Mathematical Statistician and Engineering Applications
ISSN: 2094-0343
2326-9865

An Approach towards Video Captioning in Bengali

M. M. Rushadul Mannan[1], Mostafizur Rahman[2], Md. Shahir Zaoad[3], Md. Mahbubur Rahman[4], Angshu Bikash Mandol[5], Md. Adnanul Islam[6]

[1]Department of CSE
Military Institute of Science and Technology, Dhaka, Bangladesh
rushadmannan@gmail.com

[2]Department of CSE
Military Institute of Science and Technology, Dhaka, Bangladesh
akrahman76@gmail.com

[3]Department of CSE
Military Institute of Science and Technology, Dhaka, Bangladesh
shahir.glhd@gmail.com

[4]Department of CSE
Military Institute of Science and Technology, Dhaka, Bangladesh
mahbub@cse.mist.ac.bd

[5]Department of CSE
Military Institute of Science and Technology, Dhaka, Bangladesh
angshubmandol@gmail.com

[6]Department of CSE
Military Institute of Science and Technology, Dhaka, Bangladesh
adnanul@cse.mist.ac.bd

Article Info
Page Number: 01 - 14
Publication Issue:
Vol 71 No. 3s2 (2022)

Article History
Article Received*: 28 April 2022*
Revised*: 15 May 2022*
Accepted*: 20 June 2022*
Publication*: 21 July 2022*

Abstract
Video captioning refers to the process of predicting a semantically consistent textual description from a given video clip. Even though a significant amount of research work is present for video captioning in English, for Bengali the field of video captioning is nearly unexplored. Therefore, this research aims at generating Bengali captions that plausibly describe the gist of a specific short video. To accomplish this, Long Short-Term Memory (LSTM) based a sequence-to-sequence model is used that takes the video frame features as input and generates an analogous textual description. In this study, Microsoft Research Video Description Corpus (MSVD) dataset is used which is an English dataset. Therefore, a deep learning-based translator and manual labor are used to convert English captions into appropriate Bengali ones. Finally, the model's performance is evaluated using popular evaluation metrics - BLEU and TER. The proposed approach achieves BLEU and TER scores of 0.38 and 0.76 respectively, establishing a new benchmark for the Bengali video captioning tasks.

Keywords: - Bengali Video Captioning, Convolutional Neural Network, Recurrent Neural Network, Long Short-Term Memory.

I. INTRODUCTION

In Video captioning Computer Vision (CV) and Natural Language Processing (NLP) techniques are incorporated by extracting features from video frames concerning different modalities and then aggregating them spatially and temporally to produce a compact representation of what's happening over the video. With the rapid growth of video content making thanks to social platforms like YouTube and Facebook, thousands of hours of videos are uploaded to these social media every minute, which may need to be understood right away by local and global communities. The automatic generation of captions in order to describe scenes in images or videos can effectively confront this challenge. Video curation is another significant area where video captioning is much needed. Also, recent development around the accuracy improvement of video captioning and faster textual description generation has led to real-life applications like helping the deaf and hard of hearing people understand video content.

Video captioning is one of the most challenging fields of NLP because of multiple temporal objects, scenes, actions detection, and identifying salient contents. Despite such challenges, a few endeavors have been taken [1, 2], mainly inspired by recent advancements with LSTM. LSTM is basically an artificial recurrent neural network (RNN) architecture. Whereas conventional RNNs struggle to preserve information over many time steps and suffer from vanishing gradient issues, LSTM addresses these concerns by handling long-term dependencies and updating hidden conditions by integrating memory units [3].

Above mentioned works and a vast amount of the other works in the video captioning field are focused on English. There have also been works over Chinese, Hindi, and other languages [4, 5]. Although Bengali is one of the most spoken languages with 268 million speakers [6], the number of works done for the Bengali language is very limited. Several related studies have pursued image captioning in Bengali to extract information from still images and present that in the textual form [7-8]. To the best of our awareness, recently, only one work on generating video descriptions in the Bengali language has been done. The underlying reasons can be the lack of available Bengali datasets corresponding to the benchmark video captioning datasets for English or the complexity around video captioning in Bengali itself. However, video captioning in Bengali can be highly helpful in various application domains, particularly for the large Bengali-speaking community worldwide.

II. RELATED WORK

Video streams with high Spatio-temporal dependencies and complex video with multiple moving objects and actions make captioning very difficult. Despite this, various studies have proposed many methods that substantially encouraged research in video captioning and helped overcome these difficult situations.

The early works in video captioning were rule-based, built using predefined frameworks. The basic idea was to identify and extract critical features from video frames and represent them using a predefined sequence of events. Then these sequences of events may be translated into text using templates [9-10]. Nevertheless, these early works only drew on a limited domain of images and later were replaced by the enhanced deep neural network and the sequence-to-sequence model [11].

These shortcomings led to using the RNN model that constructs an encoder-decoder framework with CNN-based architecture. Here, a sequence of video frames is fed into a Convolution Network

[12] to extract features that are passed into a deep recurrent network to translate the video into the textual form [13].

Recent advancements around Different 2D and 3D CNN models were introduced for feature extraction and have successfully improved state-of-the-art representation learning [14, 15]. Nevertheless, feature aggregation for video captioning remains an open challenge. Several techniques from different perspectives have been studied for exploring feature aggregation in video captioning. To reduce exponential error accumulation for short-term memory issues, researchers proposed a video captioning approach which is based on adversarial learning along with LSTM. Also, an improvement of the same model, Bidirectional LSTM (Bi-LSTM) [16], was introduced, which profoundly captures the bidirectional global temporal structure of the video. For this purpose, an approach of joint visual modelling was devised which combines a forward LSTM pass and a backward LSTM pass with visual features from CNNs in order to encode video data.

Also, the Transformer model [17] opened new areas to significantly enhance the performance of video captioning models. The transformer model is a simple network architecture based solely on the attention mechanism. It is an encoder-decoder-based neural network on which the encoder side implements self-attention while encoder-decoder attention in addition to the self-attention on the decoder side. Different hybrid models are introduced, recombined with the transformer model, showing sophisticated performance. A notable addition in the NLP field is the Bidirectional Encoder based Transformers (BERT) model [18], a machine learning technique for NLP pre-training using Semi-Supervised Learning which incorporates the transformer mechanism. BERT is essentially an Encoder stack of transformer architecture, meant to condition both left and correct context in all stack layers to pre-train deep bidirectional representations from the unlabeled text.

Moreover, few recent studies have proposed different deep learning-based models concerning Bengali description generation. A study conducted by Khan et al. [8] has presented an end-to-end image captioning system using a multimodal architecture. A pre-trained ResNet-50 image encoder combined with a one-dimensional CNN is used to encode a sequence of information for extracting region-based visual features. This study was performed on the BanglaLekhaImageCaptions dataset consisting of 9,154 images. In another recent study [19], a Transformer based Encoder-Decoder Network combined with a pre-trained ResNet-101 CNN model or feature extraction was outlined for Bangla Image Caption Generation. It was done over the BanglaLekhaImageCaptions dataset. This approach outperformed all existing Bengali Image Captioning works and set a new benchmark over BLEU and METEOR.

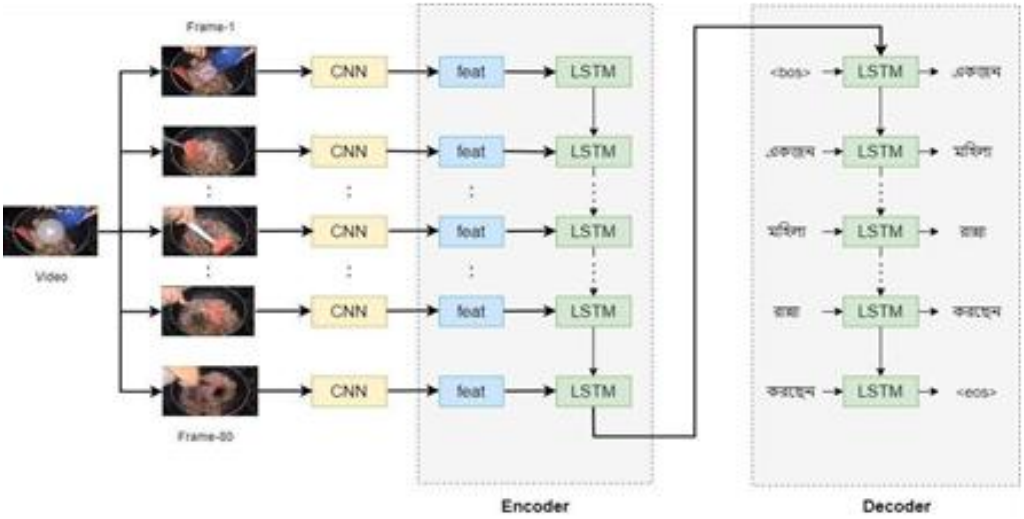

Fig. 1. Video Captioning using encoder-decoder LSTM

In the video captioning field, the Bengali language has seen too little work. The only work regarding Bengali video captioning was done by Raj et al. [20]. Here the proposed architecture is based on the encoder-decoder mechanism that combines various 2D and 3D-CNN along with BiLSTM as the encoder while the decoder comprises of two-layer LSTM. This paper had introduced a new road towards the Bengali video captioning field. They had trained and performed evaluation of the model on the converted MSVD dataset with the help of google translator API. When compared to the closely relatable Bengali image captioning works, they successfully accomplished the existing state-of-the-art result with 32.6 percent on BLEU and 51.2 percent on CIDEr.

III. Proposed Methodology

This research aims to design a model that generates Bengali captions from videos of short length where the video will be taken as input, and textual description will be developed as output. In this scenario, both input and output are sequences, as the video contains a sequence of frames while the caption is sequence of words. For this purpose, LSTM, a many-to-many sequence model, is used. Unlike many RNN, to avoid vanishing gradient problems, this network provides internal gates allowing the model to implement backpropagation through time successfully. It comprises two portions, encoder LSTM, and decoder LSTM. The encoder portion takes video frame features as *input*, and based on the encoder's result; the decoder generates textual descriptions (see "Fig. 1").

A. Word Embedding

It is done to represent a text in such a way so that the words that have the same meaning have an equal representation. This is required to bridge the human understanding of a text to a computer or machine. Therefore, each word is represented in a dictionary which is a vector collection of integers. We used the tokenizer class of Keras to create a vocabulary of 1500 in size. Individual words in this vocabulary are represented as real-valued vectors in this 1500-word-long vector space.

B. Feature Extraction

To extract features, each video is split into 80 frames irrespective of their length which are then passed to a pre-trained CNN model to extract the feature vector. VGG16, a deep convolutional network for large-scale image recognition, was adopted for this research. This model is quite advanced since it maintains 92.7 percent top-5 test accuracy in ImageNet which is a dataset

comprises of 14 million images. Before passing to the CNN model, frames are scaled to size 224x224. Almost all the videos use RGB color. Therefore, a tensor of (224x224x3) is used as input for the CNN model. After processing each frame, the model provides an output vector of 4096 values. As a result, a NumPy array of size 80x4096 is obtained for each video that contains the desired extracted features.

C. *Encoder*

The encoder LSTM is used to input the video frame features. Each frame is passed to one encoder cell which comprises an internal cell state 'c' and provides a hidden output state 's.' As each video is divided into 80 frames, therefore, the encoder architecture has 80 cells (time steps of the encoder), each of which takes an input vector of size 4096. In this research, we used an encoder with 512 hidden layers. As a result, each encoder cell maps these 4096 video frame features to a vector of size 512 in a step. As LSTM is a many-to-many sequence model, we are only concerned with the last state of the encoder, discarding all other outputs. Finally, the previous states (state-c and state-h) are sent to the decoder as their initial state.

D. *Decoder*

The decoder LSTM is used to output the textual description of the video. Using the word embedding and the last states of the encoder, the decoder predicts the words. Each word embedding is passed to a decoder cell as input. This work uses ten words as the maximum length for a sentence. Thus a total of ten decoder cells are used, making the time step for the decoder to ten. The first cell of the decoder takes the <bos> (beginning of a sentence) token as input along with the last state of the encoder and predicts the first word. The second cell takes the first-word embedding from the input caption and predicts the second word, then the third cell takes the second-word embedding from the caption and predicts the third word, and so on. The cycle continues until the <eos> (end of a sentence) token is reached.

E. *Optimization*

Neural networks have weights and biases which determine the performance of the model. However, these weights and biases cannot be calculated using an analytical method. Instead, they are calculated using an iterative optimization procedure. Adam optimization is used for this research, a different class to stochastic gradient descent. Unlike stochastic gradient descent, it uses an adaptive estimation of first-order and second-order moments.

IV. Experimental Setup

The experiment for this research is conducted in Google Colaboratory, a python development environment. Colab provides various pre-installed python and deep learning libraries. Some of the notable libraries used in this research are NumPy, TensorFlow, and Keras. Numpy is a python library that supports large multidimensional arrays and matrices. TensorFlow is an open-source library for various machine learning-related applications whereas Keras provides an interface for artificial neural networks accessible using python.

A. *Training*

The purpose of training is to make a model learn a mapping function. In our case, the mapping is between a video and a corresponding textual description. . To establish that, we used encoder-decoder LSTM. Both encoder and decoder have total hidden features of 512. The batch size is set to 320, i.e., 320 videos from the dataset will be trained at each batch. For training, another critical

hyperparameter is the epoch. A large number of epochs can cause overfitting of the model, while too few can lead to underfitting.

To avoid this problem, we used a callback approach known as early stopping. This enables defining an arbitrarily large number of epochs (100) and automatically stops the training when the performance stops improving. In this research, early stopping is implemented for five epochs, which will check five epochs before triggering early stopping. In addition, another callback is used when there is no change for a given number of training epochs, known as a plateau. In this situation, ReduceLROnPlateau is used, which adjusts the learning rate whenever a plateau is detected among 2 epochs. Finally, training with all these configurations, we got an accuracy of 0.7842 (see "Fig. 2").

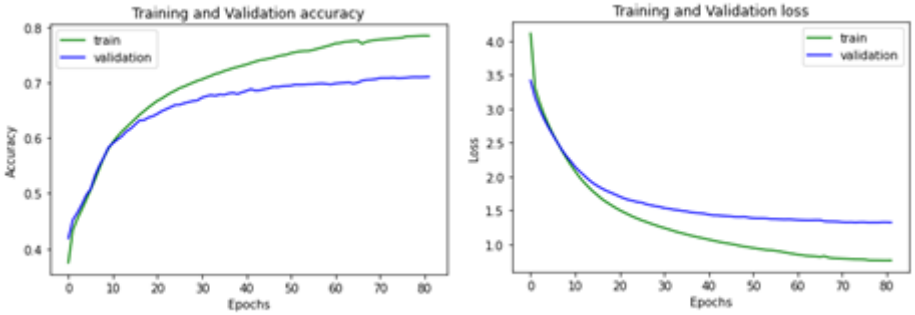

Fig. 2. Training and validation accuracy and loss

B. *Hyperparameter Selection*

Hyperparameters are configurations of the model that are assigned externally. Some significant hyperparameters are epoch, learning rate, batch size, etc. These parameters are crucial to the model's success. Firstly, to avoid the issue of overfitting and underfitting, we used early stopping on an arbitrarily large epoch. The model in this research was trained with various patience (5, 10, and 15) for two epoch sizes of 50 and 100. Patience determines the number of epochs it will check for early stopping. Another important hyperparameter is the learning rate as a large learning rate drives the model towards a suboptimal solution by quick convergence towards it, where a small one would bring the process to a standstill. We introduced our model on two different learning rates, .0003 and .00003, to find the optimal one. Furthermore, we trained with both a factor of 0.1 and 0.01 for ReduceLROnPlateau. ReduceLROnPlateau is a call-back used to reduce the learning rate by the specified factor value if the model stops improving. A summary of the combination of the hyperparameters and their respective training accuracy is presented in Table 1.

Figure 3 is the graphical representation of Table1. The figure provides a clear picture of the difference in accuracy based on mentioned hyperparameters where the best accuracy of 0.7842 is obtained for 100 Epochs, 0.0003 Learning Rate, and 0.1 ReduceLROnPleatue.

C. *Dataset*

To train the proposed video captioning model and evaluate it accordingly, a widely used dataset named MSVD is used. Video snippets from this dataset are divided into training categories and testing categories consisting of 1450 and 100 videos, respectively, with an 85% split ratio for training and validation purposes. The first outcome of our research is a semantically consistent Bengali video captioning dataset consisting of more than 32 thousand captions for 1450 video snippets altered from the original MSVD dataset. We converted the dataset into Bengali using a deep learning-based translator [21] and manual labor to achieve this goal. Some participants evaluated the

machine-translated captions to compare the consistency depending on human perception, ensuring the dataset's relevance with the Bengali language culture as closely as possible.

TABLE I. EFFECTIVENESS OF HYPERPARAMETERS ON TRAINING ACCURACY

Epochs	Learning Rate	Early Stopping	Reduce-LROn-Pleatue	Accuracy	Val-Accuracy
100	0.0003	5	0.1	0.7842	0.6912
100	0.00003	15	0.1	0.6327	0.6149
50	0.0003	5	0.1	0.7527	0.6990
50	0.00003	15	0.1	0.6582	0.6279
100	0.0003	10	0.01	0.7701	0.7086
50	0.0003	10	0.01	0.7517	0.6988

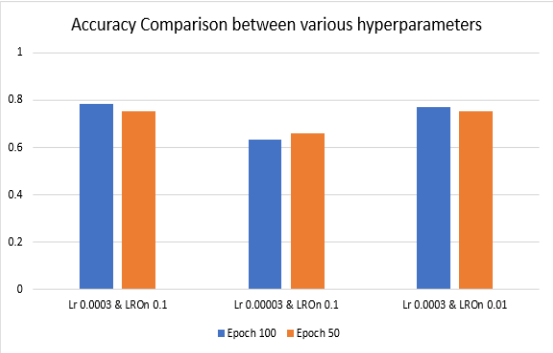

Fig. 3. Comparison of accuracy between various combinations of significant hyper parameters

V. RESULT AND EVALUATION

This section summarizes our results over testing and evaluating our proposed video captioning model.

A. *Quantitative Analysis*

When evaluating video captioning models, some consistent evaluation protocols are used. Comparing results from different approaches can be very enlightening for assessing the performance of the proposed model; Since numerous evaluation metrics exist around that have distinct methods to evaluate: BLEU, TER.

We evaluated the candidate captions generated from our model with the given set of reference captions for video clips to check the quality against some fixed standards. In this research, BLEU is used as one of the evaluating metrics. The study [22] found that BLEU performs well for corpus-level comparisons over which many n-gram matches exist. However, the n-gram match rarely occurs at the sentence level, specifically when the value of n is higher (e.g., n=4). We used BLEU-1 (uni-gram), BLEU-2 (bi-gram), BLEU-3 (tri-gram), and BLEU-4 (four-gram) to evaluate the proposed model and compare the performance over the sentence length. Among all these, BLEU-4 is considered the most appreciable BLEU metric for the qualitative analysis part of this study.

In order to assess the proposed model from a different outlook Translation Error Rate (TER) [23] is used, which evaluates the performance by giving the amount of work that a human needs to cover for matching the system output precisely to the reference translation. In terms of human perceptions of MT quality, the single-reference variant of TER coincides with the four-reference variant of

BLEU. Compared to BLEU, TER focuses on achieving higher correlations with human assessments than n-gram-based methods by assigning lower costs to phrasal shifts. In Table 2 the results of this study are presented, based on the altered MSVD dataset, using the evaluation metrics mentioned earlier.

TABLE II. PERFORMANCE SCORES OF PROPOSED MODEL USING DIFFERENT SEARCH TECHNIQUES

Evaluation Metrics	Beam Search	Greedy Search
BLEU-1	0.635	0.650
BLEU-2	0.632	0.568
BLEU-3	0.479	0.426
BLEU-4	0.382	0.3486
TER	0.76	0.84

Fig. 4-1.

Fig. 4-2.

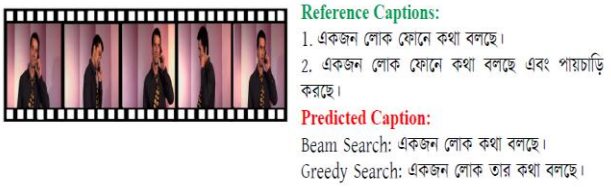

Fig. 4-3.

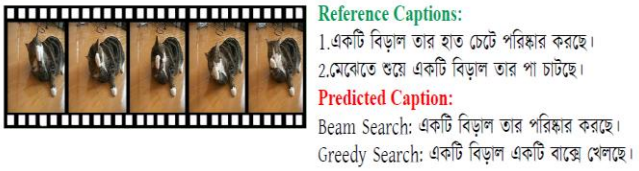

Fig. 4-4.

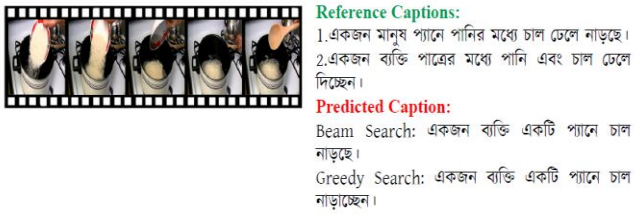

Fig. 4-5.

Fig. 4. Sample captions generated by our approach along with corresponding reference captions

B. Qualitative Analysis

The sample outputs of our model in terms of the generated caption's quality can be visualized for five videos tested in real-time.

This evaluation process is pursued with 30 participants, providing a single reference caption for each video. A subset (i.e., most appropriate) of those reference captions is selected for evaluation purposes. Example reference captions for each video are shown in Figure 4. A comprehensive qualitative analysis of the predicted captions against reference captions in terms of evaluation metrics for the tested videos is represented in "Fig. 5".

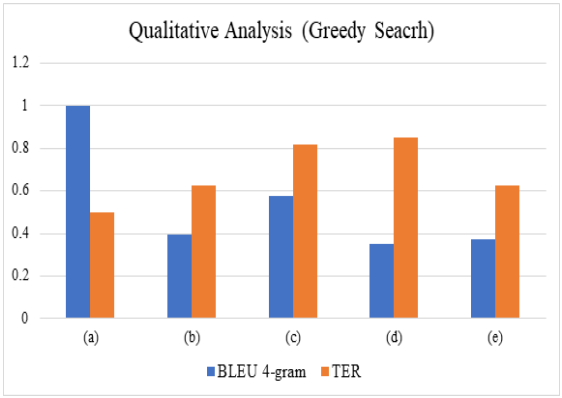

Fig. 5-1. Using Greedy Search

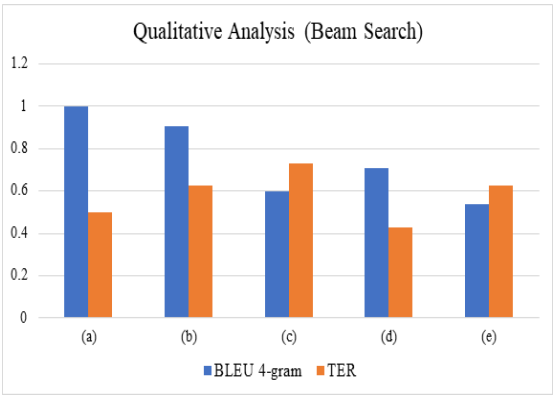

Fig. 5-2. Using Beam search

Fig. 5. Qualitative analysis of the performance Achieved by the proposed approach in terms of BLEU and TER

In “Fig. 5-1”, using greedy search, the BLEU-4 gram has a lower score while TER has higher scores representing the poor performance of the greedy search. “Fig. 5-2” illustrates the evaluation of the beam search approach using BLEU 4-gram and TER, where it is seen that the BLEU score is higher than TER, which indicates the effectiveness of beam search.

For generating the caption in Bengali, the most likely output sequences have to be decoded by searching all possible output sequences. Two different searching algorithms were used for this purpose, beam search and greedy search. The greedy search uses the local optimality to select a word for each stage to generate the caption. In contrast, the beam search algorithm, a heuristic search method, selects k possible alternatives at each location for decoding an input sequence. “Fig. 6” presents a comparative analysis of the above-mentioned tested videos using these searching methods.

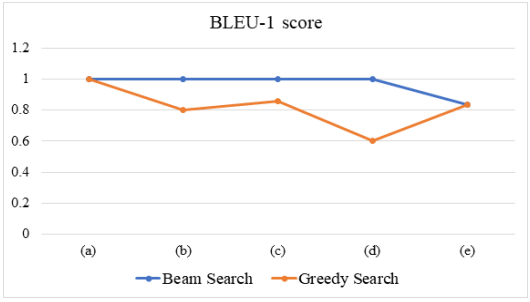

Fig. 6-1

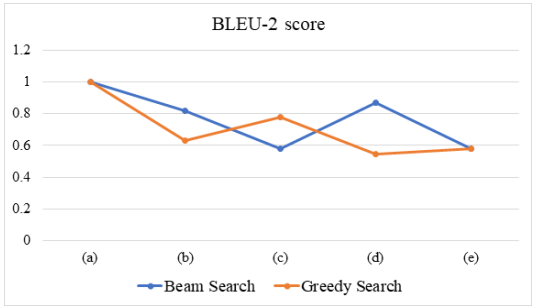

Fig. 6-2

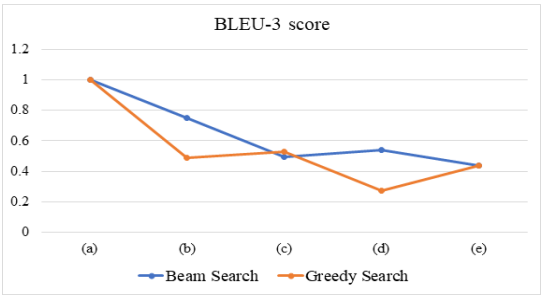

Fig. 6-3

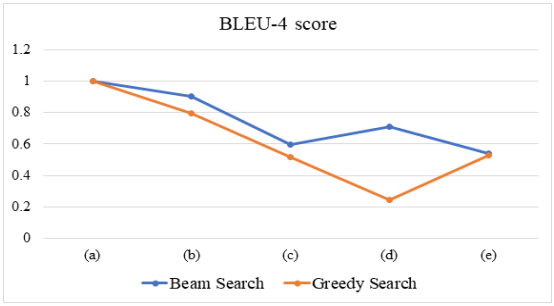

Fig. 6-4

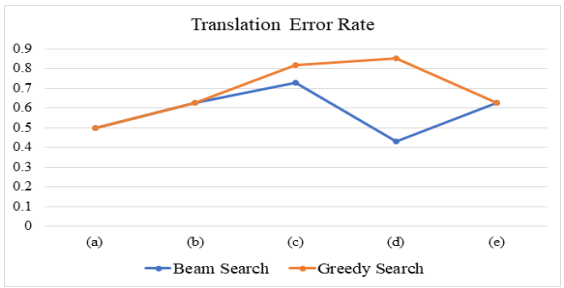

Fig. 6-5

Fig. 6. Comparative analysis between greedy and beam search techniques for captioning the five test videos in terms of different evaluation metrics - (1) BLEU-1, (2) BLEU-2, (3) BLEU-3, (4) BLEU-4, and (5) TER.

The figure reflects that the beam search (Blue Curve) performs better than the greedy search (Orange Curve) as the bleu curve overtakes the orange curve for BLEU scores (n=1 to 4), and the reverse happens for TER score. We find that the error rate in TER for our model is slightly higher than expected. This may be attributed to the fact that we have used several reference captions in BLEU measures while TER is evaluated using only one reference caption.

Finally, Table 3 presents a comparison between our proposed approach with the only other existing model [20] for video captioning in Bengali. This illustrates the proposed model's higher performance for both BLEU-3 and BLEU-4 scores. In our research along with a deep-learning-based translator, manual labor was incorporated in order to translate the English captions from the MSVD dataset into Bengali. This led to the better performance of the proposed model.

TABLE III. PERFORMANCE COMPARISON WITH EXISTING VIDEO CAPTIONING MODEL WITH PROPOSED MODEL

Model	**BLEU-3**	**BLEU-4**
Raj et al. [20]	0.432	0.326
Proposed Model	**0.479 (11% ↑)**	**0.382 (17% ↑)**

VI. CONCLUSION AND FUTURE WORK

This paper has developed a dataset for Bengali video captioning from the MSVD dataset. We employed a pre-trained CNN model, specifically VGG16, for feature extraction and then used our prepared dataset to train the LSTM model. The soundness of LSTM layers in the case of sequential

data made it possible for the generated captions to be more natural. Finally, a comparative analysis is performed on the model's performance for two different search techniques in terms of two popular evaluation metrics - BLEU and TER with a score of 0.38 and 0.76 respectively.

The proposed system is basically developed using CNN and LSTM architecture. In the future, the system's performance can be further improved by exploring other RNN-based architectures or the BERT model [24-26]. A comprehensive comparative analysis is yet to be performed among different machine learning models for video captioning in Bengali [26]. Moreover, to enhance the use of automatic video captioning in real life, we plan to convert the generated captions to audio format using NLP techniques so that it can serve blind people in various purposes (e.g., education, navigation guidance, etc.), which we leave as a potential future work.

REFERENCES

[1] L. Gao, Z. Guo, H. Zhang, X. Xu, and H. T. Shen, "Video Captioning With Attention-Based LSTM and Semantic Consistency," IEEE Transactions on Multimedia, vol. 19, no. 9, pp. 2045–2055, Sep. 2017, doi: 10.1109/tmm.2017.2729019.

[2] X. Li, Z. Zhou, L. Chen, and L. Gao, "Residual attention-based LSTM for video captioning," World Wide Web, vol. 22, no. 2, pp. 621–636, Feb. 2018, doi: 10.1007/s11280-018-0531-z.

[3] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[4] A. Singh, T. D. Singh, and S. Bandyopadhyay, "Attention based video captioning framework for Hindi," Multimedia Systems, vol. 28, no. 1, pp. 195–207, Jun. 2021, doi: 10.1007/s00530-021-00816-3.

[5] K. Lin, Z. Gan, and L. Wang, "Multi-modal Feature Fusion with Feature Attention for VATEX Captioning Challenge 2020," arXiv:2006.03315 [cs, eess], Jun. 2020, Accessed: Mar. 15, 2022. [Online]. Available: https://arxiv.org/abs/2006.03315.

[6] M. Szmigiera, "Most spoken languages in the world | Statista," Statista, Mar.30, 2021. https://www.statista.com/statistics/266808/the-most-spoken-languages-worldwide/

[7] Md. A. Jishan, K. R. Mahmud, A. K. A. Azad, M. R. A. Rashid, B. Paul, and Md. S. Alam, "Bangla language textual image description by hybrid neural network model," Indonesian Journal of Electrical Engineering and Computer Science, vol. 21, no. 2, p. 757, Feb. 2021, doi: 10.11591/ijeecs.v21.i2.pp757-767.

[8] M. Faiyaz Khan, S. M. Sadiq-Ur-Rahman, and M. Saiful Islam, "Improved Bengali image captioning via deep convolutional neural network based encoder-decoder model," in Algorithms for Intelligent Systems, Singapore: Springer Singapore, 2021, pp. 217–229.

[9] A. Kojima, T. Tamura, and K. Fukunaga. Natural Language Description of Human Activities from Video Images Based on Concept Hierarchy of Actions. International Journal of Computer Vision, vol. 50, no. 2, pp. 171–184, 2002, doi: 10.1023/a:1020346032608.

[10] P. Das, C. Xu, R. F. Doell, and J. J. Corso, "A thousand frames in just a few words: Lingual description of videos through latent topics and sparse object stitching," in 2013 IEEE Conference on Computer Vision and Pattern Recognition, 2013.

[11] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to Sequence Learning with Neural Networks," in Advances in Neural Information Processing Systems, 2014, vol. 27. [Online]. Available: https://proceedings.neurips.cc/paper/2014/file/a14ac55a4f27472c5d894ec1c3c743d2-Paper.pdf

[12] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[13] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko, "Translating videos to natural language using deep recurrent neural networks," in Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2015.

[14] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv [cs.CV], 2014.

[15] S. Xie, R. Girshick, P. Dollar, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[16] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," IEEE Transactions on Signal Processing, vol. 45, no. 11, pp. 2673–2681, 1997, doi: 10.1109/78.650093.

[17] A. Vaswani et al., "Attention is All you Need," Advances in Neural Information Processing Systems, vol. 30, pp. 5998–6008, 2017, Accessed: Nov.25, 2020. [Online].Available: https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

[18] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional Transformers for language understanding," arXiv [cs.CL], 2018.

[19] M. A. H. Palash, M. D. A. A. Nasim, S. Saha, F. Afrin, R. Mallik, and S. Samiappan, "Bangla image caption generation through CNN-transformer based encoder-decoder network," arXiv [cs.CV], 2021.

[20] A. H. Raj, A. Seum, A. Dash, S. Islam, and F. M. Shah, "Deep learning based video captioning in Bengali," in 2021 26th International Conference on Automation and Computing (ICAC), 2021.

[21] Nidhaloff, "How to translate text with python," Analytics Vidhya, Oct.17,2020.https://medium.com/analytics-vidhya/how-to-translate-text-with-python-9d203139dcf5 (accessed Mar. 16, 2022).

[22] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in Proceedings of the 40th Annual Meeting on Association for Computational Linguistics - ACL'02, 2001.

[23] M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul, "A study of translation edit rate with targeted human annotation," in Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers, 2006, pp. 223–231.

[24] M.A. Islam, M.S.H. Anik, and A.B.M.A.A. Islam, “Towards achieving a delicate blending between rule-based translator and neural machine translator”, Neural Computing and Applications, vol. 33, no. 18, pp. 12141–12167, 2021. https://doi.org/10.1007/s00521-021-05895-x

[25] Ajitha, P.Sivasangari, A.Gomathi, R.M.Indira, K."Prediction of customer plan using churn analysis for telecom industry",Recent Advances in Computer Science and Communications,Volume 13, Issue 5, 2020, Pages 926-929.

[26] "Sivasangari A, Ajitha P, Rajkumar and Poonguzhali,” Emotion recognition system for autism disordered people”, Journal of Ambient Intelligence and Humanized Computing (2019)."

[27] Ajitha, P., Lavanya Chowdary, J., Joshika, K., Sivasangari, A., Gomathi, R.M., "Third Vision for Women Using Deep Learning Techniques", 4th International Conference on Computer, Communication and Signal Processing, ICCCSP 2020, 2020, 9315196

[28] Sivasangari, A., Gomathi, R.M., Ajitha, P., Anandhi (2020), Data fusion in smart transport using convolutional neural network”, Journal of Green Engineering, 2020, 10(10), pp. 8512–8523.

[29] A Sivasangari, P Ajitha, RM Gomathi, "Light weight security scheme in wireless body area sensor network using logistic chaotic scheme", International Journal of Networking and Virtual Organisations, 22(4), PP.433-444, 2020

[30] Sivasangari A, Bhowal S, Subhashini R "Secure encryption in wireless body sensor networks",Advances in Intelligent Systems and Computing, 2019, 814, pp. 679–686

[31] Sindhu K, Subhashini R, Gowri S, Vimali JS, "A Women Safety Portable Hidden camera detector and jammer", Proceedings of the 3rd International Conference on Communication and Electronics Systems, ICCES 2018, 2018, pp. 1187–1189, 8724066.

[32] Gowri, S., and J. Jabez. "Novel Methodology of Data Management in Ad Hoc Network Formulated Using Nanosensors for Detection of Industrial Pollutants." In International Conference on Computational Intelligence, Communications, and Business Analytics, pp. 206-216. Springer, Singapore, 2017.

[33] Gowri, S. and Divya, G., 2015, February. Automation of garden tools monitored using mobile application. In International Confernce on Innovation Information in Computing Technologies (pp. 1-6). IEEE.

[34] M.S.H. Mukta, M.A. Islam, F. Khan, A. Hossain, S. Razik, S. Hossain, and J. Mahmud, “A Comprehensive Guideline for Bengali Sentiment Annotation”, ACM Trans. Asian Low-Resour. Lang. Inf. Process, vol. 21, no. 2, Article 30, pp. 1-19, 2022. doi: https://doi.org/10.1145/3474363

[35] S. Sakiba, M.M.U. Shuvo, N. Hossain, S.K. Das, J.D. Mela, M.A. Islam, “A Memory-Efficient Tool for Bengali Parts of Speech Tagging”, In: Hemanth, D., Vadivu, G., Sangeetha, M., Balas, V. (eds) Artificial Intelligence Techniques for Advanced Computing Applications. Lecture Notes in Networks and Systems, vol. 130. Springer, Singapore, 2021. https://doi.org/10.1007/978-981-15-5329-5_8