Video captioning with stacked attention and semantic hard pull

Md. Mushfiqur Rahman[1], Thasin Abedin[2], Khondokar S.S. Prottoy[2],
Ayana Moshruba[2] and Fazlul Hasan Siddiqui[3]

[1] Department of Computer Science and Engineering, Islamic University of Technology, Gazipur, Bangladesh
[2] Department of Electrical and Electronic Engineering, Islamic University of Technology, Gazipur, Bangladesh
[3] Department of Computer Science & Engineering, Dhaka University of Engineering and Technology, Gazipur, Bangladesh

ABSTRACT

Video captioning, *i.e.*, the task of generating captions from video sequences creates a bridge between the Natural Language Processing and Computer Vision domains of computer science. The task of generating a semantically accurate description of a video is quite complex. Considering the complexity, of the problem, the results obtained in recent research works are praiseworthy. However, there is plenty of scope for further investigation. This paper addresses this scope and proposes a novel solution. Most video captioning models comprise two sequential/recurrent layers—one as a video-to-context encoder and the other as a context-to-caption decoder. This paper proposes a novel architecture, namely Semantically Sensible Video Captioning (SSVC) which modifies the context generation mechanism by using two novel approaches—"stacked attention" and "spatial hard pull". As there are no exclusive metrics for evaluating video captioning models, we emphasize both quantitative and qualitative analysis of our model. Hence, we have used the BLEU scoring metric for quantitative analysis and have proposed a human evaluation metric for qualitative analysis, namely the Semantic Sensibility (SS) scoring metric. SS Score overcomes the shortcomings of common automated scoring metrics. This paper reports that the use of the aforementioned novelties improves the performance of state-of-the-art architectures.

Corresponding author
Md. Mushfiqur Rahman,
mushfiqur11@iut-dhaka.edu

INTRODUCTION

The incredible success in the image captioning domain has led the researchers to explore similar avenues like video captioning. Video captioning is the process of describing a video with a complete and coherent caption using Natural Language Processing. The core mechanism of video captioning is based on the sequence-to-sequence architecture (*Gers, Schmidhuber & Cummins, 2000*). In video captioning models, the encoder encodes the visual stream and the decoder generates the caption. Such models are capable of retaining both the spatial and temporal information which is essential for generating semantically correct video captions. This requires the video to be split up into a sequence of frames. The

Caption: "Potatoes are being chopped by a knife"

Figure 1 Video captioning task.

model uses these frames as input and generates a series of meaningful words in the form of a caption as output. In Fig. 1, an example of a video captioning task has been shown.

Video captioning has many applications, for example, the interaction between humans and machines, aid for people with visual impairments, video indexing, information retrieval, fast video retrieval, etc. Unlike image captioning where only spatial information is required to generate captions, video captioning requires the use of a mechanism that combines spatial information with temporal information to store both the higher level and the lower level features to generate semantically sensible captions. Although there are good works in this field, there is still plenty of opportunity for investigation. One of the main opportunities is improving the ability of models to extract high-level features from videos to generate a more meaningful caption. This paper primarily focuses on this aspect.

In our paper, we propose a novel architecture that is based on the seq2seq model proposed by *Venugopalan et al. (2015)*. Our novel architecture tries to improve upon this work following the guidelines laid out by preceding literature. Our novel architecture aims to show a possible direction for potential future research. The goal of our model is to encode a video (presented in the form of a sequence of images) in order to extract information from it and decode encoded data to generate a sentence (presented in the form of a sequence of words). On the encoder side, along with the bi-directional LSTM layers, our model uses the combination of two novel methods—a variation of dual-attention (*Nam, Ha & Kim, 2017*), namely, Stacked Attention, and a novel information extraction method, namely, Spatial Hard Pull. The Stacked Attention network sets the priority to the object in the video layer-by-layer. To overcome the redundancy of similar information being lost in the LSTM layers, we introduce the Spatial Hard Pull layer. On the decoding

side, we employ a sequential decoder with a single layer LSTM and a fully connected layer to generate a word from a given context produced by the encoder.

Most text generation architectures use BLEU (*Papineni et al., 2002*) as the scoring metrics. But due to it's inability of considering recall, few variations, including ROUGE (*Lin, 2004*), METEOR (*Banerjee & Lavie, 2005*), etc., are introduced. Though these automatic scoring metrics are modified in different ways to give more meaningful results, they have their shortcomings (*Kilickaya et al., 2016*; *Aafaq et al., 2019b*). On top of that, no scoring metrics, solely used for the purpose of video captioning, are available to the best of our knowledge. Some relevant works (*Xu et al., 2017*; *Pei et al., 2019*) have used human evaluation. To get a better understanding of the captioning capability of our model, we perform qualitative analysis based on human evaluation and propose our metric, "Semantic Sensibility Score" or "SS Score", in short, for video captioning.

RELATED WORKS

For the past few decades, much work has been conducted on analysing videos to extract different forms of information, such as, sports-feature summary (*Shih, 2017*; *Ekin, Tekalp & Mehrotra, 2003*; *Ekin & Tekalp, 2003*; *Li & Sezan, 2001*), medical video analysis (*Quellec et al., 2017*), video finger-print (*Oostveen, Kalker & Haitsma, 2002*) and other high-level features (*Chang et al., 2005*; *Divakaran, Sun & Ito, 2003*; *Kantorov & Laptev, 2014*). These high-level feature extraction mechanisms heavily relied on analyzing each frame separately and therefore, could not retain the sequential information. When the use of memory retaining cells like LSTM (*Gers, Schmidhuber & Cummins, 2000*) became computationally possible, models were only then capable of storing meaningful temporal information for complex tasks like caption generation (*Venugopalan et al., 2015*). Previously, caption generation was mostly treated with template based learning approaches (*Kojima, Tamura & Fukunaga, 2002*; *Xu et al., 2015*) or other adaptations of statistical machine translation approach (*Rohrbach et al., 2013*).

Sequence-to-sequence architecture for video captioning

Video is a sequence of frames and the output of a video captioning model is a sequence of words. So, video captioning can be classified as a sequence-to-sequence(seq2seq) task. *Sutskever, Vinyals & Le (2014)* introduce the seq2seq architecture where the encoder encodes an input sentence, and the decoder generates a translated sentence. After the remarkable result of seq2seq architecture in different seq2seq tasks (*Shao et al., 2017*; *Weiss et al., 2017*), it is only intuitive to leverage this architecture in video captioning works like (*Venugopalan et al., 2015*). In recent years, different variations of the base seq2seq architecture has been widely used, *e.g.*, hierarchical approaches (*Baraldi, Grana & Cucchiara, 2017*; *Wang et al., 2018a*; *Shih, 2017*), variations of GAN (*Yang et al., 2018*), boundary-aware encoder approaches (*Shih, 2017*; *Baraldi, Grana & Cucchiara, 2017*) etc.

Attention in sequence-to-sequence tasks

In earlier seq2seq literature (*Venugopalan et al., 2015*; *Pan et al., 2017*; *Sutskever, Vinyals & Le, 2014*), the decoder cells generate the next word from the context of the preceding

word and the fixed output of the encoder. As a result, the overall context of the encoded information often got lost and the generated output became highly dependent on the last hidden cell-state. The introduction of attention mechanism (*Vaswani et al., 2017*) paved the way to solve this problem. The attention mechanism enables the model to store the context from the start to the end of the sequence. This allows the model to focus on certain input sequences on each stage of output sequence generation (*Bahdanau, Cho & Bengio, 2014*; *Luong, Pham & Manning, 2015*). *Luong, Pham & Manning (2015)* proposed a combined global-local attention mechanism for translation models. In global attention scheme, the whole input is given attention at a time, while the local attention scheme attends to a part of the input at a given time. The work on video captioning enhanced with these ideas. *Bin et al. (2018)* describe a bidirectional LSTM model with attention for producing better global contextual representation as well as enhancing the longevity of all the contexts to be recognized. *Gao et al. (2017)* build a hierarchical decoder with a fused GRU. Their network combines a semantic information based hierarchical GRU, and a semantic-temporal attention based GRU and a multi-modal decoder. *Ballas et al. (2016)* proposed to leverage the frame spatial topology by introducing an approach to learn spatio-temporal features in videos from intermediate visual representations using GRUs. Similarly, several other variations of the attention exists including multi-faceted attention (*Long, Gan & De Melo, 2018*), multi-context fusion attention (*Wang et al., 2018a*) etc. All these papers use one attention at a time. This limits the available information for the respective models. *Nam, Ha & Kim (2017)* introduce a mechanism to use multiple attentions. With their dual attention mechanism, they have retained visual and textual information simultaneously. Ziqi Zhang and team have achieved commendatory scores by proposing an object relational graph (ORG) based encoder capturing more detailed interaction features and designed a teacher-recommended learning (TRL) method to integrate the abundant linguistic knowledge into the caption model (*Zhang et al., 2020*).

METHODOLOGY

As shown in Fig. 2, this paper proposes a novel architecture that uses a combination of stacked-attention (see Fig. 3) and spatial-hard-pull on top of a base video-to-text architecture to generate captions from video sequences. This paper refers to this architecture as Semantically Sensible Video Captioning (SSVC).

Data pre-processing and representation

The primary input of the model is a video sequence. The data pre-processor converts a video clip into a usable video sequence format of 15 frames before passing it to the actual model. Each converted video sequence contains 15 frames placed separated by an equal time gap. The primary output of the model is a sequence of words. The words are stacked to generate the required caption.

Visual feature extraction

A video is nothing but a sequence of frames. Each frame is a 2D image with n channels. In sequential architectures, either the frames are directly passed into ConvLSTM

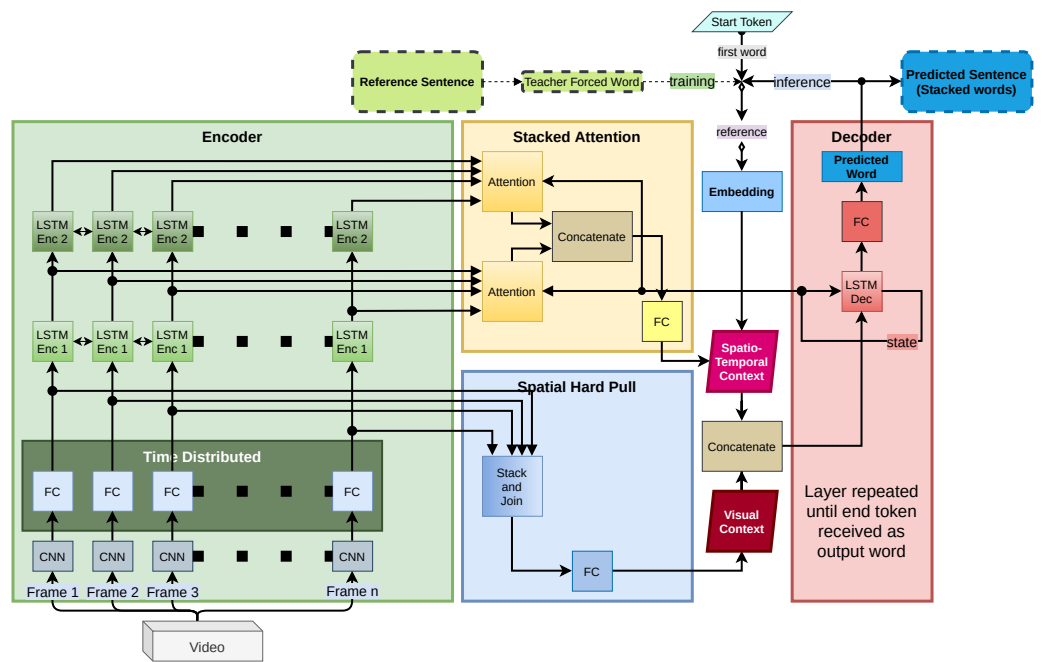
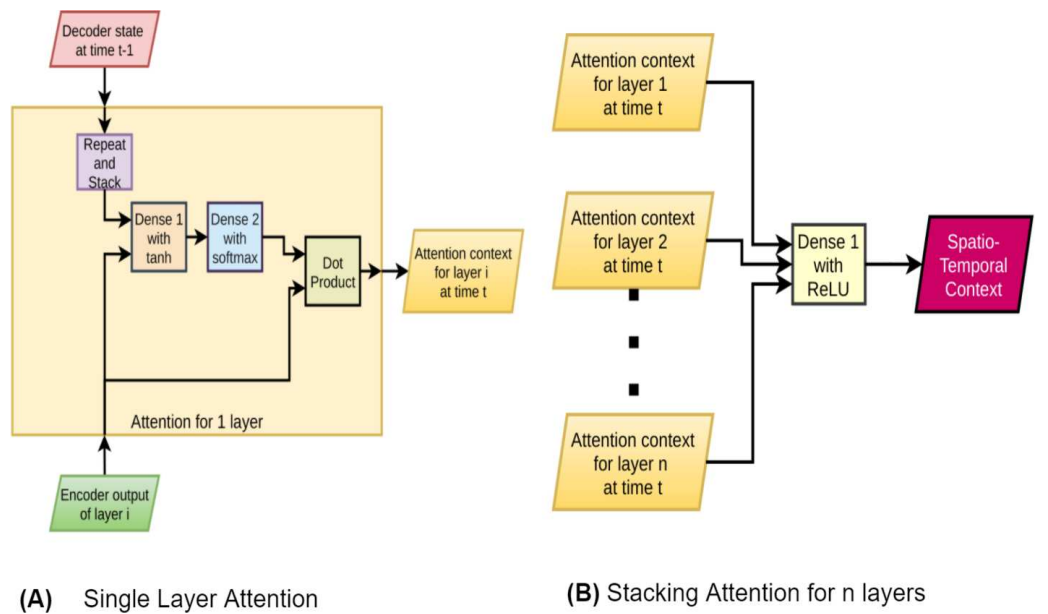



Figure 2 Proposed model with stacked attention and spatial hard pull.

(A) Single Layer Attention

(B) Stacking Attention for n layers

Figure 3 Diagram of stacked attention.

(*Xingjian et al., 2015*) layer(s) or the frames are individually passed through a convolutional block and then are passed into LSTM (*Gers, Schmidhuber & Cummins, 2000*) layer(s). For our computational limitations, our model uses the latter option. Like "Sequence to Sequence–Video to Text" (*Venugopalan et al., 2015*), our model uses a pre-trained VGG16 model (*Simonyan & Zisserman, 2014*) and extracts the *fc7* layer's output. This CNN layer converts each $(256 \times 256 \times 3)$ shaped frame into (1×4096) shaped vectors. These vectors are primary inputs of our model.

Textual feature representation

Each video sequence has multiple corresponding captions and each caption has a variable number of words. In our model, to create captions of equal length all the captions are padded with "pad" markers. The "pad" markers help create a uniformity in the data structure. The inclusion of "pad" markers do not create any change in the output as they are omitted during the conversion of tokenized words to complete sentences. A start marker, and an end marker, marks the start and end of each caption. The entire text data is tokenized, and each word is represented by a one-hot vector of shape $(1 \times uniquewordcount)$. So, a caption with m words is represented with a matrix of shape $(m \times 1 \times uniquewordcount)$. Instead of using these one-hot vectors directly, our model embeds each word into vectors of shape $(1 \times embeddingdimension)$ with a pre-trained embedding layer. The embedded vectors are semantically different and linearly distant in vector space from others on the basis of relationship of the corresponding words.

Base architecture

Like most sequence-to-sequence models, our base architecture consists of a sequential encoder and a sequential decoder. The encoder converts the sequential input vectors into contexts and the decoder converts those contexts into captions. This work proposes an encoder with double LSTM layers with stacked attention. The introduction of a mechanism that stacks attention and the mechanism of pulling spatial information from input vectors are the two novel concepts in this paper and are discussed in detail in later sections. The purpose of using the hard-pull layer is to bring superior extraction capabilities to the model. Since the rest of the model relies on time-series information, the hard-pull layer is necessary for combining information from separate frames and extract general information. The purpose of stacking attention layers is to attain a higher quality temporal information retrieval capability.

Multi-layered sequential encoder

The proposed method uses a time-distributed fully connected layer followed by two consecutive bi-directional LSTM layers. The fully connected layer works on each frame separately and then their output moves to the LSTM layers. In sequence-to-sequence literature, it is common to use stacked LSTM for encoder. For it, our intuition is, the two layers capture separate information from the video sequence. Figure 4 shows having two layers ensures optimum performance. The output of the encoder is converted into a context. In relevant literature, this context is mostly generated using a single attention

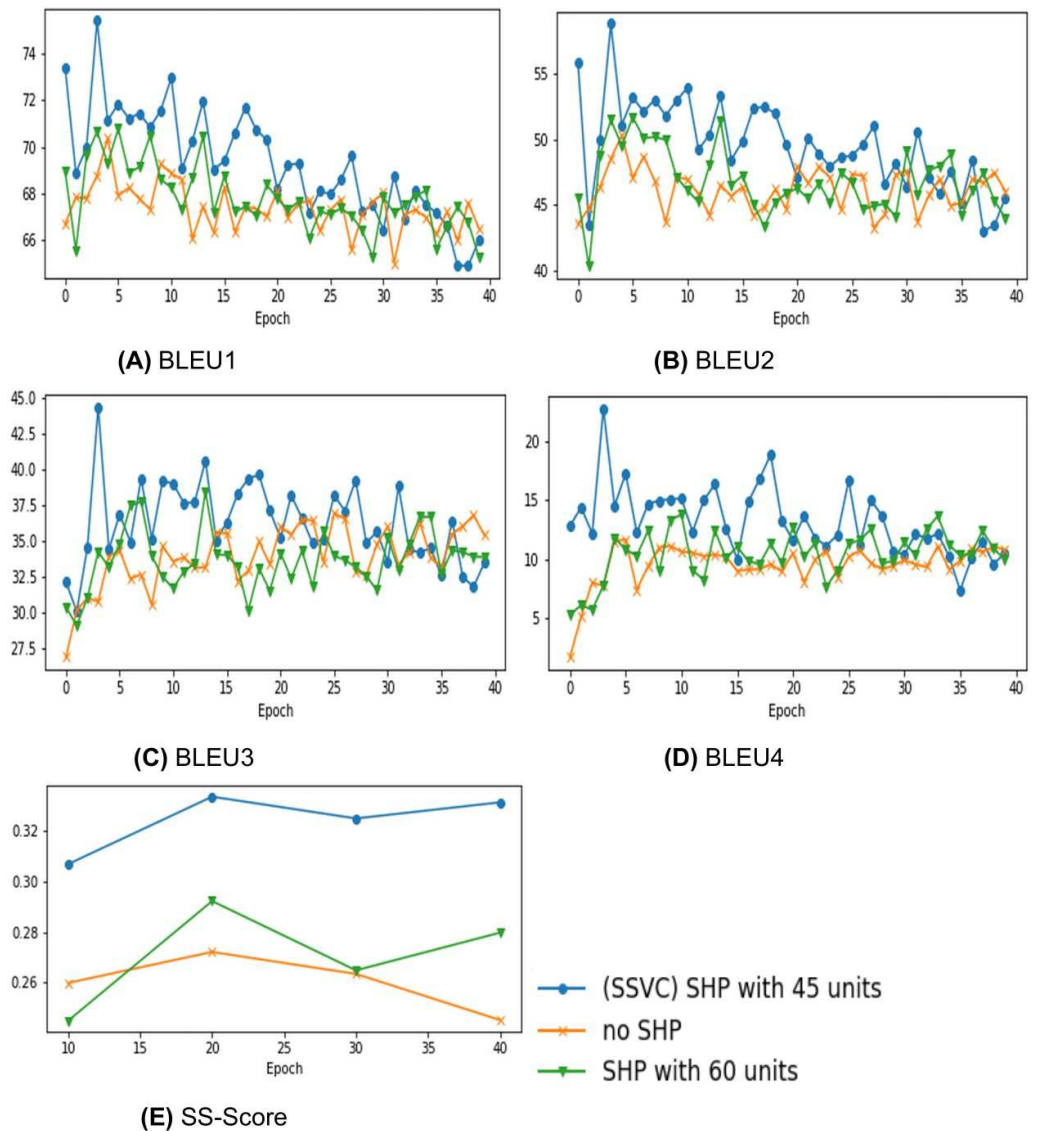



Figure 4 Comparing Stacked Attention with variations in encoder attention architecture. (A) BLEU1, (B) BLEU2, (C) BLEU3, (D) BLEU4, (E) SS-Score.

layer. This is where this paper proposes a novel concept. With the mechanism mentioned in later sections, our model generates a spatio-temporal context.

Single-layered sequential decoder

The proposed decoder uses a single layer LSTM followed by a fully connected layer to generate a word from a given context. In relevant literature, many models have used stacked decoder. Most of these papers suggest, each layer of decoder handles separate information, while our model uses a single layer. Our experimental results show that having stacked decoder does not improve the result much for our architecture. Therefore,

instead of stacking decoder layers, we increased the number of decoder cells. Specifically, we have used twice as many cells in decoder than in encoder and it has shown the optimum output during experimentation.

Training and inference behaviour

To mark the start of a caption and to distinguish the mark from the real caption, a "start" token is used at the beginning. The decoder uses this token as a reference to generate the first true word of the caption. Figure 2 represents this as "first word". During inference, each subsequent word is generated with the previously generated word as reference. The sequentially generated words together form the desired caption. The loop terminates upon receiving the "end" marker.

During training, if each iteration in the generation loop uses previously generated word, then one wrong generation can derail the entire remaining caption. Thus error calculation process becomes vulnerable. To overcome this, like most seq-to-seq papers, we use the teacher forcing mechanism (*Lamb et al., 2016*). The method uses words from the original caption as reference for generating the next words during the training loop. Therefore, the generation of each word is independent of previously generated words. Figure 2 illustrates this difference in training and testing time behaviour. During training, "Teacher Forced Word" is the word from the reference caption for that iteration.

Proposed context generation architecture

The paper proposes two novel methods. The methods show promising signs to make progress in the field of video captioning.

Stacked attention

Attention creates an importance map for individual vectors from a sequence of vectors. In text-to-text, *i.e.*, translation models, this mapping creates a valuable information that suggests which word or phrase in the input side has higher correlation to which words and phrases in the output. However, in video captioning, attention plays a different role. For a particular word, instead of determining which frame (from original video) or frames to put more emphasis on, the stacked attention emphasizes on objects. This paper uses a stacked LSTM. Like other relevant literature (*Venugopalan et al., 2015*; *Song et al., 2017*), this paper reports separate layers to carry separate information. So, if each layer has separate information, it is only intuitive to generate separate attention for each layer. Our architecture stacks the separately generated attentions and connects them with a fully connected layer with *tanh* activation. The output of this layer determines whether to put more emphasis on the object or the action.

$$f_{attn}([h,ss]) = a_s(W_2 * a_{tanh}(W_1[h,ss] + b_1) + b_2) \tag{1}$$

$$c_{attn} = dot(h, f_{attn}([h,ss])) \tag{2}$$

$$c_{st} = a_{relu}(W_{st}[c_{attn_1}, c_{attn_2}, \dots, c_{attn_n}] + b_{st}) \tag{3}$$

where,

- h = encoder output for 1 layer
- ss = decoder state is repeated to match h's dimension
- $a_s(x) = \frac{\exp(x-\max(x))}{\sum \exp(x-\max(x))atten}$
- n = number of attention layers to be stacked
- c_{attn} = context for single attention
- $dot()$ function represents scalar dot-product
- c_{st} = stacked context for n encoder layers.

Equation (1) is the attention function. Equation (2) uses the output of this function to generate the attention context for one layer. Equation (3) combines the attention context of several layers to generate the desirable spatio-temporal context. The paper also refers to this context as "stacked context". Figure 3 corresponds with these equation. In SSVC, we have particularly used $n = 2$, where n is the number of attention layers in the stacked attention.

The stacked attention mechanism generates the spatio-temporal context for the input video sequence. All types of low-level context required to generate the next word is available in this novel context generation mechanism.

Spatial Hard Pull

Amaresh & Chitrakala (2019) mentions that most successful image and video captioning models mainly learn to map low-level visual features to sentences. They do not focus on the high-level semantic video concepts - like actions and objects. By low-level features, they meant object shapes and their existence in the video. High-level features refer to proper object classification with position in the video and the context in which the object appears in the video. On the other hand, our analysis of previous architectures shows that almost identical information is often found in nearby frames of a video. However, passing the frames through LSTM layer does not help to extract any valuable information from this almost identical information. So, we have devised a method to hard-pull the output of the time-distributed layer and use it to add high-level visual information to the context. This method enables us to extract meaningful high-level features, like objects and their relative position in the individual frames.

This method extracts information from all frames simultaneously and does not consider sequential information. As the layer pulls spatial information from sparsely located frames, this paper names it "Spatial Hard Pull" layer. It can be compared to a skip connection. But unlike other skip connections, it skips a recurrent layer, and directly contributes to the context. The output units of the fully connected (FC) layer of this spatial-hard-pull layers determines how much effect will the sparse layer have on the context. Figure 5 indicates the performance improvement in the early stages due to SHP layer and the fall of scores in the later stages due to high variance.

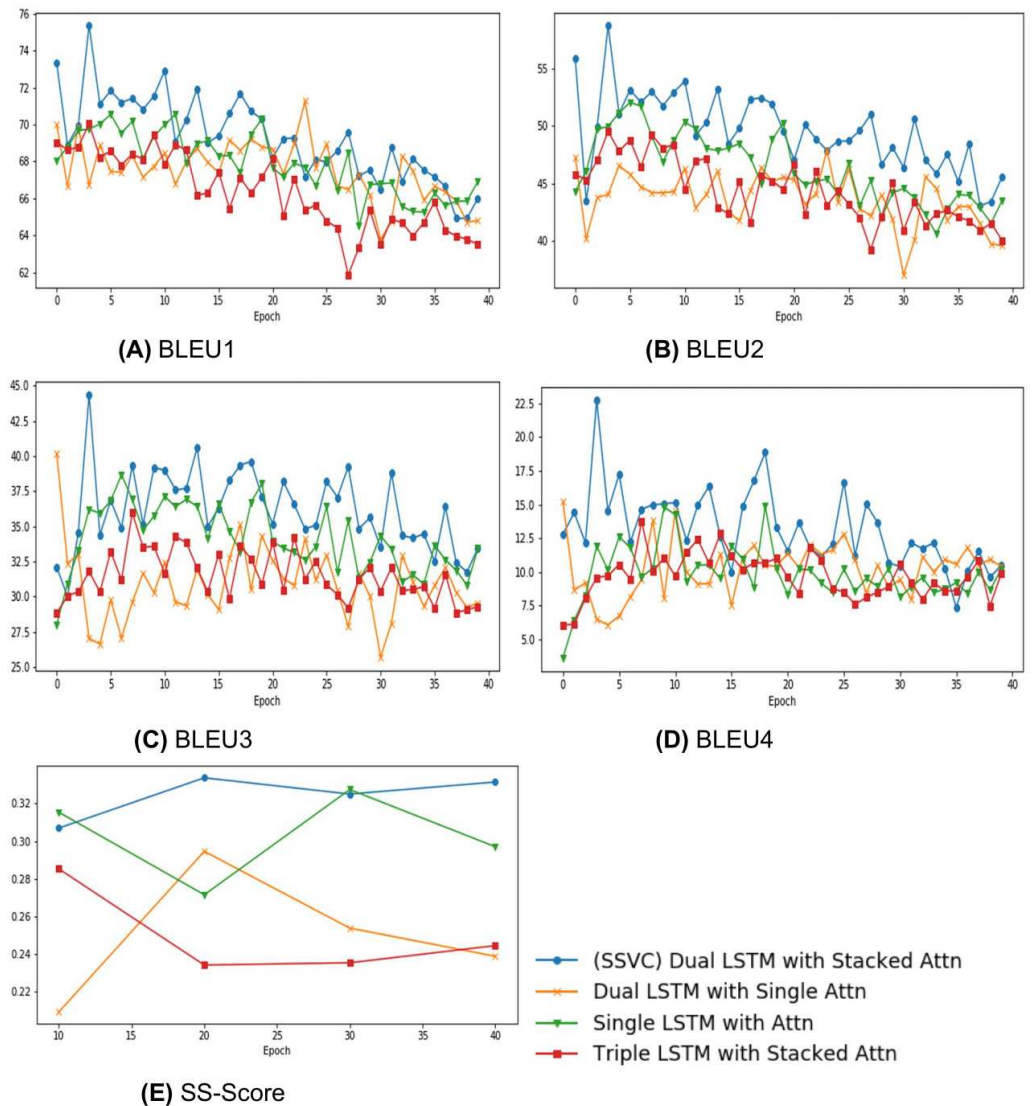



Figure 5 **Evaluating model performance with varied hard-pull units.** (A) BLEU1, (B) BLEU2, (C) BLEU3, (D) BLEU4, (E) SS-Score.

PROPOSED SCORING METRIC

No automatic scoring metric has been designed yet for the sole purpose of video captioning. The existing metrics that have been built for other purposes, like neural machine translation, image captioning, etc., are used for evaluating video captioning models. For quantitative analysis, we use the BLEU scoring metric (*Papineni et al., 2002*). Although these metrics serve similar purposes, according to *Aafaq et al. (2019b)*, they fall short in generating "meaningful" scores for video captioning.

BLEU is a precision-based metric. It is mainly designed to evaluate text at a corpus level. BLEU metric can be calculated in reference to 1 sentence or in reference to a corpus of sentences (*Brownlee, 2019*). Though the BLEU scoring metric is widely used, *Post (2018)*;

(A) SSVC: "a woman is **cutting a piece of meat**"
GT: "a woman is cutting into the fatty areas of a pork chop"
SS Score: **1.0** ,   BLEU1: **1.0**,   BLEU2: **1.0**,   BLEU3: **1.0**,   BLEU4: **1.0**

(B) SSVC: "a person is **slicing a tomato**"
GT: "someone wearing blue rubber gloves is slicing a tomato with a large knife"
SS Score:**0.825** ,   BLEU1: **1.0**,   BLEU2: **1.0**,   BLEU3: **1.0**,   BLEU4: **1.0**

(C) SSVC: "a man is **mixing ingredients in a bowl**"
GT: "chicken is being season"
SS Score:**0.94** ,   BLEU1: **1.0**,   BLEU2: **0.84**,   BLEU3: 0.61,   BLEU4: *0.0*

Figure 6 In (A) (Source: **https://www.youtube.com/watch?v=6t0BpjwYKco{&}t=230s**) and **(B)** (Source: **https://www.youtube.com/watch?v=j2Dhf-xFUxU{&}t=20s**), our model is able to extract the action part correctly and gets decent score in both SS and BLEU score. In (C) (Source: **https://www.youtube.com/watch?v=uxEhH6MPH28{&}t=29s**), the output is perfect and SS Score is high. However, BLEU4 is 0.

Callison-Burch, Osborne & Koehn (2006); *Graham (2015)* demonstrate the inefficiency of BLEU scoring metric in generating a meaningful score for tasks like video captioning. A video may have multiple contexts. So, machines face difficulty to accurately measure the merit of the generated captions as there is no specific right answer. Therefore, for video captioning, it is more challenging to generate meaningful scores to reflect the captioning capability of the model. As a result, human evaluation is an important part to judge the effectiveness of the captioning model. In fact, Figs. 6, 7, 8 and 9, and 10 show this same fact that a higher BLEU score is not necessarily a good reflection of the captioning capability. On the other hand, our proposed human evaluation method portrays a better reflection of the model's performance compared to the BLEU scores.

Semantic sensibility(SS) score evaluation

To get a better understanding of captioning capability of our model, we perform qualitative analysis that is based on human evaluation similar to *Graham, Awad & Smeaton (2018)*,

(A) SSVC: "a woman is **slicing** *an onion*"
GT: "someone is cutting red bell peppers into thin slices"
SS Score:**0.5** , BLEU1: *0.83*, BLEU2: *0.70*, BLEU3: 0.63, BLEU4: **0.537**

(B) SSVC: "a man is **cutting** *a piece of meat*"
GT: "a man is peeling a carrot"
SS Score: **0.41** , BLEU1: *0.75*, BLEU2: **0.5**, b3:*0.0*, BLEU4: **0.0**

Figure 7 In (A) (**Source: https://www.youtube.com/watch?v=VahnQw2gTQY{&}t=298s**) and (B) (**Source: https://www.youtube.com/watch?v=YS1mzzhmWWA{&}t=9s**)**, our model is able to extract only the action part correctly.** The generated caption gets mediocre score in both SS and BLEU score.

(A) SSVC: "**a man** is *jumping.*"
GT: "a guy typing on a computer"
SS Score: **0.125** , BLEU1: *0.75*, BLEU2: *0.7*, BLEU3: *0.62*, BLEU4: **0.0**

(B) SSVC: "**a kitten** is *playing with a toy*"
GT: "a cat in a cage is angrily meowing at something"
SS Score: **0.375** , BLEU1: *0.57*, BLEU2: **0.30**, BLEU3: 0.0, BLEU4: 0.0

Figure 8 In (A) (**Source: https://www.youtube.com/watch?v=R2DvpPTfl-E{&}t=20s**) and (B) (**Source: https://www.youtube.com/watch?v=1hPxGmTGarM{&}t=9s**)**, the generated caption is completely wrong in case of actions, but BLEU1 gives a very high score.** On the contrary, SS Score heavily penalizes them.

(A) SSVC: "a man is **driving a car**"
GT: "a car drives backwards while trying to escape the police"
SS Score: **0.667** , BLEU1: **1.0**, BLEU2: **1.0**, BLEU3: *1.0*, BLEU4: *0.9*

Figure 9 In the figure, the black car is driving away while being chased by a police car (Source: **https://www.youtube.com/watch?v=3opDcpPxllE{&}t=50s**). **Our SSVC model only predicts the driving part.** Thus, the generated captions only partially capture the original idea. However, BLEU evaluates them with very high score where SS Score evaluates them accordingly.

(A) SSVC: "a man is *writing*"
GT: "someone is burning the tops of two cameras with a torch"
SS Score: **0.0** , BLEU1: *0.75*, BLEU2: **0.0**, BLEU3: **0.0**, BLEU4: **0.0**

(B) SSVC: "a woman is *pouring milk into a bowl*"
GT: "a powdered substance is being shifted into a pan"
SS Score: **0.0** , BLEU1: *0.625*, BLEU2: *0.52*, BLEU3: *0.36*, BLEU4: **0.0**

(C) SSVC: "*a man is mixing a pot of water*"
GT: "a fish is being fried in a pan"
SS Score: **0.0** , BLEU1: *0.625*, BLEU2: *0.42*, BLEU3: *0.31*, BLEU4: **0.0**

Figure 10 In (A) (Source: **https://www.youtube.com/watch?v=D1tTBncIsm8{&}t=841s**), (B) (Source: **https://www.youtube.com/watch?v=Cv5LsqKUXc{&}t=71s**), and (C) (Source: **https://www.youtube.com/watch?v=2FLsMPsywRc{&}t=45s**), **the generated caption is completely wrong, but BLEU1 gives a very high score where SS Score gives straight up zero.** Therefore, BLEU performs poorly here.

Xu et al. (2017) and *Pei et al. (2019)*. We propose a human evaluation metric, namely "Semantic Sensibility" score, for video captioning. It evaluates sentences at a contextual level from videos based on both recall and precision. It takes 3 factors into consideration. These are the grammatical structure of predicted sentences, detection of the most important

element (subject or object) in the videos and whether the captions give an exact or synonymous analogy to the action of the videos to describe the overall context.

It is to be noted that for the latter two factors, we take into consideration both the recall and precision values according to their general definition. In case of recall, we evaluate these 3 factors from our predicted captions and match them with the corresponding video samples. Similarly, for precision, we judge these factors from the video samples and match them with the corresponding predicted captions. Following such comparisons, each variable is assigned to a boolean value of 1 or 0 based on human judgment. The significance of the variables and how to assign their values are elaborated below:

$S_{grammar}$

$$S_{grammar} = \begin{cases} 1, & \text{if grammatically correct} \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

$S_{grammar}$ evaluates the correctness of grammar of the generated caption without considering the video.

$S_{element}$

$$S_{element} = \frac{\frac{1}{R}\sum_{i=1}^{R} S_{element_{recall}^i} + \frac{1}{P}\sum_{i=1}^{P} S_{element_{precision}^i}}{2} \quad (5)$$

where,

- R = number of prominent objects in video
- P = number of prominent objects in caption

As S_{action} evaluates the action-similarity between the predicted caption and its corresponding video, $S_{element}$ evaluates the object-similarity. For each object in the caption, the corresponding $S_{element_{precision}}$ receives a boolean score and for the major objects in the video, the corresponding $S_{element_{recall}}$ receives a boolean score. The average recall and average precision is combined to get the $S_{element}$.

S_{action}

$$S_{action} = \frac{S_{action_{recall}} + S_{action_{precision}}}{2} \quad (6)$$

S_{action} evaluates the ability to describe the action-similarity between the predicted caption and its corresponding video. $S_{action_{recall}}$ and $S_{action_{precision}}$ separately receives a boolean score (1 for correct, 0 for incorrect) for action recall and action precision respectively. By action recall, we determine if the generated caption has successfully captured the most prominent action of the video segment. Similarly, by action precision, we determine if the action mentioned in the generated caption is present in the video or not.

SS score calculation

Combining equations Eqs. (4), (5) and (6), the equation for the SS Score can be obtained.

$$SSScore = \frac{1}{N}\sum_{n=1}^{N}\left(S_{grammar} * \frac{S_{element} + S_{action}}{2}\right) \quad (7)$$

During this research work, the SS Score was calculated by 4 final-year undergraduate students studying at the Department of Computer Science and Engineering at the Islamic University of Technology. They are all Machine Learning researchers and are fluent English speakers. Each caption was separately scored by at least two annotators to make the scoring consistent and accurate.

RESULTS

Dataset and experimental setup

Our experiments are primarily centered around comparing our novel model with different commonly used architectures for video captioning like simple attention (*Gao et al., 2017*; *Wu et al., 2018*), modifications of attention mechanism (*Yang et al., 2018*; *Yan et al., 2019*; *Zhang et al., 2019*), variations of visual feature extraction techniques (*Aafaq et al., 2019a*; *Wang et al., 2018b*) etc that provide state-of-the-art results. We conducted the experiments under identical computational environment - Framework: *Tensorflow 2.0*, Platform: *Google Cloud Platform with a virtual machine having an 8-core processor and 30GB RAM*, GPU: *None*. We used the Microsoft-Research Video Description (MSVD) dataset (*Chen & Dolan, 2011*). It contains 1970 video snippets together with 40 English captions (*Chen, Li & Hu, 2020*) for each video. We split the entire dataset into training, validation, and test set with 1200, 100, and 670 snippets respectively following previous works (*Venugopalan et al., 2015*; *Pan et al., 2016*). To create a data-sequence, frames from a video are taken with a fixed temporal distance. We used 15 frames for each data-sequence. After creating the data-sequences, we had almost 65000 samples in our dataset. Though there is a large number of samples in the final dataset, the number of distinct trainable videos are only 1200. 1200 videos is not a large enough number. Having a larger dataset would be better for the training.

For the pre-trained embedding layer, we used 'glove.6B.100d' (*Pennington, Socher & Manning, 2014*). Due to lack of GPU, we used 256 LSTM units in each encoder layer and 512 LSTM units in our decoder network and trained each experimental model for 40 epochs. To analyse the importance of the Spatial Hard Pull layer, we also tuned the Spatial Hard Pull FC units from 0 to 45 and 60 successively.

One of the most prominent benchmarks for video data is TRECVid. Particularly, the 2018 TRECVid challenge *Awad et al. (2018)* that included video captioning, video information retrieval, activity detection, etc could be an excellent benchmark for our work. However, due to our limitations like lack of enough computational resources, rigidity in data pre-processing due to memory limitation and inability to train on a bigger dataset, we could not analyse our novel model with global benchmarks like TRECVid. On top of that, some of the benchmark models use multiple features as input to the model. However, we only use a single 2D based CNN feature as input as we wanted to make an extensive study on the capability of 2D CNN for video captioning. So, we implemented some of the fundamental concepts used in most state-of-the-art works on our experimental setup with single input 2D CNN feature. Thus, we performed ablation study to make a qualitative and quantitative analysis of our model. The performance of our two proposed novelties shows potential for improvement.

We used the BLEU score as one of the two main scoring criteria. To calculate BLEU on a dataset with multiple ground-truth captions, we used the Corpus BLEU calculation method (*Brownlee, 2019*). The BLEU scores reported throughout this paper actually indicates the Corpus BLEU Score. Our proposed architecture, SSVC, with 45 hard-pull units and 2 layer stacked attention gives the BLEU score of *"BLEU1": 0.7072, "BLEU2": 0.5193, "BLEU3": 0.3961, "BLEU4": 0.1886* after 40 epochs of training with the best combination of hyper-parameters. For generating the SS Score, we considered the first 50 random videos from the test set. We obtained an SS Score of 0.34 for the SSVC model.

Ablation study of stacked attention

- **No attention:** Many previous works (*Long, Gan & De Melo, 2018*; *Nam, Ha & Kim, 2017*) mentioned that captioning models perform better with some form of attention mechanism. Thus, in this paper, we avoid comparing use of attention and no attention mechanisms.
- **Non stacked (or single) attention:** In relevant literature, though the use of attention is very common, the use of stacked attention is quite infrequent. *Nam, Ha & Kim (2017)* have shown the use of stacked (or dual) attention and improvements of performance that are possible through it. In Fig. 4, the comparison between single attention and stacked attention indicates dual attention has clear edge over single attention.
- **Triple Attention:** Since the use of dual attention has improved performance in comparison to single attention, it is only evident to create a triple attention to check the performance. Figure 4 shows that triple attention under-performs in comparison to all other variants.

Considering our limitations, our stacked attention gives satisfactory results for both BLEU and SS Score in comparison to the commonly used attention methods when performed on similar experimental setup. Graphs in Fig. 4 suggest the same fact that our stacked attention improves the result of existing methods due to improved overall temporal information. Moreover, we can clearly see that the 2 layer LSTM encoder performs much better than single or triple layer encoder. Combining these two facts, we can conclude that, our dual encoder LSTM with stacked attention has the capability to improve corresponding architectures.

Ablation study of spatial hard pull

To boost the captioning capability, some state-of-the-art works like *Pan et al. (2017)* emphasized on the importance of retrieving additional visual information. We implemented the same fundamental idea in our model with the Spatial Hard Pull. To depict the effectiveness of our Spatial Hard Pull (SHP), we conducted experiments with our stacked attention as a constant and changed the SHP FC units with 0, 45 and 60 units successively. Figure 5 shows that as the number of SHP FC units are increased from 0 to 45, both BLEU and SS Score get better and again gradually falls from 45 to 60. The performance improvement in the early stages indicate that SHP layer is indeed improving the model. The reason for fall of scores in the later stages is that the model starts to show high variance.

Hence it is evident from this analysis that our approach of using SHP layer yields satisfactory result compared to not using any SHP layer.

DISCUSSION

By performing various trials on a fixed experimental setting, we analysed the spatio-temporal behaviour of a video captioning model. After seeing that single layer encoder LSTM causes more repetitive predictions, we used double and triple layer LSTM encoder to encode the visual information into a better sequence. Hence, we were able to propose our novel stacked attention mechanism with double encoder layer that performs the best among all the variations of LSTM layers that we tried. The intuition behind this mechanism is that, as our model separately gives attention to each encoder layer, this generates a better overall temporal context for it to decode the video sequence and decide whether to give more priority to the object or the action. And the addition of Spatial Hard Pull to this model bolsters its ability to identify and map high level semantic visual information. Moreover, the results also indicate that addition of excess SHP units drastically affect the performance of the model. Hence, a balance is to be maintained while increasing the SHP units so that the model does not over-fit. As a result, both of these key components of our novel model greatly contributed to improving the overall final performance of our novel architecture, that is based upon the existing fundamental concepts of state of art models.

Although the model performed good in qualitative and quantitative analysis, our proposed SS Scoring method provides greater insight to analyse video captioning models. The auto metrics, although useful, cannot interpret the videos correctly. In our experimental results, we can see a steep rise in the BLEU Score in Figs. 4 and 5 at early epochs even though the predicted captions are not up to the mark. These suggest the limitations of BLEU score in judging the captions properly with a meaningful score. SS Score considers these limitations and finds a good semantic relationship between the context of the videos and the generated language that portrays the video interpreting capability of a model into language to its truest sense. Hence, we can safely evaluate the captioning capability of our Stacked Attention with Spatial Hard Pull mechanism to better understand the acceptability of the performance of our novel model.

CONCLUSION AND FUTURE WORK

Video captioning is a complex task. This paper shows how stacking the attention layer for a multi-layer encoder makes a more semantically accurate context. Complementing it, the Sparse Sequential Join, introduced in this paper, is able to capture the higher level features with greater efficiency.

Due to our computational limitations, our experiments use custom pre-processing and constrained training environment. We also use a single feature as input unlike most of the state-of-the-art models. Therefore, the scores we obtained in our experiments are not comparable to global benchmarks. In future, we hope to perform similar experiments with industry standard pre-processing with multiple features as input.

The paper also introduces the novel SS Score. This deterministic scoring metric has shown great promise in calculating the semantic sensibility of a generated video-caption. However, since it is a human evaluation metric, it relies heavily on human understanding. Thus, a lot of manual work is to be put behind it. For the grammar score, we can use *Naber (2003)*'s "A Rule-Based Style and Grammar Checker" technique. This will partially automate the SS Scoring method.

Funding
The authors received no funding for this work.

Competing Interests
The authors declare there are no competing interests.

Author Contributions
- Md. Mushfiqur Rahman conceived and designed the experiments, performed the experiments, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.
- Thasin Abedin conceived and designed the experiments, performed the experiments, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.
- Khondokar S.S. Prottoy and Ayana Moshruba analyzed the data, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.
- Fazlul Hasan Siddiqui conceived and designed the experiments, authored or reviewed drafts of the paper, and approved the final draft.

Data Availability
The code is available at GitHub: https://github.com/mushfiqur11/SS-VideoCaptioning. Data tabulation is available in the Supplemental File.

Supplemental Information
Supplemental information for this article can be found online at http://dx.doi.org/10.7717/peerj-cs.664#supplemental-information.

REFERENCES

Aafaq N, Akhtar N, Liu W, Gilani SZ, Mian A. 2019a. Spatio-temporal dynamics and semantic attribute enriched visual encoding for video captioning. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Piscataway: IEEE, 12487–12496.

Aafaq N, Mian A, Liu W, Gilani SZ, Shah M. 2019b. Video description: a survey of methods, datasets, and evaluation metrics. *ACM Computing Surveys (CSUR)* **52(6)**:1–37.

Amaresh M, Chitrakala S. 2019. Video captioning using deep learning: an overview of methods, datasets and metrics. In: *2019 International Conference on Communication and Signal Processing (ICCSP)*. Piscataway: IEEE, 0656–0661.

Awad G, Butt A, Curtis K, Lee Y, Fiscus J, Godil A, Joy D, Delgado A, Smeaton A, Graham Y. 2018. Trecvid 2018: benchmarking video activity detection, video captioning and matching, video storytelling linking and video search. In: *Proceedings of TRECVID 2018*.

Bahdanau D, Cho K, Bengio Y. 2014. Neural machine translation by jointly learning to align and translate. ArXiv preprint. arXiv:1409.0473.

Ballas N, Yao L, Pal C, Courville AC. 2016. Delving deeper into convolutional networks for learning video representations. In: Bengio Y, LeCun Y eds. *4th international conference on learning representations, ICLR 2016, San Juan, Puerto Rico, (2016) 2-4, Conference track proceedings*.

Banerjee S, Lavie A. 2005. Meteor: an automatic metric for MT evaluation with improved correlation with human judgments. In: *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*. 65–72.

Baraldi L, Grana C, Cucchiara R. 2017. Hierarchical boundary-aware neural encoder for video captioning. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Piscataway: IEEE, 1657–1666.

Bin Y, Yang Y, Shen F, Xie N, Shen HT, Li X. 2018. Describing video with attention-based bidirectional LSTM. *IEEE Transactions on Cybernetics* **49(7)**:2631–2641.

Brownlee J. 2019. A gentle introduction to calculating the BLEU score for text in python. Available at *https://machinelearningmastery.com/calculate-bleu-score-for-text-python/*.

Callison-Burch C, Osborne M, Koehn P. 2006. Re-evaluation the role of BLEU in machine translation research. In: *11th conference of the european chapter of the association for computational linguistics*.

Chang S-F, Hsu WH, Kennedy LS, Xie L, Yanagawa A, Zavesky E, Zhang D-Q. 2005. Columbia university TRECVID-2005 video search and high-level feature extraction. In: *TRECVID*. 0–06.

Chen D, Dolan WB. 2011. Collecting highly parallel data for paraphrase evaluation. In: *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*. 190–200.

Chen H, Li J, Hu X. 2020. Delving deeper into the decoder for video captioning. ArXiv preprint. arXiv:2001.05614.

Divakaran A, Sun H, Ito H. 2003. Methods of feature extraction of video sequences. US Patent 6, 618, 507.

Ekin A, Tekalp AM, Mehrotra R. 2003. Automatic soccer video analysis and summarization. *IEEE Transactions on Image Processing* **12(7)**:796–807 DOI 10.1109/TIP.2003.812758.

Ekin A, Tekalp M. 2003. Generic play-break event detection for summarization and hierarchical sports video analysis. In: *2003 international conference on multimedia and Expo. ICME'03. Proceedings (Cat. No. 03TH8698), volume 1*. Piscataway: IEEE, I–169.

Gao L, Guo Z, Zhang H, Xu X, Shen HT. 2017. Video captioning with attention-based LSTM and semantic consistency. *IEEE Transactions on Multimedia* **19(9)**:2045–2055 DOI 10.1109/TMM.2017.2729019.

Gers FA, Schmidhuber JA, Cummins FA. 2000. Learning to forget: continual prediction with LSTM. *Neural Computation* **12(10)**:2451–2471 DOI 10.1162/089976600300015015.

Graham Y. 2015. Re-evaluating automatic summarization with BLEU and 192 shades of rouge. In: *Proceedings of the 2015 conference on empirical methods in natural language processing*. 128–137.

Graham Y, Awad G, Smeaton A. 2018. Evaluation of automatic video captioning using direct assessment. *PLOS ONE* **13(9)**:e0202789 DOI 10.1371/journal.pone.0202789.

Kantorov V, Laptev I. 2014. Efficient feature extraction, encoding and classification for action recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Piscataway: IEEE, 2593–2600.

Kilickaya M, Erdem A, Ikizler-Cinbis N, Erdem E. 2016. Re-evaluating automatic metrics for image captioning. ArXiv preprint. arXiv:1612.07600.

Kojima A, Tamura T, Fukunaga K. 2002. Natural language description of human activities from video images based on concept hierarchy of actions. *International Journal of Computer Vision* **50(2)**:171–184 DOI 10.1023/A:1020346032608.

Lamb AM, Goyal AGAP, Zhang Y, Zhang S, Courville AC, Bengio Y. 2016. Professor forcing: a new algorithm for training recurrent networks. In: *29th Conference on Neural Information Processing Systems (NIPS 2016)*. Barcelona, Spain: Advances in neural information processing systems, 4601–4609.

Li B, Sezan MI. 2001. Event detection and summarization in sports video. In: *Proceedings IEEE workshop on content-based access of image and video libraries (CBAIVL 2001)*. Piscataway: IEEE, 132–138.

Lin C-Y. 2004. Rouge: a package for automatic evaluation of summaries. In: *Text summarization branches out*. Barcelona, Spain Publisher: Association for Computational Linguistics, 74–81.

Long X, Gan C, De Melo G. 2018. Video captioning with multi-faceted attention. *Transactions of the Association for Computational Linguistics* **6**:173–184 DOI 10.1162/tacl_a_00013.

Luong M-T, Pham H, Manning CD. 2015. Effective approaches to attention-based neural machine translation. ArXiv preprint. arXiv:1508.04025.

Naber D. 2003. A rule-based style and grammar checker. Avaialble at *http://www.danielnaber. de/languagetool/download/style_and_grammar_checker.pdf*.

Nam H, Ha J-W, Kim J. 2017. Dual attention networks for multimodal reasoning and matching. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Piscataway: IEEE, 299–307.

Oostveen J, Kalker T, Haitsma J. 2002. Feature extraction and a database strategy for video fingerprinting. In: Chang SK, Chen Z, Lee SY, eds. *Recent Advances in Visual Information Systems. VISUAL 2002. Lecture Notes in Computer Science, vol 2314*. Berlin, Heidelberg: Springer DOI 10.1007/3-540-45925-1_11.

Pan Y, Mei T, Yao T, Li H, Rui Y. 2016. Jointly modeling embedding and translation to bridge video and language. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Piscataway: IEEE, 4594–4602.

Pan Y, Yao T, Li H, Mei T. 2017. Video captioning with transferred semantic attributes. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Piscataway: IEEE, 6504–6512.

Papineni K, Roukos S, Ward T, Zhu W-J. 2002. BLEU: a method for automatic evaluation of machine translation. In: *Proceedings of the 40th annual meeting of the association for computational linguistics*. 311–318.

Pei W, Zhang J, Wang X, Ke L, Shen X, Tai Y-W. 2019. Memory-attended recurrent network for video captioning. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Piscataway: IEEE, 8347–8356.

Pennington J, Socher R, Manning CD. 2014. Glove: global vectors for word representation. In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 1532–1543.

Post M. 2018. A call for clarity in reporting BLEU scores. ArXiv preprint. arXiv:1804.08771.

Quellec G, Cazuguel G, Cochener B, Lamard M. 2017. Multiple-instance learning for medical image and video analysis. *IEEE Reviews in Biomedical Engineering* **10**:213–234 DOI 10.1109/RBME.2017.2651164.

Rohrbach M, Qiu W, Titov I, Thater S, Pinkal M, Schiele B. 2013. Translating video content to natural language descriptions. In: *Proceedings of the IEEE international conference on computer vision*. Piscataway: IEEE, 433–440.

Shao L, Gouws S, Britz D, Goldie A, Strope B, Kurzweil R. 2017. Generating high-quality and informative conversation responses with sequence-to-sequence models. ArXiv preprint. arXiv:1701.03185.

Shih H-C. 2017. A survey of content-aware video analysis for sports. *IEEE Transactions on Circuits and Systems for Video Technology* **28(5)**:1212–1231.

Simonyan K, Zisserman A. 2014. *Very deep convolutional networks for large-scale image recognition*. ArXiv preprint. arXiv:1409.1556.

Song J, Guo Z, Gao L, Liu W, Zhang D, Shen HT. 2017. Hierarchical LSTM with adjusted temporal attention for video captioning. ArXiv preprint. arXiv:1706.01231.

Sutskever I, Vinyals O, Le QV. 2014. Sequence to sequence learning with neural networks. In: *27th Conference on Neural Information Processing Systems (NIPS 2014)*. Montreal, Quebec, Canada: Advances in neural information systems, 3104–3112.

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I. 2017. Attention is all you need. In: *30th Conference on Neural Information Processing Systems (NIPS 2017)*. Long Beach, California, USA: Advances in neural information processing systems, 5998–6008.

Venugopalan S, Rohrbach M, Donahue J, Mooney R, Darrell T, Saenko K. 2015. Sequence to sequence-video to text. In: *Proceedings of the IEEE international conference on computer vision*. 4534–4542.

Wang B, Ma L, Zhang W, Liu W. 2018a. Reconstruction network for video captioning. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Piscataway: IEEE, 7622–7631.

Wang J, Wang W, Huang Y, Wang L, Tan T. 2018b. M3: multimodal memory modelling for video captioning. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2018*. Piscataway: IEEE, 7512–7520.

Weiss RJ, Chorowski J, Jaitly N, Wu Y, Chen Z. 2017. Sequence-to-sequence models can directly translate foreign speech. ArXiv preprint. arXiv:1703.08581.

Wu X, Li G, Cao Q, Ji Q, Lin L. 2018. Interpretable video captioning via trajectory structured localization. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Piscataway: IEEE, 6829–6837.

Xingjian S, Chen Z, Wang H, Yeung D-Y, Wong W-K, Woo W-c. 2015. Convolutional LSTM network: a machine learning approach for precipitation nowcasting. In: *28th Conference on Neural Information Processing Systems (NIPS 2015)*. Montreal, Quebec, Canada: Advances in neural information systems, 802–810.

Xu J, Yao T, Zhang Y, Mei T. 2017. Learning multimodal attention LSTM networks for video captioning. In: *Proceedings of the 25th ACM international conference on multimedia*. New York: ACM, 537–545.

Xu R, Xiong C, Chen W, Corso JJ. 2015. Jointly modeling deep video and compositional text to bridge vision and language in a unified framework. In: *Twenty-Ninth AAAI conference on artificial intelligence*.

Yan C, Tu Y, Wang X, Zhang Y, Hao X, Zhang Y, Dai Q. 2019. Stat: spatial–temporal attention mechanism for video captioning. *IEEE Transactions on Multimedia* **22(1)**:229–241.

Yang Y, Zhou J, Ai J, Bin Y, Hanjalic A, Shen HT, Ji Y. 2018. Video captioning by adversarial LSTM. *IEEE Transactions on Image Processing* **27(11)**:5600–5611 DOI 10.1109/TIP.2018.2855422.

Zhang Z, Shi Y, Yuan C, Li B, Wang P, Hu W, Zha ZJ. 2020. Object relational graph with teacher-recommended learning for video captioning. In: *2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*. 13275–13285.

Zhang Z, Xu D, Ouyang W, Tan C. 2019. Show, tell and summarize: dense video captioning using visual cue aided sentence summarization. *IEEE Transactions on Circuits and Systems for Video Technology* **30**:3130–3139.