is a color image so it has 3 channels (Figure: 3.3.1) of Red, Green and Blue. Convolution is performed on an image to identify certain features in an image. It helps blurring, sharpening, edge detection etc. that help the machine to learn specific characteristics of an image.

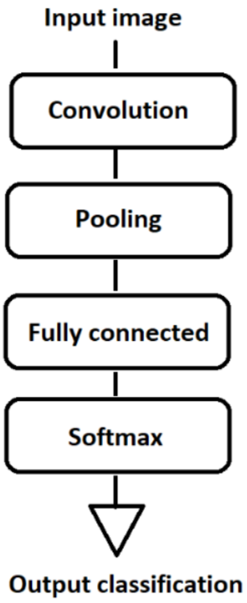
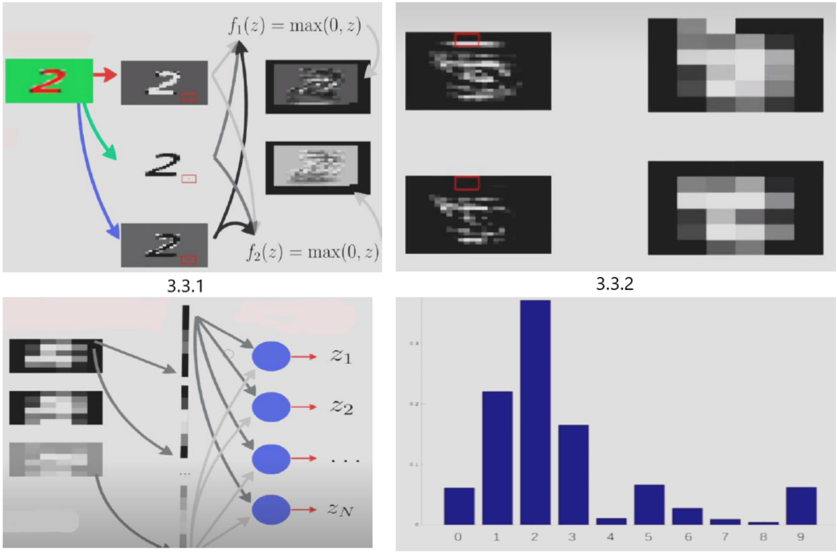


Figure 3.2: Steps of CNN.



3.3.1

3.3.2

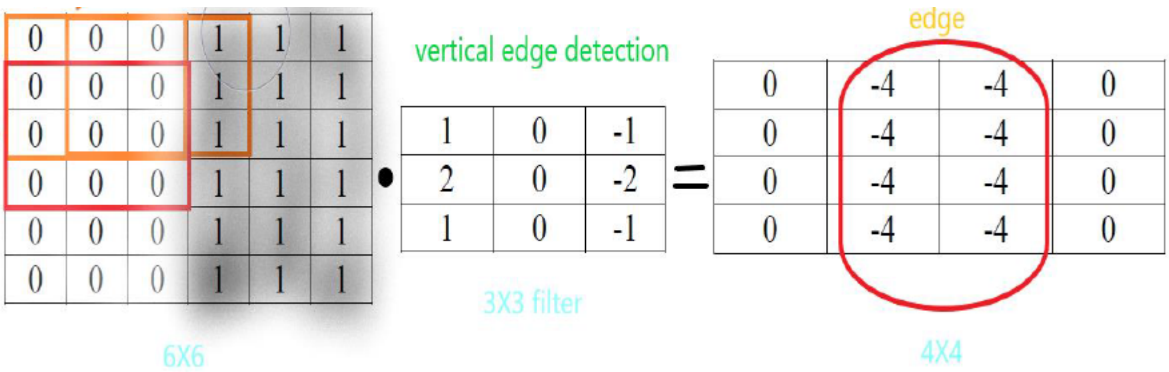
3.3.3

3.3.4

Figure 3.3: Working Procedure of CNN.[14]

A filter is used to do this task. Let, we have a picture which half part is black and another half part is white. It can be represented as (Figure:3.4).

Here, the white part is represented by 0 and black part is represented by 1 in the 6X6 matrix. The value of filter matrix (3X3) is generated randomly. The filter matrix is mapped in the



$$\begin{bmatrix}0&0&0&1&1&1\\0&0&0&1&1&1\\0&0&0&1&1&1\\0&0&0&1&1&1\\0&0&0&1&1&1\\0&0&0&1&1&1\end{bmatrix}\bullet\begin{bmatrix}1&0&-1\\2&0&-2\\1&0&-1\end{bmatrix}=\begin{bmatrix}0&-4&-4&0\\0&-4&-4&0\\0&-4&-4&0\\0&-4&-4&0\end{bmatrix}$$

Figure 3.4: Vertical Edge Detection[14]

6X6 matrix parallely (orange direction). The DOT product is generated and it is placed in the 4X4 matrix. It is called pooling (figure: 3.3.2). In this way it fills the parallel image. Now, in the red direction it fills the vertical image. If we look at the 4X4 matrix and replace -4 with 0 and 0 with 255 then we can easily understand the edge. It is called min-max theorem. This 4X4 matrix is used in classification. The other features are also fitted in the convolution layer. In the next step, fully connected (figure: 3.3.3) method is applied. The output of convolution layer are fitted in the fully connected layer and it tells us what the object is. In this step, the neural network works according to the weights that are given randomly. Here, normalization and regularization is also used. Such as ReLU, Sigmoid function etc. It is one kind of activation function of ANN. There is a formula for finding the size of the resulting matrix. Accuracy in image recognition is one of the most important advantage of CNN model as well as, it can automatically detects edge, features and so we have to give less effort.

3.4 Pre-processing of text

We can consider text as a chain of characters, phrases, terms and so on. In NLP we paintings with the phrases within the textual content. The machine treats a textual content as a series of words. The words are used to make a model which can predict the output. For this, we should acquire textual content data or words. The primary and maximum important step once we acquire the textual content information is the pre-processing. There are specially 4 techniques which are used in this step. They may be:

i. Cleaning:

In the cleaning process several tasks are done. At first, punctuations (. , ? ; :) and symbols are removed. If we are working with a social media dataset then we may find different emoji's and links in the dataset. Links are started with 'https', so we have to remove it. Next, we should convert all the words in small letter. If we don't do this then 'Hello' and 'hello' will be recognized as different words. But here is a problem. 'US' is used as a name of a country and it is also used to indicate a group of people. So, the line Karim said, "Hello!! How are you?" will be karim said hello how are you.

ii. Tokenization:

It is the second step of pre-processing. It is a process that splits a sentence into so-called tokens or words. In English we use whitespace as a tokenizer. Simply, it is a process of breaking down of sentences into words. We can not work with alphabets or full sentences, that's why it is applied. Each individual word holds a meaning. As an example we can consider the previous sentence. The output will be an array of words like ['karim', 'said', 'hello', 'how', 'are', 'you']. Here one problem is occurred. In the sentence of this is karim's table. If we split the sentence then we will get a word karim's. Though it is same as karim, but they will be treated as different words.

iii. Stemming:

To solve the problem of tokenization this step is performed. Stemming is a process of removing and replacing suffixed to get to the root form of the word, which is called steam. It is one kind of chopping technique. Example: 'all males must fill up this form'. Here if we apply tokenization then we will get a word 'males'. Then 'males' and 'male' will be treated as two different words. By applying stemming we can remove the suffix 's' from the word 'males'. And it will be 'male' after stemming. But stemming is not always too much effective as here we only chop a particular portion of a word. Let, 'I liked the cat'. Here, if we remove 'ed' from the word 'liked', then it will be 'lik' which is not a meaningful word. Again, if we remove suffix 'ing' from the word 'playing', then it will be 'play'. It is correct. But if we remove 'ing' from 'ring', then it will be 'r'. there are many algorithms to apply stemming. Porter's stemmer is one of them. Here we have to follow 5 rules.They are :

Rule	**Example**
SSES –> SS	caresses –> caress
IES –> I	ponies –> poni
SS –> SS	caress –> caress
S–>X	cats –> cat

iv. Lemmatization:

It is also known as stop words removal. Stop words are those words which does not hold much value in the dataset like and, or, is, are, being etc. it is usually refers to doing things properly with the use of a vocabulary and morphological analysis. It returns the base or dictionary form of a word, which is known as the lemma. In the Stemming process we have chopped only distinctive suffix such as 'ing', 'ed' etc. It is just like a bruteforce approach. But lemmatization is a calculative process. It involves resolving the word to its dictionary form.

There is a disadvantage too in this process. As it needs dictionary background so it is a slow process. And different languages have different dictionaries. So we have to work on them too. Another negative point about this process is to if it removes ‘isn’t’ from a sentence then after lemmatization we can not detect weather it is a positive sentence or negative. There are different lemmatizer we can use. Such as: WordNet lemmatizer.

3.5 Visual Geometry Group (VGG)

Visual Geometry Group or VGG is a form of convolutional neural network. There are exclusive versions of VGG. along with: VGG-16, VGG-19. The variety after the word ’VGG’ suggests the number of layers in the architecture manufactured from. So, VGG-16 has sixteen layers whereina photograph is filtered to predict a result. The concept in the back of this model is to stack up layers to shape a very deep convolutional neural community that could deliver a excellent accuracy. And VGG-16 has accomplished so. It performed 92.7 percent accuracy in ImageNet that’s one in all the biggest dataset available. It has 14 million photos. Because of greater layers, it permits us to extract extra parameters. How a great deal the photograph goes forward toward the layers, the extra capabilities are extracted. right here, maxpulling layers have been used which reduce the scale of the function map to half of.

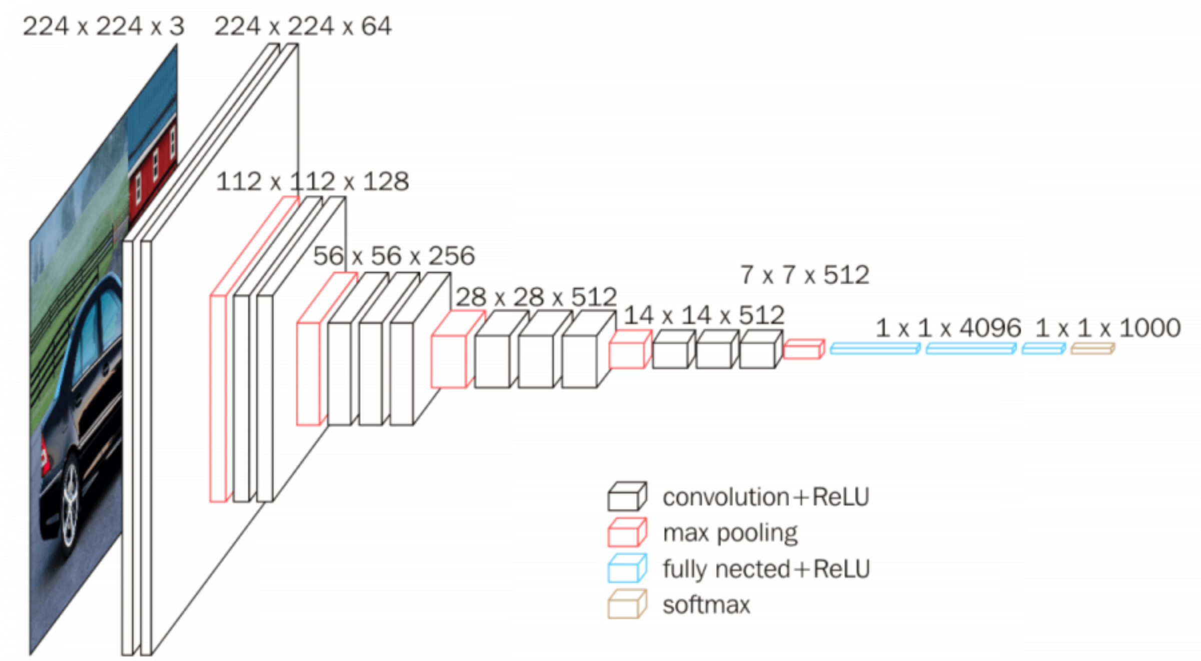


Figure 3.5: VGG-19.

In first step, VGG takes 224X224X3 pixel RGB image. As it is a RGB image so it has 3 layers. The first 2 convolutional layers have 64 channels of 3X3 filter. The image size is 224 pixels.

Here the padding is used according to the size of the picture. Then max-pulling is applied with stride (2, 2). We know,

output pixel after max-pulling = (n + 2P - f/s)+ 1

Here, n = 224, p = 2, f = 2, s = 2. So,

Output is = ((224+4-2)/2)/2 = 112 pixels,

which is the half of the input.

Then after a max-pool layer of stride (2, 2), 2 layers which have convolutional layer of 256 filter size and filter size (3, 3). This followed by a max-pooling layer of stride (2, 2) which is same as previous layer. Then there are 2 convolutional layers of filter size 3X3 and 256 filter. After that there are 2 sets of 3 convolutional layers and a max pool layer. Each have 512 filters of 3X3 size with same padding. This image is then passed to the stack of 2 convolutional layers. In these convolutional and max-pooling layers the filters we have used is of size 3X3. After the stack of convolutional and max-pooling layers we get a 7X7X512 feature map. We flatten this output to make it a 1*25088 feature vector. After this 3 fully connected layer comes. The first 2 layers, each has 4096 nodes. The first layer takes input from the last feature vector and outputs a 1X4096 vector. Second layer also outputs a vector of size 1X4096. But the third layer outputs a 1000 channels for 1000 classes. After that it passes through a soft-max layer in order to normalize the classification vector. All the hidden layers use ReLU as its activation function. ReLU is more computationally efficient. Because it results in faster learning and it also decreases the likelihood of vanishing gradient problem.

3.6 ResNext-101 Model.

The ResNeXt architecture is a refined variant of Deep Residual Network (ResNet), a CNN-based model. The ImageNet-5K dataset was used to train the ResNeXt-101 model. Split, transform, and combine are the three steps that the ResNeXt-101 model follows explicitly. ResNeXt offered a new dimension termed cardinality.The computational cost of deep neural networks is extremely high so in order to processing the original network with greater dimensions, ResNeXt separates it into multiple parallel and smaller networks, then adds these networks with reduced dimension. Figure x compares the performance of convolution on the huge input feature.

The first convolution layer contains 64 filters with a 7x7 window size and a stride of 2. This layer's output vector is 112x112 pixels in size. The next layer is a 3x3 max-pooling layer with stride 2. The pooling layer distributes the output to three subsequent ResNeXt blocks,

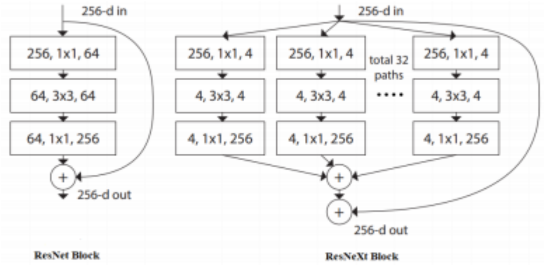


Figure 3.6: Difference between a ResNet Block and a ResNeXt block.

referred to as Conv2 layers in Figure x, each with a cardinality of 32. The conv2 layer's output is 56x56 pixels in size goes through conv3 layer made out of four ResNeXt blocks. This process continues until the conv5 layer is reached. The conv5 layer generates vectors with a 7x7 dimension. The average pooling layer passes across the 7x7 vector. The output of the average pooling layer is fed into a 1000-dimensional fully connected layer, which produces a 1x1 vector.

3.7 Evaluation Metrics

An evaluation metric quantifies the performance of a predictive model. This typically includes training a model on a dataset, the usage of the model to make predictions on a holdout dataset no longer used during training, then comparing the predictions to the expected values inside the holdout dataset. evaluation metrics are tied to machine learning obligations. There are different metrics for the tasks of classification, regression, ranking, clustering, topic modeling, and many others. As we are dealing with a machine translation output and a human translation so we have used BLEU score for overall performance measurement. We have also used ROUGE, CIDEr socres. They are described below:-

BLEU :

BLEU (Bilingual Evaluation Understudy) is an calculation for assessing the quality of content which has been machine-translated from one common dialect to another. Quality is considered to be the correspondence between a machine's yield which of a human: "the closer a machine interpretation is to a proficient human interpretation, the way better it is" – this can be the central thought behind BLEU.BLEU was one of the primary measurements to claim a tall relationship with human judgements of quality, and remains one of the foremost well known mechanized and cheap measurements. Scores are calculated for person deciphered segments—generally sentences—by comparing them with a set of great quality reference interpretations. Those scores are at that point found the middle value of over the complete corpus to reach an appraise of the translation's in general quality. Comprehensible or linguistic rightness are not taken into account.

BLEU's output is always a number between 0 and 1. This value indicates how similar the candidate text is to the reference texts, with values closer to 1 representing more similar texts. Few human translations will attain a score of 1, since this would indicate that the candidate is identical to one of the reference translations. For this reason, it is not necessary to attain a score of 1. Because there are more opportunities to match, adding additional reference translations will increase the BLEU score.

BLEU-1,2,3,4 (bilingual evaluation understudy) [11] [15] is a machine translation evaluation metric based on precision over n-grams, where an n-gram is a sequence of consecutive words of length n. For example, in the caption "the cat is on the mat" we would have: 1-grams (or unigrams): "the", "cat", "is", "on", "the", "mat" 2-grams (or bigrams): "the cat", "cat is", "is on", "on the", "the mat" 4-grams: "the cat is on", "is on the mat" Note, that a related concept of a "character n-gram" exists, where n refers to the number of individual characters instead of words. In our case, we are only interested in word n-grams. Within the following illustrations we are going outline BLEU-1 which is connected to unigrams, however the same approach generalizes to n-grams of any size. Let Count(u) be the number of words (or unigrams) within the caption that are too display within the ground truth, with each occurrence of a word expanding the number. We apply a unigram accuracy score to every caption we produce within the taking after way:

p(caption) = $\sum_{ucaption}$count (u)]/L

Given a created caption of length L, we increment the true-positive number each time we experience a word u that moreover happens within the ground truth. The issue with simple precision is that the true-positive checks may surpass the number of event of the word in the ground truth. To demonstrate this, we offer an illustration, Ground truth: the cat is on the mat. Caption 1 : the the the the the the the. Caption 2 : There is a cat on the mat. For Caption 1, the standard unigram accuracy score would be 7/7, and for Caption 2 it would be 3/7, indeed in spite of the fact that from the human viewpoint the 2nd caption is much closer to the ground truth. BLEU looks for to cure this issue by utilizing altered exactness. The modified precision for unigrams for a single caption, P*1 , is calculated by basically clipping the tally of occurrences of each word within the created caption to the most extreme number of occurrences in the ground truth:

p1* = $\sum_{ucaption}$CountClip(u)/L

Based on this condition, the adjusted unigram exactness score for Caption 1 gets to be 2/7 and for Caption 2 is once more 3/7, which is presently higher than for the clearly off-base Caption 1. In common, the higher the accuracy score, the higher is at that point the coming about BLEU-N score. The common frame of a adjusted n-gram accuracy, Pn, is connected to the whole set of candidate captions:

p_n =$\sum_{c_i candidates}\sum_{n-gram c_i}$ CountClip(n-gram) / $\sum_{c_i candidates}\sum_{n-gram c_i}$ Count(n-gram)

The final BLEU-N score is calculated by multiplying the weighted geometric mean of modified n-gram precision scores, Pn, for n = 1, . . . ,N by a brevity penalty:

$\text{BLEU-N} = \text{BP.}\ \exp_{/}\sum_{n=1}^{N} \text{w(n) ln p(n)}$

Here, the weights are regularly set so that wn = 1/N. Instinctively, BLEU-2 combines the modified accuracy P1 and P2, BLEU-3 combines P1, P2, and P3, etc. Returning to our prior case, where we have a single match of captions being compared, we can see that the created caption "There may be a cat on the mat." is longer than the ground truth caption, so we don't bring about any brevity punishment. In expansion, the geometric mean of a single number is the number itself, so our last BLEU-1 score is 3/7 = 0.4285. The crude yield of the BLEU-1,2,3,4 metric is within the run (0, 1), be that as it may, this output is ordinarily increased by a calculate of 100, to outline the ultimate score into a (0, 100) run. We follow the same approach when detailing our comes about.

ROUGE :

ROUGE is actually a set of metrics, rather than just one. We will cover the main ones that are most likely to be used, starting with ROUGE-N.

The following five evaluation metrics are available.

- ROUGE-N: Overlap of N-grams between the system and reference summaries.

- ROUGE-1 refers to the overlap of unigram (each word) between the system and reference summaries.

- ROUGE-2 refers to the overlap of bigrams between the system and reference summaries.

- ROUGE-L: Longest Common Subsequence (LCS) based statistics. Longest common subsequence problem takes into account sentence level structure similarity naturally and identifies longest co-occurring in sequence n-grams automatically.

- ROUGE-W: Weighted LCS-based statistics that favors consecutive LCSes .

- ROUGE-S: Skip-bigram based co-occurrence statistics. Skip-bigram is any pair of words in their sentence order.

- ROUGE-SU: Skip-bigram plus unigram-based co-occurrence statistics.

CIDEr :

Naturally portraying an picture with a sentence may be a long-standing challenge in computer vision and normal dialect preparing. Due to later advance in object detection, property classification, activity acknowledgment, etc., there's recharged intrigued in this zone. Be that as it may, assessing the quality of portrayals has demonstrated to be challenging. CIDEr could be a novel worldview for assessing picture depictions that employments human agreement[9]. This worldview comprises of three fundamental parts: a modern triplet-based strategy of collecting human explanations to degree agreement, a unused robotized metric (CIDEr) that captures agreement, and two modern datasets: PASCAL-50S and ABSTRACT-50S that contain 50 sentences portraying each picture. This simple metric captures human judgment of consensus better than existing metrics across sentences generated by various sources. We also evaluate five state-of-the-art image description approaches using this new protocol and provide a benchmark for future comparisons.

Chapter 4

Word Embedding

4.1 Word2Vec

Word2vec is a technique for natural language processing. It makes use of a neural network model to study word associations from a huge text. In other phrases we can say that, word2vec is a two-layer neural net that processes text by 'vectorizing' words After training, it could detect synonyms or advocate comparable words for a specific sentence. Word2vec represents or converts a word into a selected list of numbers referred to as a vector. The vectors are selected and some mathematical functions are implemented there. Then it shows the level of similarity between the words represented through those vectors. We will set up sophisticated complicated relationships between words in a higher dimensional space in Table 4.1 :

Words	**sam**	**is**	**person**	**a**	**busy**	**also**	**an**	**Intelligent**	**work**	**smartly**
Sam	3	2	2	1	1	1	1	1	1	1
is										
person										
A										
busy										
also										
an										
Intelligent										
work										
smartly										

We generally convert the sentence into bag of words. In bag of words method, semantic information is not always saved and there is a hazard of overfitting. Because many similar words are taken as exclusive words and as a result the threshold value will increase. Let a paragraph has a line of "Sam is a busy person. He is also very intelligent and works very smartly." From this paragraph, if we make bag of words then it'll look like as Table 1. Right here, in the matrix, the order is not always followed. All of the words are positioned randomly. But it will not predict output in many cases. The solution of the problem is word2vec. In this model, each phrase is basically represented as a vector of 32 or extra dimensions as opposed to a single number. As tons the dimensions increase, we can get better accuracy. Here, the semantic records and relation between exceptional words is likewise preserved. In the figure 5 we will see exceptional words like 'MAN', 'WOMAN' and 'PLAY' are positioned on a 2 - dimensional surface. The words 'MAN' and 'WOMAN' are pretty similar. And the word 'PLAY' is different from them. If we examine the graph we are able to see the fee of 'MAN' and 'WOMAN' could be very close to b.

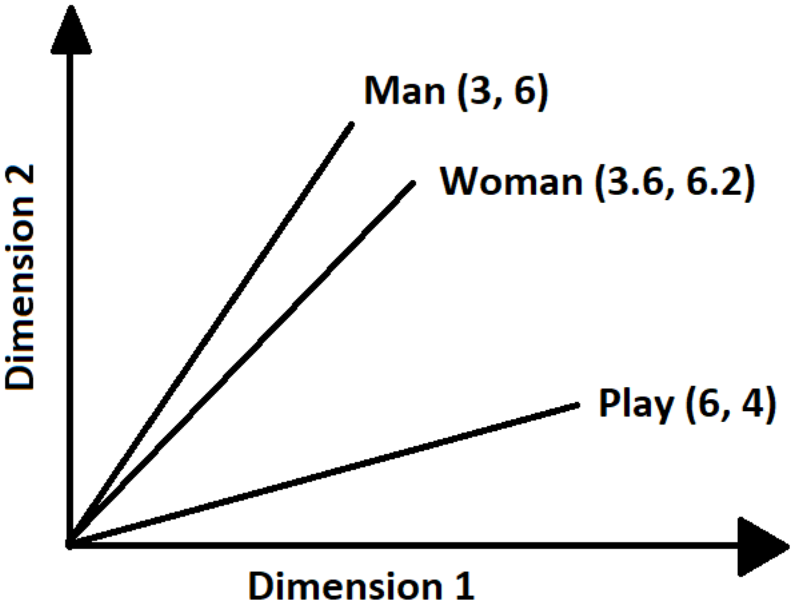


Figure 4.1: Dimension wise representation of words.

In this technique we get some advantages as the idea is very intuitive which transform the unlabeled words into labeled data by mapping the target word to its context word so, it is also easy to implement. Next, Mapping between the target word to its context word embeds the sub-linear relationship into the vector space of words such as king -> man, queen -> woman. It also saves memory.

Chapter 5

Attention Model

The word 'Attention' approach giving focus to something and taking a superb care of it. In deep learning, the attention mechanism is based on this concept of directing the model's awareness to a few factors which might be essential for the current result. The attention model has solved the problems of the encoder-decoder version and offers us a higher accuracy. The encoder-decoder version can address small sentences in case of translation. But when the sentence is just too large then it fails because the context vector can't keep track of too many phrases. Again the encoder-decoder model is uni-directional. It is going to most effective have data approximately nearest word. To triumph over these problems attention model is proposed where Bi-directional LSTM/RNN are used instead of uni-directional. As it is bi-directional so we get the future prediction of the words. And additionally, in the case of translation, the future words can have an effect on the prevailing translation. The main issue of accuracy of the attention version is the Bi-directional LSTM or RNN.

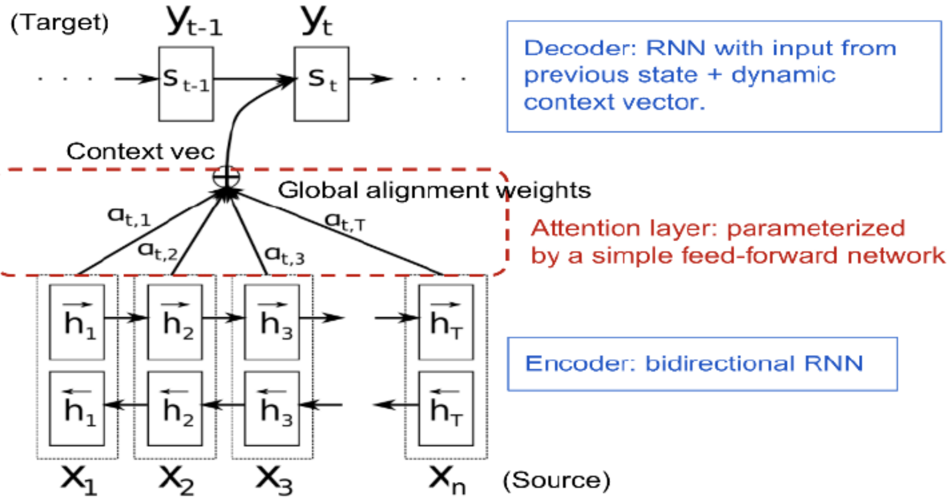

(Target)
y_{t-1}
y_t
s_{t-1}
s_t
Decoder: RNN with input from previous state + dynamic context vector.
Context vec
Global alignment weights
$a_{t,1}$
$a_{t,2}$
$a_{t,3}$
$a_{t,T}$
Attention layer: parameterized by a simple feed-forward network
$\overrightarrow{h}_1$ $\overrightarrow{h}_2$ $\overrightarrow{h}_3$ $\overrightarrow{h}_T$
$\overleftarrow{h}_1$ $\overleftarrow{h}_2$ $\overleftarrow{h}_3$ $\overleftarrow{h}_T$
Encoder: bidirectional RNN
x_1 x_2 x_3 x_n
(Source)

Figure 5.1: Attention Model Diagram.[2]

Figure 5.1 has given an idea about the layout of the attention model. Bi-directional RNN works as an encoder and uni-directional RNN works as a decoder here. X1, X2, X3.... are the input word vector. It is given as input of each the encoder layer. The word vector is passed via both these layers. each the layer produces its output and they may be mixed. Now they're surpassed within the context vector C1, C2, C3... These context vectors are created from the window size, Tx. that is every other crucial aspect. Tx = 3 means the model will provide attention to the first 3 words and after then it gives output. The context vector is getting derived from h1, h2, h3 which are the output of the bi-directional RNN.

C1 = h1+h2+h3+h4 ...(i)

If the output Y1 is dependent on the X3, then the effect of X3 will be present on Y1 because the output of X3 is merged with Y1. So, we are able to give attention to some of the words as we like. At a time we are giving attention to some of the words. Before going to the context vector the outputs are multiplied by some factors a1, a2, a3 etc. See in the below equation,

c^i = Sum $\sum_{j=1}^{Tx}$ aij*hj ...(ii)

Equation (ii) is the property of feed forward neural network (marked in red in figure 2). At first 'a' values are initialized and after that they are updated using back-propagation method. So, context vector is generated by multiplying the output of the bi-directional RNN with the 'a' values. There is a rule about the values of a. The summation of all the a is 1. So here,

a1+ a2+ a3+ a4 = 1...(iii)

We can re-write the equation (i) as,

C1 = a1*h1+ a2*h2+ a3*h3 + a4*h4 ... (iv)

Now the 'a' values are initiated using,

aij = [exp(eij)/$\sum_{k=1}^{Tx}$exp(eik)] .....(v)

where eij = a*s(i-1),hj

The main advantage of this model that it can predict future words in case of translation. It also gives us improved accuracy than encoder-decoder model.

Chapter 6

Our Approach

6.1 Why Attention?

The attention mechanism is a complex cognitive ability that human beings possess. When people receive information, they can consciously ignore some of the main information while ignoring other secondary information.

This ability of self-selection is called attention. The attention mechanism allows the neural network to have the ability to focus on its subset of inputs to select specific features. A neural network aimed with an attention mechanism can actually understand what "it" is referring to. That is, it knows how to disregard the noise and focus on what's relevant, how to connect two related words that in themselves do not carry markers pointing to the other.

Video captioning is a technique that bridges vision and language together, for which both visual information and text information are quite important. Typical approaches are based on the recurrent neural network (RNN), where the video caption is generated word by word, and the current word is predicted based on the visual content and previously generated words. However, in the prediction of the current word, there is much uncorrelated visual content, and some of the previously generated words provide little information, which may cause interference in generating a correct caption. Based on this point, we attempt to exploit the visual and text features that are most correlated with the caption

Inspired by the visual attention mechanism of human beings, temporal attention mechanism has been widely used in video description to selectively focus on important frames. However, most existing methods based on temporal attention mechanism suffer from the problems of recognition error and detail missing, because temporal attention mechanism cannot further catch significant regions in frames. In order to address above problems. we propose the use of a novel attention mechanism with spatailand temporal features for video captioningThe attention mechanism is a complex cognitive ability that human beings possess. When people receive information, they can consciously ignore some of the main information while ignoring other secondary information.

This ability of self-selection is called attention. The attention mechanism allows the neural network to have the ability to focus on its subset of inputs to select specific features. A neural network aimed with an attention mechanism can actually understand what “it” is referring to. That is, it knows how to disregard the noise and focus on what’s relevant, how to connect two related words that in themselves do not carry markers pointing to the other.

Video captioning is a technique that bridges vision and language together, for which both visual information and text information are quite important. Typical approaches are based on the recurrent neural network (RNN), where the video caption is generated word by word, and the current word is predicted based on the visual content and previously generated words. However, in the prediction of the current word, there is much uncorrelated visual content, and some of the previously generated words provide little information, which may cause interference in generating a correct caption. Based on this point, we attempt to exploit the visual and text features that are most correlated with the caption.

Inspired by the visual attention mechanism of human beings, temporal attention mechanism has been widely used in video description to selectively focus on important frames.

However, most existing methods based on temporal attention mechanism suffer from the problems of recognition error and detail missing, because temporal attention mechanism cannot further catch significant regions in frames. In order to address above problems. we propose the use of a novel attention mechanism with spatailand temporal features for video captioningThe attention mechanism is a complex cognitive ability that human beings possess.

When people receive information, they can consciously ignore some of the main information while ignoring other secondary information.

This ability of self-selection is called attention. The attention mechanism allows the neural network to have the ability to focus on its subset of inputs to select specific features.

A neural network aimed with an attention mechanism can actually understand what “it” is referring to. That is, it knows how to disregard the noise and focus on what’s relevant, how to connect two related words that in themselves do not carry markers pointing to the other.

Video captioning is a technique that bridges vision and language together, for which both visual information and text information are quite important. Typical approaches are based on the recurrent neural network (RNN), where the video caption is generated word by word, and the current word is predicted based on the visual content and previously generated words. However, in the prediction of the current word, there is much uncorrelated visual content, and some of the previously generated words provide little information, which may

cause interference in generating a correct caption. Based on this point, we attempt to exploit the visual and text features that are most correlated with the caption.

Inspired by the visual attention mechanism of human beings, temporal attention mechanism has been widely used in video description to selectively focus on important frames.

However, most existing methods based on temporal attention mechanism suffer from the problems of recognition error and detail missing, because temporal attention mechanism cannot further catch significant regions in frames. In order to address above problems. we propose the use of a novel attention mechanism with spatailand temporal features for video captioning.

6.2 Encoder

Our encoder is designed consisting of two parts of handling image features which is static features and video features which is dynamic features. The static features are also called spatial features and the dynamic features are also called temporal features. This temporal features are too important here as it is used for detecting the motion of the object or how the object differs from one frame to another. The spatial features are extracted using the VGG-19 model. We can see this in the (figure 7.1). Again, the ResNext-101 model is assigned to extract the temporal features from the video clips. Then both are given input of the encoder and the encoder combines both the features and it gives the final output vector where both the spatial and temporal features are combined. This is important approach in the field of video captioning. These spatial features are preceded down to a BiLSTM layer. The output of the BiLSTM layer is concatenated with the temporal features. Then it is passed through the activation function which is Tanh function followed by a dropout layer. The output of the Tanh function is sent as an input of the linear layer and finally we get a vector of features where both the features are combined.

6.3 Decoder

We have used word2vec technique for pre-processing the captions of the video. In this model we have given the words as input and get a vector as an output. This vector is given as an input of the decoder. The attention mechanism is mainly applied here to combine the feature vector which is produced by the encoder and the word vector which is produced by the word2vec model. The attention model is constructed here by passing the parameters of encoder dimensions, decoder dimensions and attention dimensions. All of the image is flatten to a particular size and the input data is sorted by decreasing lengths. At each time step, they are decoded by attention-weighing the encoder's output based on the decoder's previous hidden state output. Then it generates a new word in the decoder with the previous word and the attention weighted encoding model.

Chapter 7

Proposed Methodology

7.1 Overview

Our proposed model will consists of 2 main segments. They are:

1. Feature extraction from video

2. Caption generation

In our proposed model we have got used attention model in place of simple encoder-decoder model. Firstly, we have divided our dataset into two parts. One part is for feature extraction from video. As we have two different kinds of features of spatial and temporal so we have to give emphasize on both of these. For extracting the spatial features we have used VGG-19 model and for extracting the temporal features ResNext-101 is used.

We have extracted both the features and saved them as vector (.h5 file). Next, we have generated human annotated Bengali caption. The next step is to tokenizing the captions using word embedding techniques. We have used word2vec approach to do this task. The extracted features are given as input of the encoder. The output of the encoder goes as input of the decoder. We have implemented attention here. Another input of the decoder comes from the word2vec model. Then it maps the tokenized word with the extracted features. In this way our model is trained. In the testing period a video is given as an input of the model and we get the caption in Bengali.

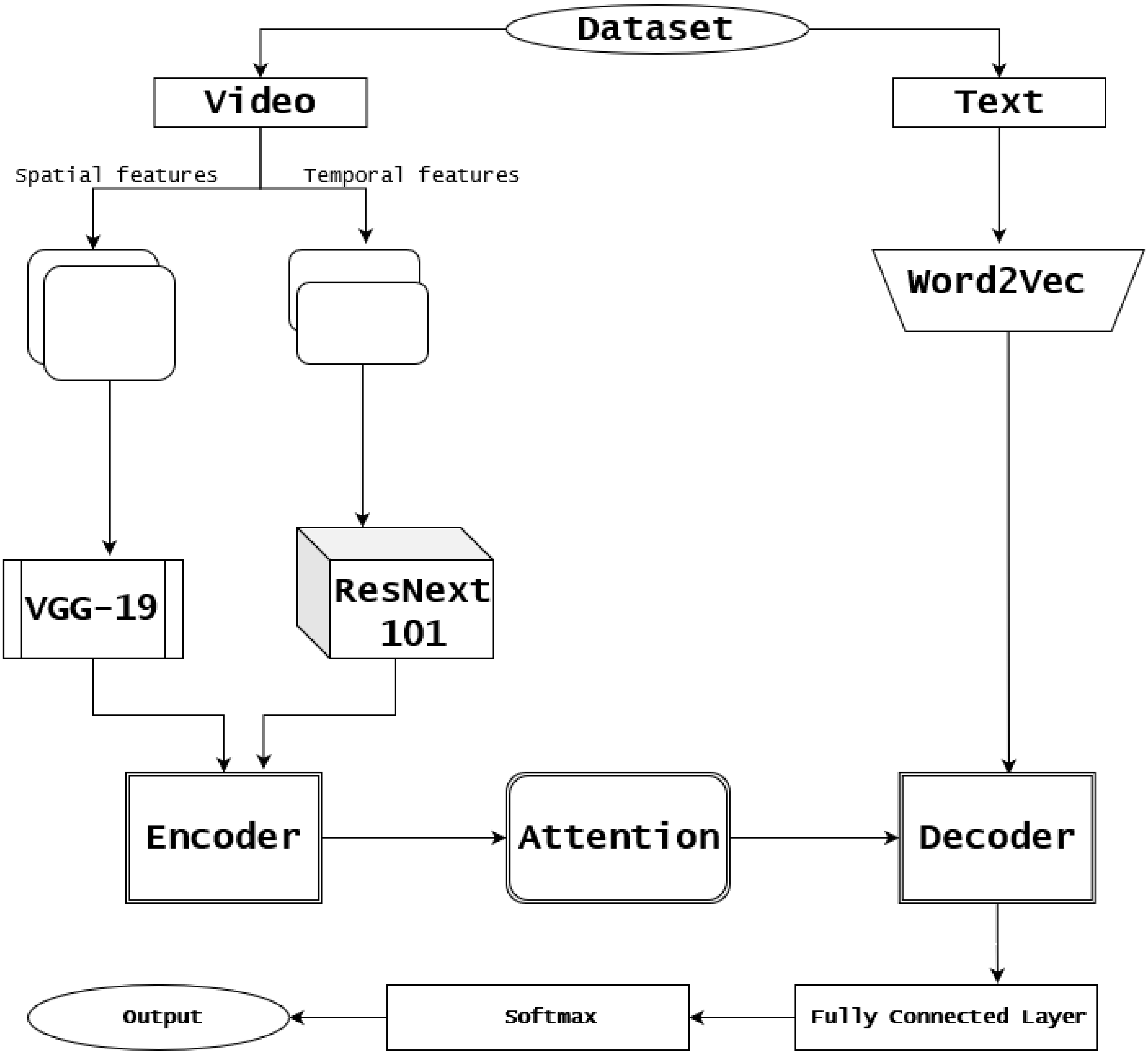


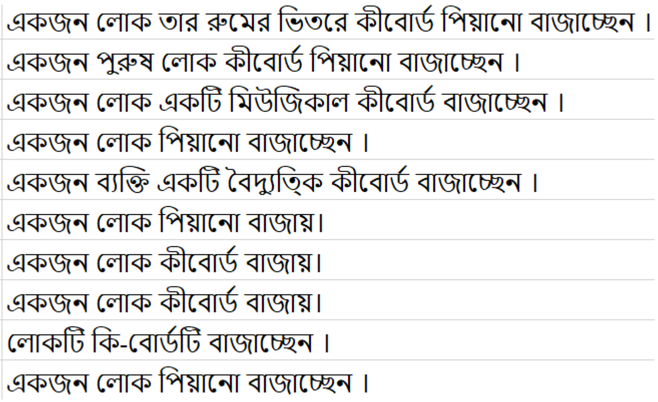
Figure 7.1: Architecture of Proposed Model.

Chapter 8

Experimental Setup

8.1 Dataset

Primarily, we have got determined to use Microsoft Video Description Corpus (MSVD), which is a popular dataset for video captioning. MSVD dataset clips is collected from YouTube and there are multiple sentences assigned for every video clip describing a simple event (including cooking, riding a motorcycle, and many others.). The dataset is produced from 1,970 YouTube clips. The clips had been annotated by using Amazon Mechanical Turk (AMT) workers. The audio is muted in all clips to avoid bias from lexical choices inside the captions. The duration of each video on this dataset is generally between 10 to 25 seconds particularly showing one activity.



একজন লোক তার রুমের ভিতরে কীবোর্ড পিয়ানো বাজাচ্ছেন ।
একজন পুরুষ লোক কীবোর্ড পিয়ানো বাজাচ্ছেন ।
একজন লোক একটি মিউজিকাল কীবোর্ড বাজাচ্ছেন ।
একজন লোক পিয়ানো বাজাচ্ছেন ।
একজন ব্যক্তি একটি বৈদ্যুতিক কীবোর্ড বাজাচ্ছেন ।
একজন লোক পিয়ানো বাজায়।
একজন লোক কীবোর্ড বাজায়।
একজন লোক কীবোর্ড বাজায়।
লোকটি কি-বোর্ডটি বাজাচ্ছেন ।
একজন লোক পিয়ানো বাজাচ্ছেন ।

Figure 8.1: Snapshot of Dataset.

We have translated the captions into Bengali by ourselves. For each video we have taken 10 captions.

8.2 Preprocessing of video and tokenization of captions

The MSVD dataset includes 1970 short video clips. To apply these videos in our framework we need to break them into frames as a video is nothing but a continuous motion of pictures. The common length of the films is 10 seconds. We divide the videos into frames on the rate of 3 frames per second. Then the two kinds of features are extracted by the VGG-19 and ResNext-101 model. All the features are saved as a vector and finally they are given input of the encoder.

For word tokenization we have used word2vec approach. The captions of the MSVD dataset have been tokenized on the way to break up words from spaces. This resulted in the vocabulary of all specific Bengali words inside the dataset. For the MSVD Bengali dataset, there had been 13,010 particular Bengali words. Accordingly, our vocabulary length changed to 13,010 words. The tokenized words are saved in a (.pkl) file.

8.3 Object and action detection

We have used VGG-19 model, which is strong to supply a wealthy representation of every sampled frame/clip from the video. The spatial features are just like static features. If we see in figure-1, we will see two players are playing with bat and ball. If we seize an image of the sure moment of playing we will see the bat, ball, players, bushes and many others. These all are static items.

But how can the model will predict that they may be playing? The answer is the movement of the ball, bat and the gamers. In a certain frame the ball is at a position. But in the subsequent frame the model sees that the pixels of the ball have shifted right or left. As the position of the ball, bat, gamers are shifting, so the model can expect that the temporal feature as 'playing'.

For doing this challenge we have selected ResNext-101 model. It can catch the difference between consecutive frames and predict the temporal feature. In this way the object is detected and the action is predicted.

8.4 Hyperparameter Selection

One major trouble of machine learning is overfitting. Overfit models have high variance. These models cannot generalize well. As an end result, this is a big hassle for video captioning. We located the performance of our model and noticed that it was laid low with overfitting rather than underfitting. To limit this overfitting problem a few hyperparameter tuning has been adapted in our model. We have taken the threshold = 3, which represents the minimum frequency of a word for inclusion in the dictionary. Next, the dimension of word embedding, attention linear layers, and decoder is 512. The learning rate is supposed to be 0.0002. Primarily, we have decided to run 200 epochs to train the model.

Chapter 9

Result Analysis

In this phase, we will show the progress that has made so far. First off, one of our largest successes is we have made our full dataset in Bengali which is fully human-annotated. In some models, the authors have used Google translator API to do the task of translation. But as Bengali is a complex language, so the translation that is made by way of this API isn't best, and it decreases the accuracy of the model. For this, we have made our complete dataset by ourselves and we agree with that it will provide us higher end result and accuracy. Next, we have implemented our key mechanism which is attention in the main model. We have run some epochs too. In the very beginning, the loss function and perplexity seems to be very big. For this we have changed some hyperparameter and some dimentions. After that we have got some better result. But still the BLUE-3 and BLUE-4 score are not so good.Different experimental setup and results are given below in the Table 9.1 :

Setup Number	Loss	Perplexity	BLEU-3	BLEU-4
Setup-1	11.6923	119641.6105	0.001644784	0.00517322
Setup-2	10.2072	55828.9588	0.00823	0.0062453
Setup-3	9.6451	40540.6597	0.0101	0.009265
Setup-4	7.4122	1656.1280	0.011068	0.01546

Chapter 10

Future Work

Within the result evaluation chapter, we have seen that our result is not up to the mark. We think the main reason behind this result is that, our model is not properly trained. Amongst 200 epochs only 2 or 3 were completed. In future, we will attempt to run all of the epochs and we hope it will deliver us a higher BLEU score. Next, as our dataset is completed so now we are able to apply it to our model. Our main focus will be on the decoder so that we get a better caption that is near to human-annotated caption.

Chapter 11

Conclusion

On this proposal, we have got attempted to find the research gaps in the area of video captioning in the Bengali language. We have also tried to provide information about the captioning gadget of video and its effect on society. We will try to fill up these research gaps in order that we are able to enhance its social fee. As we will try to use an attention-based version so we are hoping it is going to give us better accuracy than the previous models. Additionally, we would love to mention that the generated caption will be more grammatically accurate. Thus our version will generate captions for an input video.

Chapter 12

References

[1] C. Dai, X. Liu, L. Zhong and T. Yu. Video Based Action Recognition Using Spatial and Temporal Feature. In 2018 IEEE International Conference on Internet of Things (iThings).

[2] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, Trevor Darrell. Long-term Recurrent Convolutional Networks for Visual Recognition and Description. In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[3] L. Gao, Z. Guo, H. Zhang, X. Xu and H. T. Shen. Video Captioning With Attention-Based LSTM and Semantic Consistency. In Sept. 2017 on IEEE Transactions on Multimedia.

[4] Xiang Long, Chuang Gan and Gerard de Melo. Video Captioning with Multi-Faceted Attention. In 2018, on the journal of "Transactions of the Association fo Computational Linguistics".

[5] Vladimir Iashin and Esa Rahtu. A Better Use of Audio-Visual Cues: Dense Video Captioning with Bi-modal Transformer. In 17 May 2020, Accepted by BMVC.

[6] Sujin Lee and Incheol Kim. Multimodal Feature Learning for Video Captioning. In Hindawi Mathematical Problems in Engineering, 2018.

[7] Hwanhee Lee, Seunghyun Yoon, Franck Dernoncourt, Doo Soon Kim, Trung Bui and Kyomin Jung. ViLBERTScore: Evaluating Image Caption Using Vision-and-Language BERT. In conference of "Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems" in January 2020.

[8] Jacob Devlin and Ming-Wei Chang and Kenton Lee and Kristina. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Published in 19 May 2019 on “Association for Computational Linguistics”.

[9] R. Vedantam, C. L. Zitnick, and D. Parikh. 2015. Cider: Consensus-based image descrip-

tion evaluation. In IEEE CVPR.

[10] S. Banerjee and A. Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. ACL workshop on intrinsic and extrinsic evaluation measures for MT and/or summarization.

[11] K. Papineni, S. Roukos, T. Ward, and W. Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting on ACL.

[12] R. Vedantam, C. L. Zitnick, and D. Parikh. 2015. Cider: Consensus-based image description evaluation. In IEEE CVPR.

[13] J. S. Park, M. Rohrbach, T. Darrell, A. Rohrbach. Adversarial Inference for Multi-Sentence Video Description. In Computer Vision and Pattern Recognition (CVPR) 2019 IEEE Conference.

[14] Kidong Lee, Sung Yi, Soongkeun Hyun, Cheolhee Kim.Review on the Recent Welding Research with Application of CNN-Based Deep Learning Part I: Models and Applications.In Journal of Welding and Joining 2021.

[15] Olah, C., Understanding lstm networks. GITHUB blog, posted on August, 2015. 27: p. 2015.

Generated using Undegraduate Thesis LaTeX Template, Version 1.4. Department of Computer Science and Engineering, Ahsanullah University of Science and Technology, Dhaka, Bangladesh.

This thesis was generated on Sunday 27th June, 2021 at 5:41pm.