Article

Video Captioning Based on Channel Soft Attention and Semantic Reconstructor

Zhou Lei and Yiyong Huang *

School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China; leiz@shu.edu.cn

* Correspondence: huangyiyong@shu.edu.cn; Tel.: +86-13957717796

Abstract: Video captioning is a popular task which automatically generates a natural-language sentence to describe video content. Previous video captioning works mainly use the encoder–decoder framework and exploit special techniques such as attention mechanisms to improve the quality of generated sentences. In addition, most attention mechanisms focus on global features and spatial features. However, global features are usually fully connected features. Recurrent convolution networks (RCNs) receive 3-dimensional features as input at each time step, but the temporal structure of each channel at each time step has been ignored, which provide temporal relation information of each channel. In this paper, a video captioning model based on channel soft attention and semantic reconstructor is proposed, which considers the global information for each channel. In a video feature map sequence, the same channel of every time step is generated by the same convolutional kernel. We selectively collect the features generated by each convolutional kernel and then input the weighted sum of each channel to RCN at each time step to encode video representation. Furthermore, a semantic reconstructor is proposed to rebuild semantic vectors to ensure the integrity of semantic information in the training process, which takes advantage of both forward (semantic to sentence) and backward (sentence to semantic) flows. Experimental results on popular datasets MSVD and MSR-VTT demonstrate the effectiveness and feasibility of our model.

Citation: Lei, Z.; Huang, Y. Video Captioning Based on Channel Soft Attention and Semantic Reconstructor. *Future Internet* **2021**, *13*, 55. <https://doi.org/10.3390/fi13020055>

Academic Editor: Rafael Valencia-Garcia

Received: 15 January 2021
Accepted: 19 February 2021
Published: 23 February 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: video captioning; channel soft attention; semantic reconstructor; recurrent convolution networks

1. Introduction

Video captioning is a popular and challenging task. It involves both computer vision and natural-language processing. Automatic video caption generation has many practical applications. For example, it could help improve the quality of online video indexing and searching. As another example, in combination with speech synthesis technology, describe videos with natural language could help the visually impaired to understand video contents.

The most important part of video captioning is to extract a precise video representation. As the video is made up of continuous images, which is a spatial-temporal extension of the pixel, how to obtain video representation with spatial-temporal information is a crucial part of video captioning. Early methods such as Dense Trajectories (DT) [1] and Improved Dense Trajectories (iDT) [2], the spatial-temporal descriptors are first made by handcrafting and then encoded through Vector of Locally Aggregated Descriptors (VLAD) [3] or Fisher Vector [4] to form the final video representation. Deep learned descriptors have shown their strong competitiveness in video tasks after image classification made a great improvement [5]. Xu et al. [6] propose a framework Sequential VLAD (SeqVLAD), in which the descriptor encoding is learnable, and the temporal information can be detailly aggregated by using an improved recurrent convolutional network (RCNs). Although their framework has achieved great success in video feature aggregation, it has ignored the temporal structure of the feature map of each channel, which provides global nuances of each channel.

Convolutional block attention module (CBAM) [7] exploits channel-wise attention and soft attention [8] uses global information at each encoding time step. Inspired by CBAM and soft attention, we propose a channel soft attention to exploit the temporal structure for each channel of feature maps. We replace the linear multiplication with the convolution operation, combine it with the global video feature maps, and then use channel attention to compute the channel weight of each frame's feature maps. The weighted feature maps are fed into parameter shared gated recurrent convolutional units (SGRU-RCN, a variant of gated recurrent convolutional units) [6] to learn spatial-temporal assignment parameters, which could be taken as the weights for aggregating local descriptors to certain cluster centers. The proposed channel soft attention can be easily implemented in different convolutional recurrent neural networks.

Recent works have shown that visual captioning tasks can be improved by high-level semantic attributes of image or video [9,10]. With well-aggregated video features, we enrich the captioning model with semantic attributes using semantic compositional network LSTM (SCN-LSTM) [11]. First, a semantic detection network (SDN) is trained to generate tags for videos. To involve the most frequent nouns, verbs, and adjectives, we select K of most common words from ground-truth captions of the training set as the vocabulary of tags. The tags generated by SDN are combined with video features and fed into the SCN-LSTM decoder to create an accurate description of the video. The tags are represented as a one-dimension vector and are a significantly smaller size than video features. Thus, the use of tags has less impact on the model. During the training, the semantic information might be ignored or lost because of parameter calculating. To train the parameters to make full use of semantic information, the semantic vector is rebuilt by a semantic reconstructor using the output of the decoder. When the semantic information is fully used, the semantic vector will be more complete during the sentence generating, and the reconstructed semantic vector will have less difference from the original semantic vector. The dual learning method of semantic reconstructor has great feasibility in other tasks which include semantic information such as image caption and visual question answering.

As mentioned above, a video captioning model based on channel soft attention and semantic reconstructor (CSA-SR) is proposed. It aims to fully use the feature maps of videos and the semantic information of videos. Our code is available at <https://github.com/YiyongHuang/CSA-SR> accessed on 14 January 2021. The main contributions of this paper are summarized as follows:

- A deep learning framework for video captioning is proposed, which optimizes the Sequential VLAD encoding by channel soft attention and leverages the semantic information by a semantic reconstructor.
- An effective attention module is implemented, which exploits the temporal structure for each channel, where the c -th channel of input feature maps at t -th time step is the weighted sum of c -th channel of all feature maps.
- A semantic reconstructor network is proposed which is composed of LSTM to assist the decoder to fully use the semantic information.
- Experiment results on Microsoft Video Description (MSVD) dataset [12] and MSR-Video To Text (MSR-VTT) dataset [13] demonstrate the feasibility and effectiveness of the proposed method.

2. Related Work

The early visual captioning methods normally detect Subject, Verb, and Object separately using a rule-based system [14,15], then generate a sentence by joining them together. However, Vinyals et al. [16] make a breakthrough in image captioning by using deep learning methods. They generate natural languages directly from image features by using a recurrent neural network (RNN). Subsequently, video captioning based on deep learning becomes a topic of significant research interests. Compare with images, videos are more complex and diverse, which imposing great challenges for understanding video content. Donahue et al. [17] propose the long-term recurrent convolutional networks (LRCN), which

is the first video captioning model based on deep learning. It extracts the video features of each frame through the pretrained model for image classification and then inputs the features into an LSTM network after average pooling for decoding to obtain the sentences. The model has been widely adopted as a benchmark model for video captioning. Since then, many innovations and improvements have been reported on the model.

Sequence Learning Approaches: Venugopalan et al. [18] first propose an end-to-end model for generating video captions. Their model takes one frame of every 10 frames of video, extracts the output features of the full connection layer through a convolution neural network pretrained with ImageNet [19], and averages the output of all the frames of the video to get a 4096-dimensional vector, which is then used as the input of the first layer of LSTM at every time step. The output of the LSTM decoder obtains the probability distribution of each word in the vocabulary through SoftMax. Since the Venugopalan method for image features uses the average pooling, it loses the temporal information in a video. Venugopalan proposes S2VT [20], which uses LSTMs as both encoder and decoder, and incorporate with the optical flow to obtain the temporal information. To address the better temporal dynamics of the video sequence adequately, Yao et al. [21] introduce an attention mechanism to assign weights to the features of each frame and then fuse them based on the attentive weights. As for semantics, the relevance between video context and sentence semantics is considered to be a regularizer in the LSTM [22]. Pan et al. [23] employ high-level semantic attributes as a supplementary representation for caption generation by using a transfer unit. Gan et al. [11] train a semantic detection network by using a multi-label classification method to generate semantic tags and embeds the semantic information by a semantic compositional network to generate sentences. Wang et al. [24] propose a reconstructor for video captioning, a dual learning mechanism that reconstructs source from the target by a backward flow. To obtain finer motion information among successive video frames, Xu et al. [6] conduct feature clustering by parameter sharing recurrent convolutional network and then adopt an end-to-end and sequence-to-sequence framework for encoding and decoding, which shows good performance. Due to the temporal characteristic of video and language, sequence learning is always an efficient method. In this paper, the encode-decode module based on sequence video VLAD and semantic reconstructor can be both regarded as a sequence learning method.

Attention Mechanism: Selectivity is an important characteristic of the neural mechanism of the human brain, which represents the ability to filter out unimportant information, and could be called attention mechanism. Recently, many attention methods have been proposed to improve algorithm performance in various tasks. Luong et al. [8] propose global attention and local attention for neural machine translation. Global attention is also called soft attention in other research. Yao et al. [21] apply the soft attention mechanism in the video captioning task and it has shown a good performance. Stollenga et al. [25] introduce a deep attention selective network to solve image classification tasks. Chorowski et al. [26] explore attention-based recurrent networks for speech recognition. Pan et al. [27] make a great improvement in the image caption task by X-Linear Attention. A recent work of Hu et al. [28] designs the squeeze-and-excitation module (SEnet) to exploit the inter-channel relationship. Based on SEnet, Woo et al. [7] exploit both spatial and channel-wise attention. In addition, introduce the convolutional block attention module (CBAM), which can be used as a plug-and-play module for pre-existing CNN-based architectures. These two modules [7,28] have shown the great importance of channel attention. However, due to videos having a temporal structure, the global structure of the channel has not been taken seriously. We combine the advantages of channel attention and soft attention to exploit the global structure for each channel.

Based on these previous works above, the model proposed in this paper considers the attention mechanism comprehensively from temporal and channel. Furthermore, the semantic concepts are embedded in the decoder and the semantic vector is reconstructed to fully use the semantic information. This can be seen as a novel loss for generating accurate captions.

3. Method

In this paper, a channel soft attention improved sequential video VLAD (SeqVLAD) [6] with a semantic reconstruction network (CSA-SR) for video captioning is proposed. The overall structure is shown in Figure 1. The proposed CSA-SR mainly contains three parts, including encoder, decoder, and semantic reconstructor. The encoder extracts video features and encodes features for the decoder. Frame-level features are extracted from video by using 2D convolutional neural networks, and we also employ 3D convolutional neural networks to extract video features from video clips. The fine motion information among successive video frames is encoded by our proposed channel soft attention and SeqVLAD. In addition, the features extracted from 2D and 3D convolutional neural networks are both used in the semantic detection network (SDN). As for decoder, the semantic LSTM is used to combine the video representation and semantic vectors to generate sentences. The reconstructor will rebuild semantic vectors. By fully using the bidirectional flows of semantic vector and sentence, the performance of video captioning can be further boosted. In the following part, we briefly introduce the VLAD [3] and SeqVLAD first. Then present the details of the channel soft attention and semantic reconstruction network of our model.

Figure 1. The overall structure of the model based on channel soft attention and semantic reconstructor.

3.1. VLAD and SeqVLAD

The VLAD can be seen as a kind of simplified Fisher Vector. The Fisher vector encoding uses GMM (Gaussian Mixture Model) to construct visual descriptors. In addition, the VLAD does not store second-order information about the features and typically use K-Means instead of GMMs to generate descriptors. The VLAD calculates the differences between the local descriptor and its nearest cluster center, then aggregates the local descriptors into a fixed dimension vector. Suppose we employ K-means to generate K centers $C = \{c_1, c_2, \dots, c_K\}$. In addition, the local descriptors with dimension D are $X = \{x_1, x_2, \dots, x_N\}$, the VLAD encoding can be expressed by the following formulation:

$$v_k = \sum_{n=1}^N a_k(x_n)(x_n - c_k) \quad (1)$$

where N is the number of descriptors, x_n is the n -th descriptor, c_k is the k -th cluster center, and the assignment $a_k(x_n)$ represents the relationship between current descriptor with k -th cluster center. If c_k is the nearest cluster center to current descriptor, $a_k(x_n) = 1$, and otherwise, $a_k(x_n) = 0$. VLAD calculates v_k over all cluster centers as $V = \{v_1, v_2, \dots, v_k\}$. With the dimension D of each center, we can compute the final representation with dimension

$K \times D$. The VLAD performs well in image retrieval. However, because assignment $a_k(x_n)$ is discontinuous and non-differentiable, how to train VLAD in network is a difficult problem.

The purpose of SeqVLAD is to plug VLAD into a deep neural network to aggregate the video representation locally and temporally. SeqVLAD tries to leverage the benefits of both convolutional units and the VLAD encoding method in RNNs architecture, using parameter shared gated recurrent convolutional units (SGRU-RCN) as the assignment of VLAD encoding. Recurrent convolutional networks (RCNs) successfully combine CNNs and RNNs which could obtain the video features in both spatial and temporal for video processing tasks. It has been shown that RNN focuses more on global feature changes, discards the fine motion information between successive video frames. To get more detailed information, the recurrent unit named gated recurrent convolutional units (GRU-RCN) which replace the fully connected operation with convolution operation, inputs a 3-dimensional feature map and outputs learned feature map with the same dimension. To reduce the number of parameters and the chance of overfitting, SGRU-RCN shared the input-to-hidden convolutional kernels. An SGRU-RCN unit for VLAD encoding is defined as follows:

$$q_t = W_q * x_t, \quad (2)$$

$$z_t = \sigma(q_t + U_z * a_{t-1}), \quad (3)$$

$$r_t = \sigma(q_t + U_r * a_{t-1}), \quad (4)$$

$$\tilde{a}_t = \tanh(q_t + U * (r_t \odot a_{t-1})), \quad (5)$$

$$a_t = (1 - z_t)a_{t-1} + z_t\tilde{a}_t \quad (6)$$

where $*$ denotes a convolution operation, σ is a *sigmoid* function, and $\odot$ is an element-wise multiplication. The input x_t at time step t is a 3-dimensional feature map with shape $C \times H \times W$, where C , H and W denote the number of channels, height, and width of the feature maps, respectively. The shape of W_q is $K \times C \times 1 \times 1$. W_q is the shared convolutional kernel for z_t , r_t , and $\tilde{a}_t$, where z_t is an update gate indicating whether previous information needs to be updated, and r_t is a reset gate indicating whether previous information needs to be reset. Parameters U_z , U_r , U are 2-dimensional convolutional kernels. The output of SGRU-RCN a_t with shape $K \times H \times W$ can be taken as the weights of spatial and temporal assignment for VLAD encoding. a_t can be viewed as $a_t^k(i, j)$, denoting the assignment weights of aggregating the descriptor at location (i, j) of t -th frame to the k -th cluster center. Because the range of assignment values should be 0 to 1, the output $a_t^k(i, j)$ is wrapped by a *softmax* function, which can be described as:

$$\text{softmax}(a_t^k(i, j)) = \frac{e^{a_t^k(i, j)}}{\sum_{n=1}^K e^{a_t^n(i, j)}} \quad (7)$$

where K denotes the number of total cluster centers. The SeqVLAD encoding would be:

$$v_k = \sum_{t=1}^T \sum_{j=1}^H \sum_{i=1}^W a_t^k(i, j)(x_t(i, j) - c_k) \quad (8)$$

where $x_t(i, j)$ represents the descriptor at location (i, j) . In addition, T represents the input sequence length. Referring to Equation (8), it accumulates the differences between descriptors and cluster centers locally and temporally. The output of SeqVLAD concatenates v_k over K cluster centers and the final representation with dimension $K \times D$ is the same as VLAD.

3.2. Channel Soft Attention

The input of SeqVLAD at each time step is a 3-dimensional feature map, which is the output of effective models pretrained on ImageNet [19]. If SeqVLAD receives the feature maps of one frame at each time step, it would lose the ability to exploit the global

temporal structure of the video. The features and information in the later frames of a video are associated with the previous frames. Inspired by soft attention applied in natural-language processing successfully, we propose a channel soft attention to exploit the temporal structure for 3-dimensional image feature maps. As shown in Figure 2, in a sequence of video features, the shape of feature maps at each time step is $C \times H \times W$. Each block in Figure 2 with different colors denotes one feature map with shape $H \times W$. The feature map at i -th channel at each time step is generated by the same convolution kernel. Our approach is to exploit the temporal structure for each channel. Instead of taking each feature map of one frame at each time step, we take the dynamic weighted sum of the temporal feature maps for each channel. The input of SeqVLAD at each time step is defined as:

$$\varphi_t(V_c) = \sum_{i=1}^N A_{c,i}^t v_{c,i} \quad (9)$$

where $\sum_{i=1}^N A_{c,i}^t = 1$ and $A_{c,i}^t$ is computed at each time step t inside the SGRU-RCN encoder, it denotes the attention weight of i -th frame's channel c at time step t . The value of attention weight reflects the relevance of the i -th temporal feature of each channel in the input video given all the previously generated assignments. $v_{c,i}$ denotes channel c 's feature map of i -th frame. Concatenating all the $\varphi_t(V_c)$ can get the feature maps with shape $C \times H \times W$, which is the same as the original shape of feature maps of one frame. We obtain the attention weight $A_{c,i}^t$ by normalizing the relevance scores $e_{c,i}^t$:

$$A_{c,i}^t = \frac{e_{c,i}^t}{\sum_{j=1}^N e_{c,j}^t} \quad (10)$$

Figure 2. An illustration of the weighted feature maps for each channel at each time step.

We design a function that takes the previous assignment a_{t-1} of the SGRU-RCN as input and summarizes all the previously generated assignments and the feature maps of the i -th temporal feature. The function gives the unnormalized relevance score e_i^t :

$$e_i^t = CA(\text{relu}(W_a * a_{t-1} + U_a * v + b_a)), \quad (11)$$

$$CA(f) = \tanh(\text{MLP}(\text{MaxPool}(f)) + \text{MLP}(\text{AvgPool}(f))) \quad (12)$$

where W_a, U_a are the convolutional kernels and b_a are the parameters, those are trained together in the encoder–decoder networks, v is the input feature maps of the entire video sequence. The relevance score e_i^t with shape $N \times C$ involves the score of each channel at

each time step, and $CA(\cdot)$ is a channel attention module. To compute the 2-dimensional relevance score with shape $N \times C \times 1 \times 1$, we use the channel attention module in CBAM [7], and make a small change as follows. Consider that *sigmoid* ranges from 0 to 1 and *tanh* ranges from -1 to 1. To expand the differentiation of each weight in a controllable range, the activate function *sigmoid* is replaced by *tanh*. e_i^t with C channels is the relevance score of i -th frame at t -th time step. It can be viewed as $e_{c,i}^t$ to represent the score for each channel.

3.3. Semantic Reconstruction Network

A convolutional GRU can get rich temporal and spatial information of a video. However, if we want to generate semantically accurate sentences, semantic information is needed to enrich the features that get from the SeqVLAD encoder. In our approach, the semantic detection network (SDN) is used for generating the semantic vectors and semantic compositional network (SCN, a variant of LSTM) is took as the decoder. Moreover, a dual learning mechanism is adopted to maintain the information of the semantic vector during training, and the semantic information in forward flow can be better used.

For semantic detection, the candidate semantic representation should be obtained first, there are some state-of-art methods such as keyword-based representation [29] and semantic-based representation [30]. Our ground-truth labels are some short sentences, and those sentences of a video might be composed of different words. Moreover, the vocabulary does not contain huge amounts of words, some videos cannot be contained by semantic representation if using semantic represent methods. To avoid the influence of semantic represents methods, we built the semantic representation tags by the manual selection which are K most common words in the training set. Manual selection is not feasible if the dataset is extremely large, then the semantic representation tags should build by semantic represent methods such as TF-IDF [29]. The task of semantic detection can be seen as a multi-label classification task. For more accurate classification, the fully connected features of the 3D convolutional network and 2D convolutional network are extracted, then concatenate these two features as the input of SDN. Suppose the representation of i -th video is F_i , then, the ground-truth is $y_i = \{y_{i,1}, y_{i,2}, \dots, y_{i,K}\} \in \{0, 1\}^K$. If the j -th tag exists in the i -th video's annotations, then $y_{i,j} = 1$, otherwise $y_{i,j} = 0$. A semantic detection network can be implemented by a deep multi-layer perceptron (MLP), which can be defined as:

$$s_i = \sigma(MLP(F_i)) \quad (13)$$

where σ is the *sigmoid* activation function, and s_i is the semantic vector of i -th video. The loss function of SDN is defined as:

$$Loss_{SDN}(s, y) = \frac{1}{N} \sum_{i=1}^N \sum_{j=1}^K (y_{ij} \log(s_{ij}) + (1 - y_{ij}) \log(1 - s_{ij})) \quad (14)$$

As for decoder, SCN-LSTM could avoid the problem of long-term dependency, and embeds the semantic attributes of the input video. Suppose there are encoded feature x_t at time step t , the hidden state h_{t-1} at time step $t - 1$ and semantic feature s .

$$\tilde{x}_{z,t} = W_{z,a}(W_{z,b}s \odot W_{z,c}x_t), z \in \{c, i, f, o\}, \quad (15)$$

$$\tilde{h}_{z,t-1} = U_{z,a}(U_{z,b}s \odot U_{z,c}h_{t-1}), z \in \{c, i, f, o\} \quad (16)$$

where c denotes the cell state, i denotes the input gate, f denotes the forget gate, and o denotes the output gate. w and U are learnable weight matrices. The SCN-LSTM can be described as:

$$i_t = \sigma(\tilde{x}_{i,t} + \tilde{h}_{i,t-1} + b_i), \quad (17)$$

$$f_t = \sigma(\tilde{x}_{f,t} + \tilde{h}_{f,t-1} + b_f), \quad (18)$$

$$o_t = \sigma(\tilde{x}_{o,t} + \tilde{h}_{o,t-1} + b_o), \quad (19)$$

$$\tilde{c}_t = \sigma(\tilde{x}_{c,t} + \tilde{h}_{c,t-1} + b_c), \quad (20)$$

$$c_t = i_t \odot \tilde{c}_t + f_t \odot c_{t-1}, \quad (21)$$

$$h_t = o_t \odot \tanh(c_t) \quad (22)$$

where σ demotes the *sigmoid* function and b is a bias for each gate.

The semantic vector for each video is a 1-dimensional vector with shape 300 (as in paper [11]). It is a smaller vector than video features, so there are fewer parameters calculate with semantic vector, semantic vector might lose its impacts on the result because of a few bad trained parameters. To make better use of semantic information, we propose a semantic reconstruction network to rebuild the semantic vector which input to the decoder, if the semantic information can be used effectively, the reconstructed semantic vector will be more complete. As shown in Figure 3, an LSTM reconstructor is implemented to the decoder the semantic information, then average pooling the output of each time step to compute the reconstructed semantic vector:

$$s_r = \phi\left(\frac{1}{N} \sum_{i=1}^N h_i\right) \quad (23)$$

where h_i is the output of LSTM at i -th time step, $\phi(\cdot)$ is an average pooling process, and s_r represents the reconstructed semantic vector. The reconstruction loss is measured by Mean Square Error (MSE) loss:

$$Loss_{rec}(s, s_r) = MSE(s, s_r) \quad (24)$$

Figure 3. An illustration of the proposed semantic reconstructor module.

3.4. Training

Formally, the encoder–decoder module and reconstructor module are trained together. To generate the correct captions, the encoder–decoder module is trained by minimizing the negative log-likelihood. The loss of the encoder–decoder module can be defined as follows:

$$Loss_{e-d} = \sum_{t=1}^T (-\log P(y_t | v, y_{t-1}; \theta)) \quad (25)$$

where y_t denotes the word generated at time t , v is the encoded video features and semantic vector, and θ represents the parameters need to optimize. The total loss involving the encoder–decoder and reconstructor can be defined as:

$$Loss_{total} = \lambda MSE(s, s_r) + \sum_{t=1}^T (-\log P(y_t|v, y_{t-1}; \theta)) \quad (26)$$

where λ is a hyper-parameter to trade-off the encoder–decoder module and reconstructor module.

4. Experimental Evaluation

Our model is evaluated by using two benchmark datasets of video description, including the MSVD (Microsoft Video Description) dataset [12] and MSR-VTT (MSR-Video To Text) dataset [13]. We briefly introduce these datasets and the evaluation metrics in our experiments first, then show the experimental details and results.

4.1. Datasets

MSVD is composed of 1970 YouTube open domain videos that are popular for video captioning. Each clip in this dataset shows one single activity. Generally, it spans over 10 to 25 s. The ground-truth captions are annotated by multiple humans. On average, each clip has approximately 40 ground-truth captions in English. For benchmarking, the training, validation, and testing splits are set to 1200, 100, 670 respectively in our experiments referred to [31].

MSR-VTT is a large-scale benchmark dataset which consists of 10,000 clips that are transformed from 7180 videos. The clips were divided into 20 different categories. In addition, each video clip is annotated by Amazon Mechanical Turk (AMT) workers using 20 single sentences. Following the official evaluation protocol provided by [13], the training, validation, and testing splits include 6513, 497, and 2990 clips, respectively.

4.2. Evaluation Metrics

To compare our methods objectively with existing methods, we evaluate our models with four popular metrics such as METEOR (Metric for Evaluation of Translation with Explicit Ordering) [32], BLEU (Bilingual Evaluation Understudy) [33], ROUGE-L (Recall Oriented Understudy of Gisting Evaluation) [34] and CIDEr (Consensus-based Image Description Evaluation) [35]. The results can be computed using the Microsoft COCO server [36]. The higher score of these metrics, the higher quality of the generated captions.

4.3. Implementation Details

Before training, each dataset contains a small number of captions with too few or too many words, which is bad for training. We removed captions where the number of words is more than 30 or less than 4. In addition, build a semantic vocabulary by manual selection of the most common words in the dataset. It contains the keywords in a sentence such as “man”, “boy”, “playing”, “riding”. The vocabulary is all composed of “subjects”, “verbs” and “objects”, which can represent the main idea of the video. Our experiments are based on MSVD and MSR-VTT datasets, because of the low complexity of datasets, the size of semantic vocabulary is set to 300 (as in paper [11]).

For semantic detection, 2D and 3D video features are needed to be extracted for training a semantic detection network (SDN). The image representation we used is the output of 1536-way last *AvgPool2d* layer from InceptionResNetV2 [37] pretrained on the ImageNet dataset [19]. In addition, the video representation is extract by a 3D CNN (C3D) [38] which pretrained on the Sports-1M video dataset [39]. The video representation is a 4096-way vector obtained from the *fc7* layer of C3D. In addition, uniformly extract 40 frames from each video and take the RGB video frames as input. Video frames are resized to 112×112 for C3D and resized to 299×299 for InceptionResNetV2. The input of C3D is video clips with 16 frames (as in paper [39]) and has 8 frames overlap. We train

the semantic detector using the procedure described in model architecture. For sequence VLAD encoding, we take the output of *Conv2d_7b* layer from InceptionResNetV2 which with shape $1536 \times 8 \times 8$.

For model training, there are some well-known word embedding methods, such as GloVe [40], FastText [41] and word2vec [42]. The end-to-end video caption model proposed in this paper can be learned online, but GloVe is a count-based method that needs overall statistics. FastText uses n-gram and subword to consider the order of words, so it would build a large corpus. The last linear layer in our model which output the vectors with the size of the corpus takes up nearly half of the parameters. Using FastText will double the number of parameters, so we initialize the word embedding vectors with the publicly available word2vec vectors. In the decoder, the size of word embedding, feature embedding, and the hidden state of semantic LSTM is set to 512. Because of various lengths of sentences, every sentence added <BOS> and <EOS> to indicate the begin of a sentence and the end of a sentence. To avoid overfitting, the dropout value is set to 0.5. The batch size during training is 64. The model is optimized by Adam [43] with learning rate 2×10^{-4} . We empirically train our model with 50 epochs.

For testing, we employ a beam search algorithm to generate sentences. Beam search selects k highest probability words at each time step, then input these k words into the next time step to generate new k highest probability words. The beam size is set to 5 in our experiments.

4.4. Results and Analysis

We obtained the configuration of the modules through experiments first, then evaluate them on the dataset with the best performing model.

4.4.1. Study on Modules

The channel soft attention could have a different structure by using different channel attention module. We use the channel attention part of SEnet [28] and CBAM [7] combined with soft attention, respectively. These two kinds of channel attention are quite similar and both with 200 k parameters. We evaluated these two original attentions and the attention replaced the activate function *sigmoid* with *tanh*. The results are shown in Table 1. We use SeqVLAD as a baseline which denotes by “SVLAD”, “SE” and “CA” denote the original SEnet and CBAM (channel attention part). “SE – tanh” and “CA – tanh” denote the attention module with *tanh* activate function. As Table 1 shows, the results of SeqVLAD optimized with different channel soft attention are very close. This is because the difference between these two channel attentions is negligible, CBAM has one more maxpool operation than SEnet, and the performance of CBAM is not much better than SEnet in the classification task. The activate function *tanh* may not have much improvement, but it could work well in this model. All these models with attention have made a significant improvement compared to the baseline, which shows the effectiveness of channel soft attention.

Table 1. Comparison of different channel soft attention variation on MSVD dataset.

Model	METEOR	BLEU-4	CIDEr	ROUGE-L
SVLAD	33.8	45.9	77.5	68.9
SVLAD _{SE}	34.9	48.8	80.6	70.2
SVLAD _{SE-tanh}	34.8	49.8	81.9	71.1
SVLAD _{CA}	34.9	50.2	81.5	71.4
SVLAD _{CA-tanh}	35.1	49.9	83.7	71.5

As for semantic reconstructor, the parameter λ in Equation (26) indicates adding how much reconstruction loss in training, too large λ may ruin the caption performance. To select the best λ , we did some comparative experiments with different λ on two structures. As shown in Figure 4, The comparison on metric BLEU-4 shows that larger is not better. We empirically set λ to 0.2 in the following experiments.

Figure 4. BLEU-4 effects of parameter λ in semantic reconstructor. CSA-SR denotes the complete model based on channel soft attention and semantic reconstructor, and SR is a simplified CSA-SR which without channel soft attention.

The parameters and multiply-accumulate operations (MACs) of each module are shown in Table 2. The MACs of each module denotes the multiply-accumulate operations of one video. “CA” denotes the channel attention module, “SDN” denotes the semantic detection module and “CSA-SR” denotes the final module in our paper. As Table 2 shows, the Decoder module occupies more than half the parameters of “CSA-SR” and MACs. The total parameters and MACs are not equaled to the final model “CSA-SR” because there are some extra operations to connect these modules.

Table 2. Parameters and multiply-accumulate operations (MACs) of modules.

Module	Parameters	MACs
CA	2.05×10^5	4.93×10^5
SeqVLAD	1.08×10^6	1.97×10^7
SDN	1.39×10^7	3.48×10^7
Decoder	2.72×10^7	5.81×10^8
Reconstructor	9.77×10^5	1.47×10^5
CSA-SR	4.43×10^7	7.83×10^8

4.4.2. Results on MSVD Dataset

Before comparing it with other methods, we did some comparative experiments to prove the effectiveness of our method, and report the results of the different combinations of our method in Table 3. We first evaluate the original SeqVLAD method and its combination with channel soft attention, semantic LSTM decoder, and semantic reconstructor, where “SVLAD” denotes the SeqVLAD, “CSA” denotes the channel soft attention, “SLSTM” denotes the semantic LSTM which semantic information has been added, and “SR” denotes the semantic LSTM with reconstructor. The cluster center is set to 32 because of the good overall performance shown in paper [6]. As Table 3 shows, a single channel soft attention (SVLAD_{CSA}) can make a significant improvement in METEOR with 3.85%. The single semantic LSTM (SVLAD_{SLSTM}) and semantic reconstructor (SVLAD_{SR}) gain over the original SeqVLAD 2.34% and 2.96%, respectively. The channel soft attention improved

model has a higher score in CIDEr than others. It shows that channel soft attention makes more improvement. However, semantic reconstructor got a higher score in BLEU-4. This is because the semantic reconstructor has fully used the semantic tags. These tags indicate the most common words in the dataset. In generated captions, the more matched words, the higher the BLEU-4 score. The channel soft attention combined with semantic LSTM decoder (SVLAD_{CSA-SLSTM}) does not have much improvement compared to SVLAD_{CSA}. Channel soft attention with semantic reconstructor (SVLAD_{CSA-SR}) has shown the best overall performance.

Table 3. Comparison of different architecture variation on MSVD dataset.

Model	METEOR	BLEU-4	CIDEr	ROUGE-L
SVLAD	33.8	45.9	77.5	68.9
SVLAD _{CSA}	35.1	49.9	83.7	71.5
SVLAD _{SLSTM}	34.6	48.9	79.7	69.8
SVLAD _{SR}	34.8	51.3	80.4	70.5
SVLAD _{CSA-SLSTM}	35.2	49.8	80.1	71.8
SVLAD _{CSA-SR}	35.6	52.2	83.4	72.7

Then our method is compared with some state-of-the-art methods. The main ideas of the methods in Table 4 are introduced as follow:

1. S2VT [20] is an encoder–decoder framework with two layers of LSTM, where the first layer encodes the video representation and the second generates video descriptions.
2. SA [21] which is based on temporal attention focuses on both the local and global temporal structure of the video, and select the most relevant temporal segments automatically.
3. h-RNN [44] introduces a hierarchical structure for sentence generator and paragraph generator. Both sentence generator and paragraph generator are made up of the recurrent neural network.
4. HRNE [45] is a hierarchical recurrent neural encoder with a soft attention mechanism and explores the temporal transitions between frame chunks with different granularities.
5. SCN-LSTM [11] detects semantic tags before training caption models, and uses semantic compositional networks to embed semantic information for caption generating.
6. PickNet [46] selects the useful part of video frames for the encoder–decoder framework. It aims to reduce the computation and the noise caused by redundant frames.
7. TDDF [47] uses a task-driven dynamic fusion mechanism to reduce the ambiguity in the video description and refines the degree of the description of the video content.
8. RecNet [24] is a dual learning model that has forward flow and backward flow while training a model. The video representation will be reconstructed using the output of the decoder.
9. TDConvED [48] introduces a TDConvED that constructs encoder and decoder by fully use convolutions.
10. GRU-EVE [49] mainly applied Short Fourier Transform into CNN features which could enrich the visual features with temporal dynamics.

The performance of our model with baseline and existing models above are compared on the MSVD dataset. As Table 4 shows, SeqVLAD is a baseline in our experiments, the baseline with single semantic reconstructor (SR) and the baseline with single channel soft attention (CSA) both show a great improvement in METEOR, BLEU-4, and CIDEr. Moreover, our final model (CSA-SR) achieves a strong 83.4 value of CIDEr, which is 3.86% higher than the closest competitor RecNet [24]. The score of METEOR has only 1.71% higher than GRU-EVE [49]. This might be because GRU-EVE has applied object information and the object information might be effective in improving performance. For other metrics, the score of our method remains competitive with other methods.

Table 4. Performance comparison of our CSA-SR and other state-of-the-art methods on MSVD dataset with BLEU-4, METEOR, CIDEr and ROUGE-L metrics.

Model	METEOR	BLEU-4	CIDEr	ROUGE-L
S2VT [20]	29.2	-	-	-
SA [21]	29.6	41.9	51.6	-
h-RNN [44]	32.6	49.9	65.8	-
HRNE [45]	33.1	43.8	-	-
SCN-LSTM [11]	33.5	51.1	77.7	-
PickNet [46]	33.1	46.1	76.0	69.2
TDDF [47]	33.3	45.8	73.0	69.7
RecNet [24]	34.1	52.3	80.3	69.8
TDConvED [48]	33.8	53.3	76.4	-
GRU-EVE [49]	35.0	47.9	78.1	71.5
SeqVLAD [6]	33.8	45.9	77.5	68.9
SR	34.8	51.3	80.4	70.5
CSA	35.1	49.9	83.7	71.5
CSA-SR	35.6	52.2	83.4	72.7

Figure 5 shows some examples of generated captions in MSVD datasets. We divided them into three categories: correct captions, relevant but incorrect captions, and irrelevant captions. Generally, the correct captions are very close to the ground-truth. For instance, our model generates “a man is pouring some ingredients into a bowl” in the third video, which can describe in detail the corresponding video content. There are mainly two kinds of mistake in relevant but incorrect captions. For example, in the first relevant video, “a man is jumping on a horse” has an incorrect verb “jumping on” which should be replaced by “riding”. This mistake also appears in the second relevant video: “cutting” in “a man is cutting a bread” should be replaced by “buttering”. “a man is riding a horse” in the third relevant video contains the wrong noun “horse” which should be replaced by “motorcycle”. These mistakes might be due to the lack of training samples. Although our proposed model has achieved satisfactory results, the existence of irrelevant captions shows that our model does not always generate what we need. Our method still has a lot of room for improvement, in the sentence generation part, the quality of sentences may be improved by other learning strategies such as sequence-level reinforcement learning.

4.4.3. Results on MSR-VTT Dataset

The performance of our method is compared with recent models on MSR-VTT dataset, such as RecNet [24], TDDF [47], RUC-UVA [50], Alto [51], PickNet [46], TDConvED [48] and GRU-EVE [49]. The results are summarized in Table 5. It shows our method is competitive to the state-of-the-art method on MSR-VTT on these four metrics. These results confirm the effectiveness of our proposed channel soft attention and semantic reconstructor for video captioning.

Correct				CSA-SR: a boy is playing with a dog GT1: a boy playing with dog GT2: a man plays high five with his small dog GT3: man plays with a trained dog
	CSA-SR: a man is playing a piano GT1: a boy is playing a piano GT2: a small japanese boy playing some sort of notes on a piano GT3: a little boy is playing a grand piano
	CSA-SR: a man is pouring some ingredients into a bowl GT1: a man is pouring a liquid into a pan GT2: a man is pouring oil into a skillet GT3: a chef preparing food
Relevant				CSA-SR: a man is jumping on a horse GT1: a woman rides her horse at walking speed GT2: a girl is riding a horse GT3: a jockey rides a horse around an arena
	CSA-SR: a man is cutting a bread GT1: a man is buttering bread GT2: a man is making garlic bread GT3: a man is spreading butter on garlic bread
	CSA-SR: a man is riding a horse GT1: a man is riding a motorcycle GT2: a person rides a motorcycle on a beach GT3: a man rides a bike through the beach
Irrelevant				CSA-SR: a man is kicking a soccer ball GT1: two boys are demonstrating fighting moves GT2: two boys are doing fighting moves GT3: two people are dancing
	CSA-SR: a man is filling a pot of water GT1: a bird is bathing in a sink GT2: a bird is taking a bath GT3: a bird is taking a shower in a sink
	CSA-SR: a man is holding a piece of wood GT1: the man used a hammer to hit his hand GT2: a man hits his thumb while hammering GT3: a man injures his hand with a hammer

Figure 5. Examples of correct, relevant and irrelevant video captions. CSA-SR denotes the sentence generated by our proposed model based on channel soft attention and semantic reconstructor. GT1, GT2 and GT3 denote the ground-truth sentences.

Table 5. Comparison to related approaches on MSR-VTT dataset with BLEU-4, METEOR, CIDEr and ROUGE-L metrics.

Model	METEOR	BLEU-4	CIDEr	ROUGE-L
Alto [51]	26.9	39.8	45.7	59.8
RUC-UVA [50]	26.9	38.7	45.9	58.7
TDDF [47]	27.8	37.3	43.8	59.2
PickNet [46]	27.2	38.9	42.1	59.5
RecNet [24]	26.6	39.1	42.7	59.3
TDConvED [48]	27.5	39.5	42.8	-
GRU-EVE [49]	28.4	38.3	48.1	60.7
CSA-SR(ours)	28.2	41.3	48.6	61.9

4.4.4. User Study

The automatic sentence evaluation metrics may not show the quality of generated sentences. To evaluate the model with human judgment, a user study should be conducted. We got 10 people to evaluate the generated sentences of two testing subsets. These two testing subsets are randomly selected from the MSVD dataset and MSR-VTT dataset with 20 samples, respectively. In addition, the testing sentences are generated by our final model (CSA-SR). Users watch the same 20 videos of MSVD and MSR-VTT testing subsets, then select a proper sentence for the corresponding video from our generated sentence and a ground-truth sentence, where the ground-truth sentence is randomly selected. As shown in Table 6, “pred” denotes the number of generated sentences chosen by users, “gt” denotes the ground-truth sentences, and “rate” denotes the proportion of generated sentences. The higher “rate” indicates that the generated sentences are more appropriate from the user’s perspective. The average rate shows generated sentences have a competitive quality, especially on the MSVD dataset.

Table 6. Human judgment of generated sentences on testing subsets of MSVD and MSR-VTT.

User	$MSVD_{pred}$	$MSVD_{gt}$	$MSVD_{rate}$	$MSR - VTT_{pred}$	$MSR - VTT_{gt}$	$MSR - VTT_{rate}$
No.1	9	11	45%	7	13	35%
No.2	9	11	45%	6	14	30%
No.3	7	13	35%	8	12	40%
No.4	10	10	50%	7	13	35%
No.5	9	11	45%	9	11	45%
No.6	8	12	40%	7	13	35%
No.7	11	9	55%	9	11	45%
No.8	6	14	55%	7	13	35%
No.9	9	11	45%	5	15	25%
No.10	8	12	40%	7	13	35%
Avg	8.6	11.4	43%	7.2	12.8	36%

5. Conclusions

Existing convolutional RNNs for video feature extraction have exploited the video’s global temporal structure for fully connected features, but not include the global temporal structure of each channel, and the semantic information is not fully used. In this paper, a video captioning model based on channel soft attention and semantic reconstructor is proposed. Specifically, the soft attention is combined with channel attention, to compute every weighted channel for each time step based on the global feature maps. A semantic reconstructor is introduced to make semantic information fully used. Compared with the state-of-the-art methods on two popular benchmarks MSVD dataset and MSR-VTT dataset, the results of our experiment have shown that our methods achieve competitive or even superior performance. In the future work, we will explore more effective approaches such as reinforcement learning to optimize the sentence-level metrics for more interpretable sentences.

Author Contributions: Abstract, introduction, related works, conclusions, and the revising of the entire manuscript, Z.L.; algorithms, experiments, and results, Y.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research is partially supported by The National Natural Science Foundation of China (NSFC) under grant No. 61572306, The National Key Research and Development Program of China (NO. 2017YFB0701600) and The Science and technology committee of Shanghai Municipality under grant No. 19511121002.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank reviewers for their fruitful feedback and comments that have helped them improve the quality of this work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Wang, H.; Klaser, A.; Schmid, C.; Liu, C.L. Action Recognition by Dense Trajectories. In Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Springs, CO, USA, 20–25 June 2011; IEEE Computer Society: Los Alamitos, CA, USA, 2011; pp. 3169–3176. [\[CrossRef\]](#)
2. Wang, H.; Schmid, C. Action Recognition with Improved Trajectories. In Proceedings of the 2013 IEEE International Conference on Computer Vision (ICCV), Sydney, Australia, 1–8 December 2013; IEEE Computer Society: Los Alamitos, CA, USA, 2013; pp. 3551–3558. [\[CrossRef\]](#)
3. Jegou, H.; Schmid, C.; Douze, M.; Perez, P. Aggregating local descriptors into a compact image representation. In Proceedings of the 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, CA, USA, 13–18 June 2010; IEEE Computer Society: Los Alamitos, CA, USA, 2010; pp. 3304–3311. [\[CrossRef\]](#)
4. Sánchez, J.; Perronnin, F.; Mensink, T.; Verbeek, J. Image Classification with the Fisher Vector: Theory and Practice. *Int. J. Comput. Vision* **2013**, *105*, 222–245. [\[CrossRef\]](#)
5. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 12–17 June 2015; pp. 1–9.
6. Xu, Y.; Han, Y.; Hong, R.; Tian, Q. Sequential Video VLAD: Training the Aggregation Locally and Temporally. *IEEE Trans. Image Process.* **2018**, *27*, 4933–4944. [\[CrossRef\]](#)
7. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the 2018 European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., Eds.; Springer: Cham, Germany, 2018; pp. 3–19.
8. Luong, T.; Pham, H.; Manning, C.D. Effective Approaches to Attention-based Neural Machine Translation. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015; Association for Computational Linguistics: Lisbon, Portugal, 2015; pp. 1412–1421. [\[CrossRef\]](#)
9. You, Q.; Jin, H.; Wang, Z.; Fang, C.; Luo, J. Image Captioning with Semantic Attention. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 4651–4659.
10. Wu, Q.; Shen, C.; Liu, L.; Dick, A.; Van Den Hengel, A. What Value Do Explicit High Level Concepts Have in Vision to Language Problems? In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 203–212.
11. Gan, Z.; Gan, C.; He, X.; Pu, Y.; Tran, K.; Gao, J.; Carin, L.; Deng, L. Semantic Compositional Networks for Visual Captioning. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 1141–1150.
12. Chen, D.L.; Dolan, W.B. Collecting Highly Parallel Data for Paraphrase Evaluation. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Portland, OR, USA, 19–24 June 2021; Association for Computational Linguistics: St. Stroudsburg, PA, USA, 2011; Volume 1, pp. 190–200.
13. Xu, J.; Mei, T.; Yao, T.; Rui, Y. MSR-VTT: A Large Video Description Dataset for Bridging Video and Language. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 5288–5296.
14. Lebre, R.; Pinheiro, P.O.; Collobert, R. Phrase-Based Image Captioning. In Proceedings of the 32nd International Conference on International Conference on Machine Learning, Lille, France, 6–11 July 2015; Volume 37, pp. 2085–2094.
15. Rohrbach, M.; Qiu, W.; Titov, I.; Thater, S.; Pinkal, M.; Schiele, B. Translating Video Content to Natural Language Descriptions. In Proceedings of the 2013 IEEE International Conference on Computer Vision (ICCV), Sydney, Australia, 1–8 December 2013; IEEE Computer Society: Los Alamitos, CA, USA, 2013; pp. 433–440. [\[CrossRef\]](#)
16. Vinyals, O.; Toshev, A.; Bengio, S.; Erhan, D. Show and tell: A neural image caption generator. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 12–17 June 2015; pp. 3156–3164.
17. Donahue, J.; Hendricks, L.A.; Rohrbach, M.; Venugopalan, S.; Guadarrama, S.; Saenko, K.; Darrell, T. Long-Term Recurrent Convolutional Networks for Visual Recognition and Description. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 677–691. [\[CrossRef\]](#)
18. Venugopalan, S.; Xu, H.; Donahue, J.; Rohrbach, M.; Mooney, R.; Saenko, K. Translating Videos to Natural Language Using Deep Recurrent Neural Networks. *arXiv* **2014**, arXiv:1412.4729. doi:10.3115/v1/N15-1173.
19. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; Berg, A.C.; Fei-Fei, L. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vision* **2015**, *115*, 211–252. [\[CrossRef\]](#)
20. Venugopalan, S.; Rohrbach, M.; Donahue, J.; Mooney, R.; Darrell, T.; Saenko, K. Sequence to Sequence—Video to Text. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 11–18 December 2015; IEEE Computer Society: Los Alamitos, CA, USA, 2015; pp. 4534–4542. [\[CrossRef\]](#)

21. Yao, L.; Torabi, A.; Cho, K.; Ballas, N.; Pal, C.; Larochelle, H.; Courville, A. Describing Videos by Exploiting Temporal Structure. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 11–18 December 2015; IEEE Computer Society: Los Alamitos, CA, USA, 2015; pp. 4507–4515. [\[CrossRef\]](#)
22. Pan, Y.; Mei, T.; Yao, T.; Li, H.; Rui, Y. Jointly Modeling Embedding and Translation to Bridge Video and Language. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 4594–4602.
23. Pan, Y.; Yao, T.; Li, H.; Mei, T. Video Captioning with Transferred Semantic Attributes. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 984–992.
24. Wang, B.; Ma, L.; Zhang, W.; Liu, W. Reconstruction Network for Video Captioning. In Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 7622–7631.
25. Stollenga, M.F.; Masci, J.; Gomez, F.; Schmidhuber, J. Deep Networks with Internal Selective Attention through Feedback Connections. In Proceedings of the 27th International Conference on Neural Information Processing Systems (NIPS), Montreal, QC, Canada, 8–11 December 2014; MIT Press: Cambridge, MA, USA, 2014; pp. 3545–3553.
26. Chorowski, J.; Bahdanau, D.; Serdyuk, D.; Cho, K.; Bengio, Y. Attention-Based Models for Speech Recognition. In Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS), Montreal, QC, Canada, 7–10 December 2015; MIT Press: Cambridge, MA, USA, 2015; pp. 577–585.
27. Pan, Y.; Yao, T.; Li, Y.; Mei, T. X-Linear Attention Networks for Image Captioning. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 14–15 June 2020; pp. 10968–10977. [\[CrossRef\]](#)
28. Hu, J.; Shen, L.; Albanie, S.; Sun, G.; Wu, E. Squeeze-and-Excitation Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 2011–2023. [\[CrossRef\]](#)
29. Ray, S.; Chandra, N. Domain Based Ontology and Automated Text Categorization Based on Improved Term Frequency—Inverse Document Frequency. *Int. J. Mod. Educ. Comput. Sci.* **2012**, *4*. [\[CrossRef\]](#)
30. Kastrati, Z.; Imran, A.S.; Yayilgan, S.Y. The impact of deep learning on document classification using semantically rich representations. *Inf. Process. Manag.* **2019**, *56*, 1618–1632. [\[CrossRef\]](#)
31. Guadarrama, S.; Krishnamoorthy, N.; Malkarnenkar, G.; Venugopalan, S.; Mooney, R.; Darrell, T.; Saenko, K. YouTube2Text: Recognizing and Describing Arbitrary Activities Using Semantic Hierarchies and Zero-Shot Recognition. In Proceedings of the 2013 IEEE International Conference on Computer Vision (ICCV), Sydney, Australia, 1–3 December 2013; IEEE Computer Society: Los Alamitos, CA, USA, 2013; pp. 2712–2719. [\[CrossRef\]](#)
32. Lavie, A.; Agarwal, A. Meteor: An Automatic Metric for MT Evaluation with High Levels of Correlation with Human Judgments. In Proceedings of the Second Workshop on Statistical Machine Translation, Stroudsburg, PA, USA, 23 June 2007; pp. 228–231.
33. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.J. BLEU: A Method for Automatic Evaluation of Machine Translation. In Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Philadelphia, PA, USA, 6–12 July 2002; Association for Computational Linguistics: St. Stroudsburg, PA, USA, 2002; pp. 311–318. [\[CrossRef\]](#)
34. Lin, C.Y. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*; Association for Computational Linguistics: Barcelona, Spain, 2004; pp. 74–81.
35. Vedantam, R.; Zitnick, C.L.; Parikh, D. CIDEr: Consensus-based image description evaluation. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 4566–4575.
36. Chen, X.; Fang, H.; Lin, T.Y.; Vedantam, R.; Gupta, S.; Dollar, P.; Zitnick, C.L. Microsoft COCO Captions: Data Collection and Evaluation Server. *arXiv* **2015**, arXiv:1504.00325.
37. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A.A. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; pp. 4278–4284.
38. Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; Paluri, M. Learning Spatiotemporal Features with 3D Convolutional Networks. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; IEEE Computer Society: St. Stroudsburg, PA, USA, 2015; pp. 4489–4497. [\[CrossRef\]](#)
39. Karpathy, A.; Toderici, G.; Shetty, S.; Leung, T.; Sukthankar, R.; Fei-Fei, L. Large-Scale Video Classification with Convolutional Neural Networks. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 24–27 June 2014; IEEE Computer Society: Los Alamitos, CA, USA, 2014; pp. 1725–1732. [\[CrossRef\]](#)
40. Pennington, J.; Socher, R.; Manning, C. Glove: Global Vectors for Word Representation. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Doha, Qatar, 25–29 October 2014.
41. Joulin, A.; Grave, E.; Bojanowski, P.; Douze, M.; Jégou, H.; Mikolov, T. FastText.zip: Compressing text classification models. *arXiv* **2016**, arXiv:1612.03651.
42. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; Dean, J. Distributed Representations of Words and Phrases and Their Compositionality. In Proceedings of the 26th International Conference on Neural Information Processing Systems, Sydney, Australia, 12–15 December 2019; Curran Associates Inc.: Red Hook, NY, USA, 2019; Volume 2, pp. 3111–3119.
43. Kingma, D.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the 3rd International Conference for Learning Representations, San Diego, CA, USA, 7–9 May 2015.

44. Yu, H.; Wang, J.; Huang, Z.; Yang, Y.; Xu, W. Video Paragraph Captioning Using Hierarchical Recurrent Neural Networks. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 4584–4593.
45. Pan, P.; Xu, Z.; Yang, Y.; Wu, F.; Zhuang, Y. Hierarchical Recurrent Neural Encoder for Video Representation with Application to Captioning. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 1029–1038.
46. Chen, Y.; Wang, S.; Zhang, W.; Huang, Q. Less Is More: Picking Informative Frames for Video Captioning. In Proceedings of the 2018 European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., Eds.; Springer: Cham, Germany, 2018; pp. 367–384.
47. Zhang, X.; Gao, K.; Zhang, Y.; Zhang, D.; Li, J.; Tian, Q. Task-Driven Dynamic Fusion: Reducing Ambiguity in Video Description. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6250–6258.
48. Chen, J.; Pan, Y.; Li, Y.; Yao, T.; Chao, H.; Mei, T. Temporal deformable convolutional encoder-decoder networks for video captioning. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 8167–8174.
49. Afaq, N.; Akhtar, N.; Liu, W.; Gilani, S.Z.; Mian, A. Spatio-Temporal Dynamics and Semantic Attribute Enriched Visual Encoding for Video Captioning. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seoul, Korea, 27 October–2 November 2019; pp. 12479–12488.
50. Dong, J.; Li, X.; Lan, W.; Huo, Y.; Snoek, C.G. Early Embedding and Late Reranking for Video Captioning. In Proceedings of the 24th ACM International Conference on Multimedia, Amsterdam, The Netherlands, 15–19 October 2016; Association for Computing Machinery: New York, NY, USA, 2016; pp. 1082–1086. [[CrossRef](#)]
51. Shetty, R.; Laaksonen, J. Frame- and Segment-Level Features and Candidate Pool Evaluation for Video Caption Generation. In Proceedings of the 24th ACM International Conference on Multimedia, Amsterdam, The Netherlands, 15–19 October 2016; Association for Computing Machinery: New York, NY, USA, 2016; pp. 1073–1076. [[CrossRef](#)]