Deep Learning based, a New Model for Video Captioning

Elif Güşta Özer[1], İlteber Nur Karapınar[2], Sena Başbuğ[3], Sümeyye Turan[4], Anıl Utku[5], M. Ali Akcayol[6]
Department of Computer Engineering, Faculty of Engineering
Gazi University, Ankara, Turkey

***Abstract*—Visually impaired individuals face many difficulties in their daily lives. In this study, a video captioning system has been developed for visually impaired individuals to analyze the events through real-time images and express them in meaningful sentences. It is aimed to better understand the problems experienced by visually impaired individuals in their daily lives. For this reason, the opinions and suggestions of the disabled individuals within the Altınokta Blind Association (Turkish organization of blind people) have been collected to produce more realistic solutions to their problems. In this study, MSVD which consists of 1970 YouTube clips has been used as training dataset. First, all clips have been muted so that the sounds of the clips have not been used in the sentence extraction process. The CNN and LSTM architectures have been used to create sentence and experimental results have been compared using BLEU 4, ROUGE-L and CIDEr and METEOR.**

***Keywords*—Video captioning; CNN; LSTM**

I. Introduction

In order to facilitate the lives of visually impaired individuals, the new technologies have been developed at last decade. These technologies can help people who are having trouble with their vision ability. About 1.3 billion people have been lived with this problem, in 2018 [1]. The technologies can take these people where they want to go, or they can help them read the texts. In this study, it has been aimed that the visually impaired individuals can perceive the events in their environment. Also, the event analysis over video should be worked in real time. The images detected from the camera are divided into frames in real time. By analyzing these images, the objects are recognized then the differences between the frames are detected and a prediction has been done for the actual action. Then, the basic event in the perceived real-time image is converted into a sentence and finally converted to sound. Thus, visually impaired individuals can be able to learn what is happening around them without the help of another person. Especially individuals who have lost their vision later can easily understand and visualize the descriptions.

In the literature, it is possible to examine the video subtitle creation methods that proposed in this context in two groups. The first of these is generative based methods and the second is retrieval-based methods such as Recurrent Neural Network (RNN). The basic approach of the first group of methods is object recognition on a visual content and the creation of subtitle with natural language creation techniques [2, 3]. The methods in the second group, unlike the first group, utilize from the visual similarities of the visual contents on the data set and the textual similarities of the subtitles, at the same time and select the most likely subtitle for the images from the appropriate subtitles [4, 5].

Yao et al., have developed an automatic video description system [4]. A directory has been created to improve search quality in online videos and to enable visually impaired people to use video definition. Youtube2Text and the DVS dataset have been used as datasets. From the deep learning techniques, combination of RNN and ConvNet models and GoogleNet as the architecture have been utilized. The results have been tested by applying Long Short Term Memory Network (LSTM) model which is a RNN type without soft-attention mechanism and no mechanism. After that the reason for using the soft-attention mechanism, has not been to include in the definition of less important actions and objects in the clip. Experimental results showed that the soft-attention mechanism improves the identification performance.

Venugopalan et al., have developed Sequence-to-Sequence Video to Text (S2VT) model to make description one of the videos there [5]. They implemented the developed model using Microsoft Video Description (MSVD) dataset, MPII Movie Description dataset (MPII-MD) and Montreal Video Annotation Dataset (MVAD) dataset. As the method, LSTM has been used in actualized study. Also, in this study no pooling method has been used. In the proposed model, the successive (sequence to sequence) video frames have been taken as an input and as the output, it has been given successive words. LSTM resolves the video, frame by frame. To do this, convolutional neural network (CNN) output which applied on every frame taken as an input. After reading all frames, the model has been created a sentence from the words. AlexNet and VGG-16 have been used as architecture in this study.

Li et al., have developed an architecture called Residual Attention-based LSTM (Res-ATT) [6]. To describe the video in detail, the mechanism called temporal attention with CNN and LSTM has been applied by them. This mechanism has been used to better identify the important events in the video. They also used a technique they called Residual to avoid the degradation problem when using RNN. MSVD and MSR-VTT datasets have been used to test the architectures which they developed and as the metric, BLUE, METEOR, CIDEr have been used in this study. Also, Microsoft COCO caption evaluation has been used as an evaluation code.

Rohrbach, et al., have studied on HD films [7]. The MPII-MD dataset has been used for this study. AD (Audience Descriptions), defines films for blind or visually impaired people. Multiple sentence definitions and long videos presents in TACoS Multi-Level and YouCook datasets on this topic. However, for shorter term videos, dataset options are increasing. Apart from the definition with AD, the study may also have tasks such as creating a story from relationships in the film and analyzing relationships.

Xu et al., have looked at video captioning from a different perspective [8]. Videos has been modelled as a sequence of frames. The Attentive Multi-Grained Encoder (AMGE) model for the encoder phase has been used.

Krishna et al., have defined the detected events simultaneously with natural language. Using developed model, all events have been identified in a single transition of the video [9]. ActivityNet Captions, has been selected as dataset. A hierarchical RNN structure has been used to provide more detailed events in the videos. Semantic information in videos has been taken as input. This information has been given in the form of an Array structure. Then, LSTM fed. However, there has been time differences between events because detailed events have been described. Events have this time difference have been handled separately. BLEU, METEOR and CIDEr have been used as metric.

Wu et al., have focused on the video classification problem [10]. Short-term events occur on videos and a hybrid model has been created because of these short-term events in the study. Two different feature extraction methods have been used for a given input video: Spatial CNN and Short-term stacked motion optical flows. Extracted features and LSTM models have been fed separately. The results have been combined to make the final prediction. The UCF-101 Human Actions and the Columbia Consumer Videos (CCV) have been selected as the dataset of the study.

Yue-Hei Ng et al., have proposed two deep neural network architectures for combining long-time videos' information [11]. Models has been used to understand the hidden events existing in image in every stream. Various convolutional temporal feature pooling architectures have been tried in the first proposed architecture. They have used LSTM cells connected to the output of the CNN in the second proposed architecture. Performance of the proposed architecture for Sports 1 million dataset is 73.1% and UCF-101 dataset is 88.6%.

Wang et al., have presented a new model that combines audio and visual cues called HACA (Hierarchically Aligned Cross-modal Attentive) network [12]. The proposed new HACA model learns and aligns both global and local contexts between different forms of video. In this study, hierarchical encoder-decoder network including visual attention, audio attention, and decoder attention have been used by them. CNN models that was pre-trained has been used to extract visual features and audio features. For image classification they used the ResNet model and for audio classification. They used the VGGish model. Besides, MSR-VTT has been used for the model training.

In this study, sequence-to-sequence (Seq2Seq) models have been used. The VGG-16 and VGG-19 CNN architectures are used with the LSTM. Developed model has been trained on video to text pairs. In this study, it is aimed that the developed model has been learned to associate a variable-sized square array with a variable-sized word array. The performance of developed model has been evaluated on the Microsoft Video Description Corpus (MSVD) and the BLUE, ROUGE, CIDEr and METEOR metrics.

II. Video Captioning

Video captioning is a popular research field for computer vision, image processing and natural language processing. In video captioning, it is aimed to automatically obtain a natural sentence from a video. However, automatically creating natural language definitions of videos is a challenge for machines. Automatic video description model should be able to express objects and events presented in the video. Automatic video description model also explains their relationships with each other in a natural sentence.

Fig. 1 shows differences between video tagging, image captioning and video captioning. The video tag is the name of a extraction of particular object or event in the video. Image (frame) captioning is automatically generating a single sentence or multiple sentence that define an image. Video captioning should also capture the causality between events, actions and objects, as well as the speed and direction of the objects involved [13].

Video caption generation can be classified in two main categories. Template-based models depend on specific grammar rules. Sequence learning-based model learn the probability distribution of visual content, and create new sentences with syntactic structure. These operations explore general (Seq2Seq) models for the generation of video captions. The sequence learning is shown in Fig. 2. Given an input video, 2D and/ or 3D CNN are used to extract visual characteristics in raw video frames. Mean pooling or soft attention operations are performed on visual features. Then, LSTM is trained.

BLEU [14], ROUGE [15] and CIDEr [16] metrics are used for evaluation of the video captioning task. BLEU is based on precision and only controls the n-gram matches in the estimated and basic references. ROUGE has different n-gram versions and calculates recall. CIDEr measures Term Frequency Inverse Document Frequency (TF-IDF) calculating for each n-gram. The performance of our models are evaluated using these 3 important metrics.

By the increasing interest of the video captioning, several large datasets have been released. The YouTube cooking video dataset (YouCook), contains videos about scenes where people cook various recipes [17]. Similarly, TACoS-ML is mainly built based on MPII Cooking Activities dataset and contains cooking videos and descriptions [18]. M-VAD is composed of about 49,000 DVD movie snippets extracted from 92 DVD movies [19]. MPII-MD is another collection of movie descriptions dataset that is similar to M-VAD. It contains around 68,000 movie snippets from 94 Hollywood movies [20]. We perform all our experiments on the MSVD dataset. Microsoft Video Description (MSVD) dataset [21] is a collection of 1,970 YouTube snippets with human annotated sentences. Comparisons of video captioning datasets have shown in Table I.

TABLE. I. COMPARISON OF VIDEO CAPTIONING DATASETS

Dataset	Context	Source	Video	Clip	Sentence	Word	Duration (hrs)
YouCook	cooking	labeled	88	-	2668	42457	2.3
TACos	cooking	AMT workers	123	7206	18227		-
M-VAD	movie	DVS	92	48986	55905	519933	84.6
MPII-MD	movie	DVS+Script	94	68337	68375	653467	73.6
MSVD	multi-category	AMT workers	-	1970	70028	607339	5.3

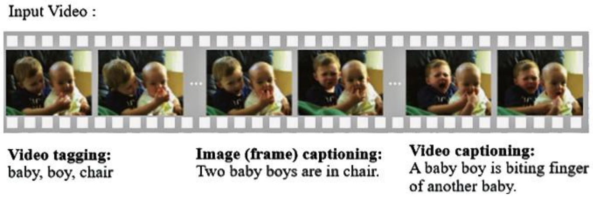

Input Video :
Video tagging:
baby, boy, chair
Image (frame) captioning:
Two baby boys are in chair.
Video captioning:
A baby boy is biting finger of another baby.

Fig. 1. Video Tagging, Image Captioning and Video Captioning.

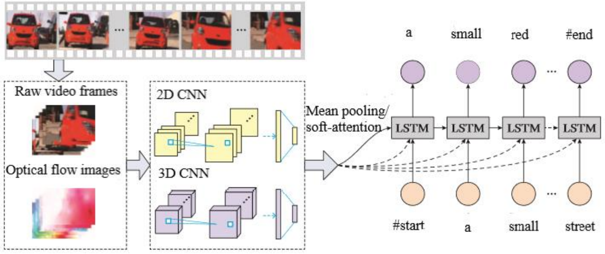


Fig. 2. A Common Architecture with Sequence Learning for Video Captioning.

III. DEEP LEARNING

For years, human being propensity to manage their world more easily by modeling on computers. To overcome this, algorithms that can learn to construct context by themselves have been developed by researchers. Artificial Intelligence (AI) concept has emerged in this development process. AI is a sub-branch of machine learning and uses many non-linear layers for feature extraction.

Neural networks make a trained prediction based on categories and analysis. A machine learning system makes this prediction based on the greatest possibilities. Many of them, learn from their mistakes and this makes them a more accurate system.

The deep learning process is based on learning from data. Computational models consisting of multiple processing layers are used to have a good predictive system in deep learning. The complex structure in complex data sets is discovered using the back propagation algorithm. In this way, more complex concepts are learned using the created hierarchy of concepts. The architectures used in the project are CNN and LSTM.

A. *CNN*

CNN is an important structure for object detection and classification. The major advantage of CNN is that important features can be detected automatically without any human supervision. To illustrate, when given pictures that are many cats and dogs, the class's unique characteristics are learned for each class.

To give an example structure for the CNN architecture as shown in Fig. 3: first, it takes the regions of the input image one by one under the name of receptive field. Convolution process and pooling are performed on the incoming input in CNN. Feature maps are generated as a result of the convolution layer which is the main block of CNN. Pooling, on the other hand, shortens training time and reduces size to combat overfitting. Property maps are sent to fully connected layer input by making flatten. The classification process is performed on fully connected layers and outputs. The hyper parameters such as the number and convolutional and pooling layers, the number of fully connected layers and activation functions have been determined before training phase.

B. *LSTM*

The inefficient process of the RNN architecture, which works with backward dependence due to the gradient problem that called vanishing gradient problem, caused this architecture to remain in the background. This problem has been solved by LSTM. LSTM is an effective model for capturing long-term temporal dependencies. It is particularly preferred for speech and text processing.

An LSTM layer consists of a series of blocks that are repeatedly connected, known as memory blocks. Each LSTM layer includes one or more repetitive connected memory cells and input, output and forget gates as shown in Fig. 4. The outputs of the memory cells are connected to all the gates and the cell itself. The gates are optionally a means of transmitting information, and they comprise a layer of sigmoidal neural network and a dot multiplication process. In the sigmoid layer, how much of each component must pass is defined between zero and one. The value of zero means “don’t let anything to pass”, whereas the value of one means “let everything to pass”. The cell condition is like a kind of conveyor belt. Additionally, LSTM has learning rate, hidden layer size and unit parameters. Units are the memory part of LSTM. Increasing the number of units is a widely used option to achieve a powerful model. On the other hand, the training time takes longer time with more unit. This means that the model has learned more.

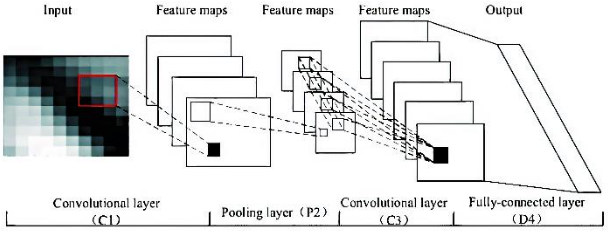


Fig. 3. CNN Structure as an Example.

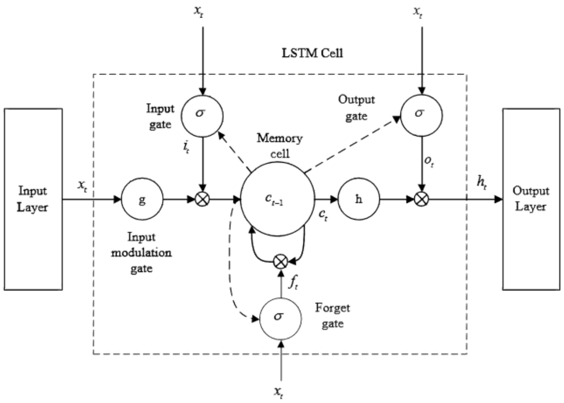


Fig. 4. LSTM Gates.

LSTM, which is used as a decoder within the project, uses feature vectors from CNN for word production and combines language knowledge. LSTM can do the sentence building process because of it can store previously defined objects in its memory. In the word generation phase, the next word is produced using current state and past states. Word generation is continued until end of sentence token is received.

IV. Developed Model

In this study, it has been aimed to develop a system that can accurately express the events in the videos with subtitles. In this section, the structure and steps of the study has been explained. Then, the experimental results of each architecture used in the study have been analyzed.

Fig. 5 illustrates the architecture of the system. The structure of the project consists of feature extraction, learning and prediction steps. In the developed system, videos have been given as input to the system. The videos provided to the system have been converted into frames before being exported to the CNN model to be used and feature extraction was made from these created frames. Caffe has been used for feature extraction in the project.

In the system, Caffe is loaded before feature extraction starts. The models included in Caffe are specially trained models with ImageNet for object detection and UCF101 (Fig. 6) for motion classification. Frames in selected intervals have been selected from the videos. Frames are a series of images. This sequence of images creates the representation of the video. This resized representation sequence has been sent to the selected CNN model as input data. Objects and movements in representations have been determined. Specified properties have been added as attributes to a numpy file. The Caffe has some pre-trained models. In the project, VGG16, VGG19 and HybridCNN models have been used and the results of these models have been examined.

Then, the LSTMs have been used for the second step, the learning process, and the third step, to generate explanations of different lengths. In this step, two LSTM layers have been used as Encoder and Decoder. Using LSTM units in different numbers, the effect on training has been examined. The first layer of LSTM has been used for video processing; the other layer has been used to learn the sentence structure.

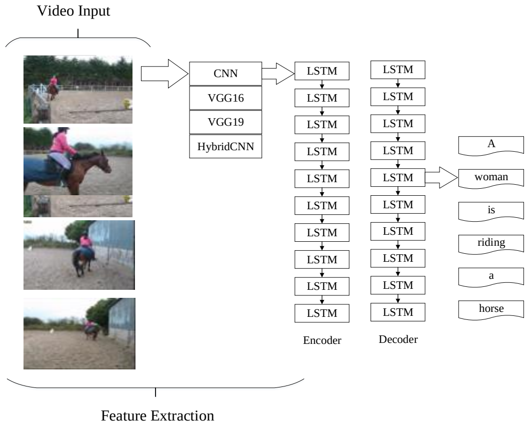


Fig. 5. Implemented Application Model.

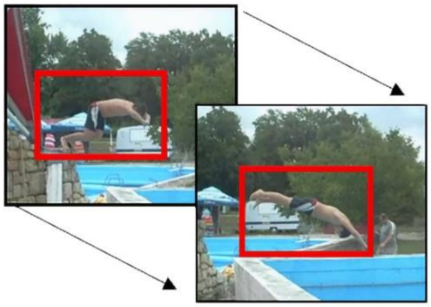
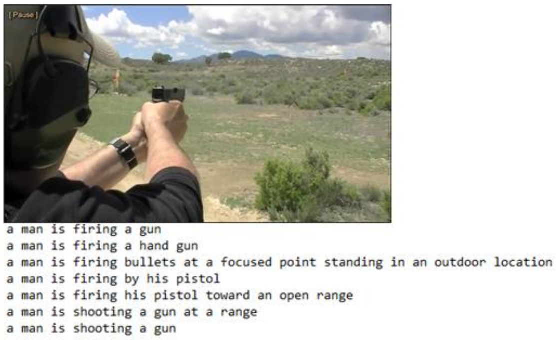


Fig. 6. Detect Motion with UCF101.



```
a man is firing a gun
a man is firing a hand gun
a man is firing bullets at a focused point standing in an outdoor location
a man is firing by his pistol
a man is firing his pistol toward an open range
a man is shooting a gun at a range
a man is shooting a gun
```

Fig. 7. Example of Reference Sentences in the MSVD Dataset.

The properties specified for each video in the feature extraction have been processed in the encoder LSTM layer, creating hidden representation and feeding each other. The purpose of hidden representation is to teach the algorithm its own feature engineering and to make the process more reliable. After this stage is finished for the whole video, the model comes to the decoder layer. The encoder LSTM outputs have been sent to the decoder LSTM units with reference sentences (Fig. 7) that the videos have from the very beginning. Estimates have been compared with actual reference sentences and back propagation is performed in LSTM units.

Loss values are calculated as shown in Fig. 8. It has been ordered in accordance with the sentence structure in English in accordance with the sentence sequence that is learned from the reference sentences.

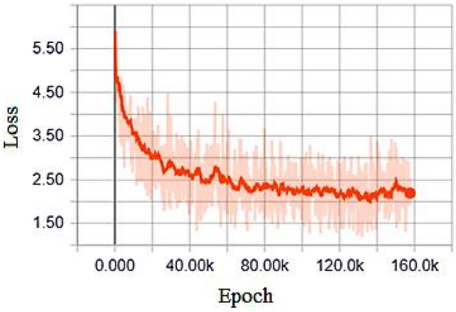


Fig. 8. Change in Loss Value.

During the last step, two LSTM layers have been used as in the training phase. The predicted sentences have been recorded in a file with the name of the video tagging the beginning and end. The purpose of these records is to measure the accuracy of the study. BLEU, CIDEr, ROUGE-L and METEOR metrics have been used for this measurement.

V. Experimental Results

The experimental studies have been conducted using BLEU, ROUGE_L, CIDEr and METEOR. The BLEU algorithm compares the array of expressions generated by the model with the reference expressions of the video and gives scores based on the number of matches. Different n-gram values have been used for BLEU. This means that n expressions have been compared according to the value of n. The difference between ROUGE and BLEU is that BLEU measures the incidence of machine-generated words (and/or n-grams) in the reference summaries. However, ROUGE measures the frequency of words (and/or n-grams) in the machine-generated summaries. CIDEr evaluates the quality of image descriptions. CIDEr measures the consensus between reference sentences and candidate image descriptions. For calculating this metric, each sentence has been represented with a set of 1-4 grams. Finally, METEOR uses harmonic mean of precision and recall of n-gram. It corrects some of the shortcomings of BLEU such as better matching of synonyms, though METEOR and BLEU measures are often used together in evaluation. Co-occurrences of n-grams in the candidate and reference sentences are calculated. Finally, the cosine similarity has been computed between n-grams of the candidate and the references.

The first of the different situations created for the evaluation of the study is the different epoch numbers applied in training. During the training of the properties determined with the VGG16 caffe model, epoch values of 1, 300 and 600 have been applied respectively (Table II). Rapid decrease in epoch score values means that the results obtained from the training are not efficient. The variation of the score between the value of 300 and the value of 600 can be interpreted, as the increase of the epoch value does not have a positive effect on training after a point.

Another situation created for evaluation is the use of different values in LSTM units. The experiment with 1000 epoch has more successful. This is because the back memory is larger. In this evaluation, the difference in CIDEr value is significant. It is seen in Table II that more successful results have been obtained in 300 epochs in experimental studies using 1000 LSTM units. 0.755 BLEU_1, 0.627 BLEU_2, 0.530 BLEU_3, 0.412 BLEU_4, 0.665 ROUGE_L, 0.651 CIDEr and 0.308 METEOR ratios have been obtained.

Finally, the situation evaluated is the results on different models. When the score results have been examined, it has been seen that the models used in feature extraction has a significant effect on the results. It has been seen from the high difference between BLEU and CIDEr that the treatment with VGG19 is more accurate than the other.

The test results in Fig. 9 also explain the difference between CIDEr score values. As a result of the experiments, a number of cases affecting the accuracy of the explanation to be created for a Video were examined and its effect documented with some metrics. Microsoft MSCOCO evaluation scripts have been used to make these comparisons.

TABLE. II. Epoch test results with VGG16 + 1000 unit LSTM

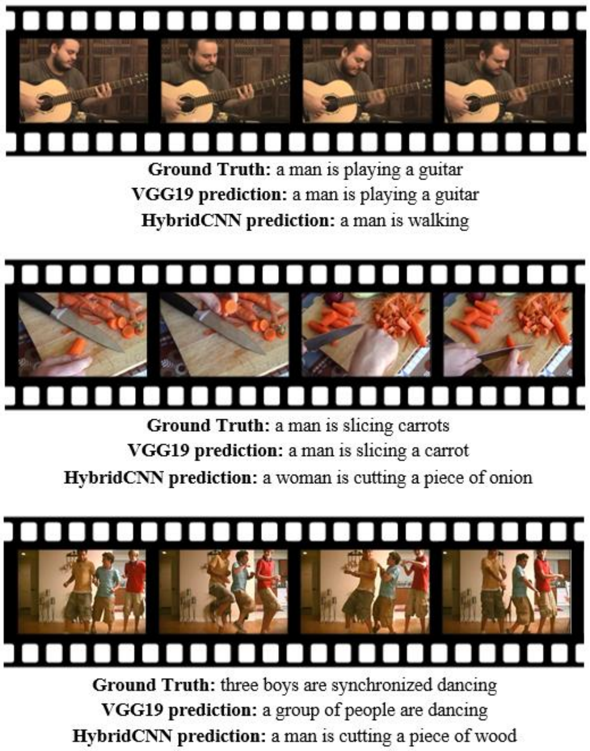
Epoch	BLEU_1	BLEU_2	BLEU_3	BLEU_4	ROUGE_L	CIDEr	METEOR
1 epoch	0.568	0.038	0.000	0.000	0.417	0.002	0.081
300 epoch	0.755	0.627	0.530	0.412	0.665	0.651	0.308
600 epoch	0.742	0.596	0.473	0.344	0.656	0.660	0.293

TABLE. III. LSTM units test results with VGG19

Unit	BLEU_1	BLEU_2	BLEU_3	BLEU_4	ROUGE_L	CIDEr	METEOR
200 LSTM	0.593	0.436	0.328	0.215	0.575	0.298	0.236
1000 LSTM	0.719	0.569	0.460	0.355	0.639	0.646	0.294

TABLE. IV. Experimental results for HybridCNN and VGG19 with 1000 epoch

Model	BLEU_1	BLEU_2	BLEU_3	BLEU_4	ROUGE_L	CIDEr	METEOR
HybridCNN	0.606	0.381	0.250	0.149	0.520	0.096	0.182
VGG19	0.719	0.569	0.460	0.355	0.639	0.646	0.294



Ground Truth: a man is playing a guitar
VGG19 prediction: a man is playing a guitar
HybridCNN prediction: a man is walking

Ground Truth: a man is slicing carrots
VGG19 prediction: a man is slicing a carrot
HybridCNN prediction: a woman is cutting a piece of onion

Ground Truth: three boys are synchronized dancing
VGG19 prediction: a group of people are dancing
HybridCNN prediction: a man is cutting a piece of wood

Fig. 9. A Comparison between Test-Generated Sentences and Reference Sentences.

VI. Conclusions

The automatic description of videos with natural sentences is a research problem that has been recently studied in the literature. In this study, it has been aimed to automatically obtain a natural sentence from a video. In this way, it is aimed to contribute to robotic vision tasks and help people with visual impairments. Automatic video description model should be able to express objects and events presented in the video. Automatic video description model also explains their relationships with each other in a natural sentence. To approach this problem, (Seq2Seq) model has been proposed to generate for video captioning. In this paper, the modern VGG-19 and VGG-16 CNN architectures have been used in conjunction with the LSTM. It has been aimed that the developed model has been learned to associate a variable-sized square array with a variable-sized word array. The performances of proposed models have been evaluated on the MSVD and are quantified using the BLEU, ROUGE, CIDEr and METEOR.

References

[1] World Health Organization, “Global Data on Visual Impairments 2018,” Geneve: WHO, 2018.

[2] N. Krishnamoorthy, G. Malkarnenkar, R. Mooney, K. Saenko, and S. Guadarrama, “Generating natural-language video descriptions using text-mined knowledge,” In Twenty-Seventh AAAI Conference on Artificial Intelligence, 2013.

[3] A. Farhadi, M. Hejrati, M. A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, D. Forsyth, “Every picture tells a story: Generating sentences from images,” In European conference on computer vision Springer, Berlin, Heidelberg, pp. 15-29, 2010.

[4] L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville, “Video description generation incorporating spatio-temporal features and a soft-attention mechanism,” arXiv preprint arXiv:1502.08029, 2015.

[5] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko, “Sequence to sequence-video to text,” In Proceedings of the IEEE international conference on computer vision, pp. 4534-4542, 2015.

[6] 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, 2019.

[7] A. Rohrbach, M. Rohrbach, N. Tandon, and B. Schiele, “A dataset for movie description,” In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3202-3212, 2015.

[8] N. Xu, A. Liu, Y., Wong, Y. Zhang, W. Nie, Y. Su, and M. Kankanhalli, “Dual-stream recurrent neural network for video captioning,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 29, no. 8, pp. 2482-2493, 2018.

[9] R. Krishna, K. Hata, F. Ren, L. Fei-Fei, and J. Carlos Niebles, “Dense-captioning events in videos,” In Proceedings of the IEEE international conference on computer vision, pp. 706-715, 2017.

[10] Z. Wu, X. Wang, Y. G. Jiang, H. Ye, and X. Xue, “Modeling spatial-temporal clues in a hybrid deep learning framework for video classification,” In Proceedings of the 23rd ACM international conference on Multimedia, pp. 461-470, 2015.

[11] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, “Beyond short snippets: Deep networks for video classification,” In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4694-4702, 2015.

[12] X. Wang, Y. F. Wang, and W. Y. Wang, “Watch, listen, and describe: Globally and locally aligned cross-modal attentions for video captioning,” arXiv preprint arXiv:1804.05448, 2018.

[13] Z. Wu, T. Yao, Y. Fu, and Y. G. Jiang, “Deep learning for video classification and captioning,” In Frontiers of multimedia research, pp. 3-29, 2017.

[14] 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, pp. 311-318, 2002.

[15] C. Y. Lin, “Rouge: A package for automatic evaluation of summaries,” Text Summarization Branches Out, 2004.

[16] R. Vedantam, C. Lawrence Zitnick, and D. Parikh, “Cider: Consensus-based image description evaluation,” In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4566-4575, 2015.

[17] 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 Proceedings of CVPR, pp. 2634–2641, 2013.

[18] M. Rohrbach, W. Qiu, I. Titov, S. Thater, M. Pinkal, and B. Schiele, “Translating video content to natural language descriptions,” In IEEE International Conference on Computer Vision (ICCV), 2013.

[19] A. Torabi, C. Pal, H. Larochelle, and A. Courville, “Using descriptive video services to create a large data source for video annotation research,” arXiv preprint, 2015.

[20] A. Rohrbach, M. Rohrbach, N. Tandon, and B. Schiele, “A dataset for movie description,” In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3202-3212, 2015.

[21] D. L. Chen, and W. B. Dolan, “Collecting highly parallel data for paraphrase evaluation,” In ACL: Human Language Technologies, Volume 1. Association for Computational Linguistics, pp. 190–200, 2011.