NCE Journal of Science and Engineering (NJSE), Vol. 2, June 2025

Automatic Video Script Generation based on Natural Language Processing using Deep Learning

Subash Panday[1,*], Sharmila Bista[1], Aavash Adhikari[1], Ashish Kumar Pokharel[1], Saksham Maharjan[1], and Aayush Shrestha[1]

[1]Department of Electronics and Computer Engineering, National College of Engineering, Lalitpur, Nepal

[*]Correspondence: subash@nce.edu.np

Manuscript received October 15, 2024; accepted January 5, 2025

***Abstract*—Detecting their surroundings and describing any image or video scenario in their native language is a very familiar and easy task for humans. However, extracting the temporal aspects of video scene, understanding its context and generating sequence of respective scripts is a challenging job for a machine. Development of methodologies that utilizes the concept of deep learning for applications that generates automatic scripts for the given video scene will be of great importance for people with visual impairment, creation of metadata for images and videos for use by search engines, robot vision systems and film makers and video editors. This article discusses the implementation of Trans- former architecture along with Convolutional Neural Network (CNN) for the effective generation of script for the given video sequence. This work involves the use of CNN to extract the frame features on datasets collected from ActivityNet and YouCookII. After the feature extraction, positional encoding is used for input embedding followed by implementation of Transformer architecture that includes encoder and decoder for effective processing of input frame sequence and respective generation of video scripts. The effectiveness of the work is then tested using Bilingual Evaluation Understudy (BLEU),Recall Oriented Understudy for Gisting Evaluation(ROUGE),Consensus-based Image Description Evaluation(CIDEr) to gauge the precision of text generation and Metric for Evaluation of Translation with Explicit Ordering (METEOR) for a comprehensive evaluation of model as performance metrics.**

***Keywords*—Deep Learning, Transformer Architecture, and Convolutional Neural Network.**

I. INTRODUCTION

WITH the advent of Information Technology, extensive practice of video making and posting is evident in the internet [1].Video making and posting has been an effective means of communication for today's world because of the fact that human find it easier to understand and interpret the content displayed in the given video sequence. However, understanding and interpreting a video content is a difficult task for a machine [2]. The problem with machine finding it difficult to interpret the video sequence is that video consists of temporal aspects that considers the relationship between the current frame and previous or subsequent frames in a video sequence. Also, video is a multimodal data and it consists of various foreground and background events and identifying the key events from the video sequence is difficult.

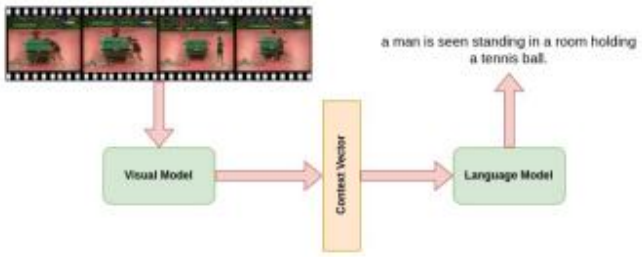

Fig. 1. Overall process of video script generation.

Additionally, generating a natural language script is a challenging task because natural language is context sensitive and bears ambiguity at various levels such as lexical ambiguity, syntax level ambiguity and referential ambiguity.Therefore, developing an application that can effectively interpret the video sequence and generate the script for the respective action delivered in the video is the need of the hour.

An approach to developing such application is to convert a video sequence to natural language i.e. generating the textual script for the given video sequence that provides a basis for creating a multimedia repository for video analysis, retrieval and summarization tasks [2]. Technologies such as Computer Vision, Natural Language Processing and Deep learning are used to achieve this goal.

In the prior works, deep learning methods such as Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs) have been used to achieve video scripting. Such methods use neural networks to learn features from videos, and then the features are used to generate scripts. Detecting an object, its attribute, its interaction with other object, the relationship among objects, and finally explaining the same so that anyone can quickly get what is happening in the video are the processes incorporated by such methods [3]. The overall process of video script generation is illustrated in Fig. 1.

However, these previous methods for video script generation have certain shortcomings. A key component of Recurrent Neural Networks (RNNs), recurrent layers are excellent at processing sequential data. The strength of this architecture is its capacity for sequential operations, which is essential for jobs

like time series analysis and language processing. The output of one step is transmitted back into the network as the input for the following step in a recurrent layer. The network's ability to retain prior knowledge is made possible by this looping mechanism, which is essential for comprehending sequence context. But there are two major ramifications to this sequential processing. Due to the fact that each step depends on the one before it, parallel processing can result in lengthier training and because of the vanishing gradient problem, where the network becomes less stable with time, they frequently suffer long-term interdependence. Conversely, Convolutional Layers, which are the fundamental components of Convolutional Neural Networks (CNNs), are well known for their effectiveness in handling spatial data, such as images. These layers extract features from the input data by scanning it through kernels, or filters. Depending on the job at hand, the network can focus on large or small features by adjusting the width of these kernels. Convolutional Layers have difficulties with long-term dependencies, despite their remarkable ability to capture spatial hierarchies and patterns in data. They are less appropriate for activities requiring a knowledge of the sequence's order or context since they do not naturally take sequential information into account.

Therefore, in this work use of the Transformer architecture that incorporates the concept of Encoder and Decoder is employed to overcome the shortcomings of the prior methods. The Transformer architecture is characterized by their unique attention mechanisms and parallel processing abilities. It is a neural network that learns the context of sequential data and generates new data out of it. They are specifically made to analyze the relationships between various elements in order to understand context and meaning and in order to do this, they nearly exclusively rely on a mathematical concept known as attention. The encoder in the transformer model takes an input and outputs a matrix representation of that input while the decoder takes in that encoded representation and iteratively generates an output. Transformer models don't have recurrency, which sets them apart from systems with recurrent layers such as RNNs. The Attention layer of the Transformer evaluates both aforementioned issues of RNNs and also transformer model overcomes the problem of CNN not being able to process sequential information by employing their capacity to comprehend long-range dependencies.

II. Related Works

A number of research works has been done by leveraging the concept of deep learning for video script generation. M.U.G. Khan, et al. [1] implemented a template-based approach built on a context free grammar incorporating spatial and temporal information to generate the natural language description of a video sequence. The approach faced the challenge during feature extraction and thus developed an approach to accommodate the missing information by creating a coherent description i.e. sample automatic annotations for video clips were created. M. Liu, et al. [2] developed an image caption generation system implementing dual attention mechanism. This method investigates visual attention to enhance comprehension of the image by using the Fully Convolutional Network's (FCN) produced image labels to create captions for the images. Additionally, the model makes use of textual attention to improve the information's integrity. M. Nabati, et al. [4] implemented video captioning using Boosted and Parallel Long Short-Term Memory Networks. The method comprised two LSTM layers and a word selection module. The first LSTM layer had the job of encoding frame features retrieved by a pre-trained deep Convolutional Neural Network (CNN). In the second LSTM layer, a unique architecture was implemented for video captioning by leveraging many decoding LSTMs in a parallel and boosting architecture. This layer, which is dubbed Boosted and Parallel LSTM (BPLSTM) layer, was produced by iteratively training LSTM networks using a special sort of AdaBoost algorithm during the training phase. During the testing phase, the outputs of BP-LSTMs were concurrently merged using the maximum probability criterion and word selection module.

M. Hoshino, et al. [5] proposed a method of automatic caption generation for video clips using keyframes and documentation summarization techniques. In order to reduce the processing time, instead of processing the entire video clips only the key frames from the video clips were extracted using KTS (Kernel Temporal Segmentation) and the respective captions for the keyframes were generated using NIC (Neural Image Captioning) which were aggregated using LSTM (Long Short Term Memory method). H. Xiao, et al. [6] implemented an attribute selection mechanism for video captioning. The framework made a soft selection over the identified video attributes by imposing an attention mechanism that was driven by the visual attention. The reinforcement learning technique was also utilized to enhance the selection of the valuable features. The framework employed CNN (Convolutional Neural Network) as an encoder to generate the visual representation and a RNN (Recurrent Neural Network) as a decoder to generate a sequence of words. H. Li, et al. [7] proposed implementation of REVNET for video captioning. This approach combined the traditional encoder-decoder framework with a reviewing network (REVnet) to recreate the previous concealed state. Backward flow was incorporated into the caption generation process by REVnet, which promoted the hidden state embedding of more information and made the resultant sentence's semantics more coherent. Additionally, REVnet regularized the framework's attention mechanism, which helped the model make better use of the semantic data gleaned from several frames. A. Ramani, et al. [8] developed automatic subtitle generation system for videos. The system implemented Deepspeech and CMU Sphinx to achieve the goal of generating the subtitles for the video being played in a media player. N. Yadav, et al. [9] proposed an approach for generating short video description using Deep-LSTM and attention mechanism. The approach used Deep-Long Short-Term Memory (Deep-LSTM) and Bahdanau Attention to introduce an Encoder-Decoder architecture. Convolution Neural Network (CNN) VGG16 and Deep-LSTM were employed in the encoder

to extract information from frames, and Deep-LSTM in conjunction with an attention mechanism was utilized to describe actions taken in the video. L. Yuan, et. Al [10] developed video captioning system with Listwise supervision. This model provided a novel framework for video captioning called Long Short-Term Memory with Listwise Supervision (LSTM-LS) and proposed to represent relative relationships of various video-sentence pairs. Using nearest-neighbor search, they were able to obtain a ranking list of sentences related to a specific sentence connected with each video in the training set. A set of rank triplets that can be used to evaluate the ranking list's quality represented the ranking information. Then, by building an LSTM model for sentence generation and optimizing the ranking quality throughout the entire list of sentences, the video captioning problem was resolved. T. Jin, et al. [11] achieved video captioning with Sparse Boundary-Aware Transformer. The method describes the technique to lessen redundancy in video representation: the sparse boundary-aware transformer (SBAT). Boundary-aware pooling procedure is used by SBAT to select different features from various scenarios and score multihead attention. In order to offset the local information loss caused by sparse operation, SBAT further incorporates a local correlation method. To improve the multimodal interaction, aligned cross-modal encoding approach based on SBAT was used. Y. Tian, et al. [12] developed an Audio-Visual interpretable and controllable video captioning system. The goal in this study was to separate the way the two modalities interact and present the first attempt at interpretable captioning for audio-visual videos. To make the design of an interpretable structure easier, they specifically proposed a novel multimodal convolutional neural network based audio-visual video captioning framework that does not require an RNN decoder. Additionally, they introduced a modality-aware feature aggregation module with a defined activation energy to determine which modality is more informative for word generation. Furthermore, the framework's interpretability granted control over the production of audiovisual sentences. To facilitate the generation of varied modality-aware captions by the proposed model, an audio-visual controller was implemented that manipulated the parameters of the modality-aware feature aggregation module.

III. Methodology

In the proposed methodology, initially, a dataset of video clips and their related scripts were gathered. Next, feature extraction using the CNN model was performed. The model was developed with the help of Large Language Model (LLM) i.e. Transformer. At last, the script generation is done by utilizing Long Short Term Memory (LSTM). The detailed information about the proposed approach is preseneted in the Fig. 2 and explained as well.

A. Datasets Collection

YouCook II: The YouCook II dataset is a large collection of culinary videos with approximately 2000 high-definition films

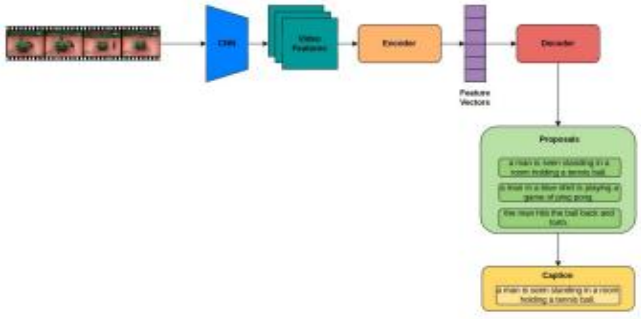

Fig. 2. Block diagram of proposed system.

and 14,000 clips overall [13]. The videos in the collection are heavily annotated with many sentences that accurately depict the items and events seen in the video frames. Moreover, the videos are separated into an average of 7.7 segments per video, ranging from 3 to 16 segments. The videos in the collection are great for researching activity understanding, action recognition, and fine-grained temporal localization because each one focuses on a particular recipe and offers detailed directions.3200 clips are used for validation, 9600 clips are used for training in the dataset and remaining for testing purpose.

ActivityNet: Approximately 849 hours and 20,000 video segments with 100,000 temporally annotated sentences covering a variety of intricate human behaviors are included in this dataset [14]. The dataset has a hierarchical structure with videos split into temporal chunks with a single caption describing each section. An average of 3.65 temporally localized sentences (or chunks) can be found in each of the 20,000 videos. When these descriptions are combined, they typically account for 94.6% of the content in the entire video.

B. Feature Extraction

Using the ActivityNet and YouCookII datasets, a pre-trained video feature extractor known as TSP (Temporally-Sensitive Pretraining) is employed to extract features [15]. The TSP is a novel supervised pretraining paradigm for clip features that improves temporal sensitivity by taking into account background clips and global video information in addition to training to identify activities. TSP's fundamental method makes use of 3D CNNs, or convolutional neural networks. Large-scale video datasets are used to pre-train these networks so they can extract temporally-sensitive characteristics from untrimmed videos. By emphasizing temporal region distinction, TSP improves learning and benefits the model in temporal action localization (TAL) tasks by improving boundary contrast between actions and their backdrop context.

C. Training Process

In the training process, the transformer architecture was implemented that consists of two main components: Encoder and Decoder.The overall workflow of the proposed methodology implementing transformer architecture is given in the Fig 3. The frames extracted from CNN was passed through the input embedding process which lays out the input tokens

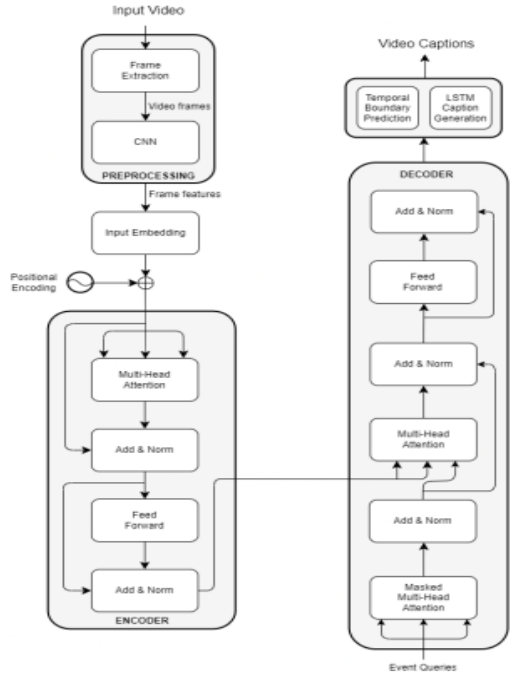

Fig. 3. Overall workflow of proposed methodology.

into vectors of a specified dimension (commonly d model =512) using learned embedding. The output from input embedding is then subjected to positional encoding which describes the location or position of an entity in a sequence so that each position is assigned a unique representation. The process of encoding and decoding is explained as:

Encoding Process: The transformer's encoder is made up of several encoder blocks. The output of the final encoder block serves as the input features for the decoder after an input vector passes through the encoder blocks. It is divided into multiple layers, each of which is composed of a feed-forward neural network and a self-attention mechanism. By calculating their relative relevance, the self-attention mechanism assists the model in concentrating on various input data points. As a result, regardless of their location, the model is able to capture interactions between various input components. The feed-forward neural network analyzes the data again to identify intricate patterns and features following the self-attention stage. This procedure is repeated by each layer of the encoder, which refines the data representation before forwarding it to the subsequent layer.

Decoding Process: There are several decoder blocks in the decoder as well. The encoder provides the features to each block of the decoder. It also includes numerous layers, with each layer consisting of three primary components: a feed-forward neural network, an encoder-decoder attention mechanism, and a self-attention mechanism. The decoder's self-attention mechanism ensures consistency by assisting in focusing on various portions of the output generated thus far. By taking into account the encoder's processed information, the encoder-decoder attention mechanism aids the decoder in concentrating on relevant parts of the incoming data.

C. Script Generation

Two key components are used to process the features from the transformer's decoder: temporal boundary prediction and LSTM caption synthesis. Events from the features are counted and predicted prior to these two elements. The features are then sent to both components in parallel.

Temporal Boundary Prediction

For video content to be reliably segmented into discrete event segments, which are then utilized to generate captions, the boundary prediction block is essential. The start and end times of each event are predicted by this block using an event counter and a localization head. The localization head uses the refined event queries to identify the temporal bounds of the events, while the event counter counts the number of events in the video. The boundary prediction block guarantees accurate temporal segmentation, which greatly enhances the coherence and correctness of the output captions. It does this by feeding upgraded representations of event queries into the localization and captioning heads in parallel.

LSTM Script Generation

The captioning head generates words in a sequential fashion using an LSTM network. The reference point of each event query is modified to more accurately capture the attributes of the event. When generating each word, the attention module focuses on the most essential frame elements surrounding these reference points, ensuring that the captions are logical and contextually appropriate. For every event, many captions are generated; the model's output predicts the captions with the highest likelihood. The model can more accurately concentrate on the segments of the video that are most relevant to each word thanks to this attention mechanism, which also makes the generated captions easier to read.

III. Evaluation and Fine Tuning

With YouCookII and ActivityNet datasets, the model passes through the feature extraction process, implementation of transformer for encoding and decoding. The experimental setup was conducted on NVIDIA 4060 GPU with 8GB of memory. The software environment included Python 3.11.9, PyTorch 2.3.1, and CUDA toolkit 11.8. for training purpose. Proposed system used a batch size of 1 to balance computational performance with memory limitations and started with a starting learning rate of 5e-5 and used a learning rate scheduler to gradually reduce it by 0.5 as the model converged. Based on the calculated gradients, the weights of the model was modified using an Adam optimizer. To avoid over-fitting, dropout with a probability of 0.1 was employed in the transformer layers.

Various loss functions are evaluated such as classification loss, counter loss, GIoU loss and caption loss. For measuring the difference between predicted class probabilities and true class labels cross-entropy loss is used. Counter Loss is measured to know the error in predicting the number of events. Bounding Box Loss is utilize to measure the accuracy of predicted bounding boxes for events, using L1 loss. GIoU Loss is used to measure the

Generalized Intersection over Union (GIoU) between predicted and ground-truth bounding boxes, which helps improve the quality of the bounding box predictions. Caption Loss measures the difference between predicted captions and ground-truth captions, using cross-entropy loss for sequence prediction. Evaluation is required in order to compare various approaches and assess machine learning models. Various approaches to model evaluation have been proposed [16]. Using the n-gram for both machine-produced and human-annotated sentences, the generated sentence's accuracy is compared to the ground truth sentence. The following provides a detailed description of the assessment metrics that are frequently employed in the field of video captioning: BLEU [17], ROUGE [18], METEOR [19], and CIDEr [20].

A. Bilingual evaluation understudy (BLEU)

A well-liked metric in natural language processing (NLP) activities, BLEU was first introduced in 2002 and is used to compare the accuracy of a machine-generated sentence to a set of human-generated sentences using the n-gram precision. BLEU is widely adopted in video captioning as it measures both the lexical similarity and word order between the generated and reference captions [20].

BLEU-1: Measures the precision of 1-grams (individual words) in the generated text compared to the reference texts. It provides an indication of how well the individual words in the generated text match those in the reference texts.

BLEU-2: Measures the precision of 2-grams (pairs of consecutive words) in the generated text compared to the reference texts. It captures some of the word pair dependencies.

BLEU-3: Measures the precision of 3-grams (triplets of consecutive words) in the generated text compared to the reference texts. It provides a more detailed measure of the sequence quality by considering triplets of words.

BLEU-4: Measures the precision of 4-grams (quadruplets of consecutive words) in the generated text compared to the reference texts. It gives an even finer measure of sequence quality by accounting for four-word sequences.

B. Recall oriented understudy for gisting evaluation (ROUGE)

ROUGE quantifies the overlap of ngrams, or consecutive word sequences, between the generated and reference captions in the video captioning domain. ROUGE and BLEU vary in that whilst BLEU measures n-gram overlap regardless of order and concentrates on precision-oriented evaluation, ROUGE takes into account n-gram order and captures significant content [21].

C. Metric for evaluation of translation with explicit ordering (METEOR)

The METEOR metric uses a combination of precision, recall, and alignment-based metrics to determine how similar a generated sentence is to a reference sentence.The METEOR is determined by first calculating recall and precision scores using the content word overlap between the reference and candidate captions. After that, the alignment score is determined by taking stemming and synonyms into consideration and tallying the number of matched unigrams. If a candidate's caption has more or less words than necessary, there will be a penalty. Ultimately, a harmonic mean is used to combine the precision, recall, and alignment scores to provide the METEOR score, which indicates how well the generated caption is done [22].

TABLE I
PERFORMANCE METRICS FOR THE PROPOSED MODEL

Model	METEOR	SODA	ROUGE	CIDEr
Proposed Model	0.1179	0.1008	0.2826	0.31150

TABLE II
BLEU SCORES FOR THE PROPOSED MODEL.

Model	$BLEU_1$	$BLEU_2$	$BLEU_3$	$BLEU_4$
Proposed Model	0.523503	0.326151	0.20083	0.1261

D. Consensus-based image description evaluation (CIDEr)

By calculating the cosine similarity between the weighted n-gram vectors of the reference captions and the created caption, CIDEr calculates the consensus between the various reference captions and the generated caption. It captures the significance and uniqueness of the generated captions by taking into account both precision and diversity in the evaluation process. As a result, CIDEr and human judgment correlate more closely [23] [24].

Data presented in Table I shows the METEOR, SODA, ROUGE and CIDer values evaluated after the successful completion of model development and values in Table II shows the evaluation BLUE score with various levels for video script generation.

Model Checkpointing: The model checkpoints were updated at the end of each epoch, and the best-performing model was also saved. The model was trained over 30 epochs, each epoch took an average time of approximately 20 minutes taking a total time of approximately 10 hours.

V. Conclusion

In this study, a deep learning framework is proposed to boost the video script generation by learning spatio-temporal aspects of the video scene perfectly. The implementation of transformer helps to capture interaction between the various input components and refines the data in each steps before forwarding to next. Localization of events with its corresponding scripts are accurately performed during the model training phase and achieved improved performance parameters. Furthermore, model can be developed using diverse datasets and focused on high level semantics of video frames with long duration videos. Additionally, the system can be modified for machine translation application that effectively converts natural language scripts of video sequence from one natural language to another.

VI. Acknowledgement

The authors gratefully acknowledge the research grant provided by the National College of Engineering, Institute of Engineering, Tribhuvan University.

REFERENCES

[1] Maofu Liu, Lingjun Li, Huijun Hu, Weili Guan, and Jing Tian. Image caption generation with dual attention mechanism. *Information Processing Management*, 57(2):102178, 2020.

[2] Muhammad Usman Ghani Khan, Nouf Al Harbi, and Yoshihiko Gotoh. A framework for creating natural language descriptions of video streams. *Information Sciences*, 303:61–82, 2015.

[3] Mohammad Saif Wajid, Hugo Terashima-Marin, Peyman Najafirad, and Mohd Anas Wajid. Deep learning and knowledge graph for image/video captioning: A review of datasets, evaluation metrics, and methods. *Engineering Reports*, 6(1):e12785, 2024.

[4] Huanhou Xiao and Jinglun Shi. A novel attribute selection mechanism for video captioning. In 2019 IEEE International Conference on Image Processing (ICIP), pages 619–623, 2019.

[5] Huidong Li, Dandan Song, Lejian Liao, and Cuimei Peng. Revnet: Bring reviewing into video captioning for a better description. In 2019 IEEE International Conference on Multimedia and Expo (ICME), pages 1312–1317, 2019.

[6] Aditya Ramani, Asmita Rao, V Vidya, and VR Badri Prasad. Automatic subtitle generation for videos. In 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), pages 132–135, 2020.

[7] Naveen Yadav and Dinesh Naik. Generating short video description using deep-lstm and attention mechanism. In 2021 6th International Conference for Convergence in Technology (I2CT), pages 1–6, 2021

[8] Yuan Liu, Xue Li, and Zhongchao Shi. Video captioning with listwise supervision. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 31, 2017.

[9] Tao Jin, Siyu Huang, Ming Chen, Yingming Li, and Zhongfei Zhang. Sbat: Video captioning with sparse boundary-aware transformer, 2020.

[10] Yapeng Tian, Chenxiao Guan, Justin Goodman, Marc Moore, and Chenliang Xu. Audio-visual interpretable and controllable video captioning. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition workshops, 2019.

[11] Anoop Cherian, Jue Wang, Chiori Hori, and Tim Marks. Spatio-temporal ranked-attention networks for video captioning. In Proceedings of the IEEE/CVF winter conference on applications of computer vision, pages 1617–1626, 2020.

[12] Murk Chohan, Adil Khan, Muhammad Saleem Mahar, Saif Hassan, Abdul Ghafoor, and Mehmood Khan. Image captioning using deep learning: A systematic literature review. International Journal of Advanced Computer Science and Applications, 11(5), 2020.

[13] Luowei Zhou, Chenliang Xu, and Jason Corso. Towards automatic learning of procedures from web instructional videos. Proceedings of the AAAI Conference on Artificial Intelligence, 32, 04 2018.

[14] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In 2017 IEEE International Conference on Computer Vision (ICCV), pages 706–715, 2017.

[15] Humam Alwassel, Silvio Giancola, and Bernard Ghanem. Tsp: Temporally-sensitive pretraining of video encoders for localization tasks. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops, 2021.

[16] Phan Sang, Gustav Henter, Yusuke Miyao, and Shin'ichi Satoh. Consensus-based sequence training for video captioning. 12 2017.

[17] Soheyla Amirian, Khaled Rasheed, Thiab R Taha, and Hamid R Arabnia. Automatic image and video caption generation with deep learning: A concise review and algorithmic overlap. IEEE access, 8:218386–218400, 2020.

[18] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In Text summarization branches out, pages 74–81, 2004.

[19] Andrei de Souza Ina´cio and Heitor Silve´rio Lopes. Evaluation metrics for video captioning: A survey. Machine Learning with Applications, 13:100488, 2023.

[20] R Vedantam, C Lawrence Zitnick, and D Parikh. Consensus-based image description evaluation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 4566–4575.

[21] Chenggang Yan, Yunbin Tu, Xingzheng Wang, Yongbing Zhang, Xinhong Hao, Yongdong Zhang, and Qionghai Dai. Stat: Spatial-temporal attention mechanism for video captioning. IEEE transactions on multimedia, 22(1):229–241, 2019.

[22] Nayyer Aafaq, Ajmal Mian, Wei Liu, Syed Zulqarnain Gilani, and Mubarak Shah. Video description: A survey of methods, datasets, and evaluation metrics. ACM Computing Surveys (CSUR), 52(6):1–37, 2019.

[23] Sheng Li, Zhiqiang Tao, Kang Li, and Yun Fu. Visual to text: Survey of image and video captioning. IEEE Transactions on Emerging Topics in Computational Intelligence, 3(4):297–312, 2019.

[24] Xin Wang, Wenhu Chen, Jiawei Wu, Yuan-Fang Wang, and William Yang Wang. Video captioning via hierarchical reinforcement learning. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 4213–4222, 2018.

[25] Akira Shibata and Takashi Yukawa. An automatic text generation system for video clips using machine learning technique. In *TREC Video Retrieval Evaluation*, 2018.

[26] Adel Jalal Yousif and Mohammed H. Al-Jammas. Exploring deep learning approaches for video captioning: A comprehensive review. *e-Prime - Advances in Electrical Engineering, Electronics and Energy*, 6:100372, 2023.

[27] Dhruv Sharma, Chhavi Dhiman, and Dinesh Kumar. Evolution of visual data captioning methods, datasets, and evaluation metrics: A comprehensive survey. *Expert Systems with Applications*, 221:119773, 2023.

[28] Soheyla Amirian, Khaled Rasheed, Thiab R. Taha, and Hamid R. Arabnia. Automatic image and video caption generation with deep learning: A concise review and algorithmic overlap. *IEEE Access*, 8:218386–218400, 2020.

[29] Ravi Bansal and Sandip Chakraborty. Visual content based video retrieval on natural language queries. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, SAC '19, page 212–219, New York, NY, USA, 2019. Association for Computing Machinery.