Applied Soft Computing 117 (2022) 108332

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

An attention based dual learning approach for video captioning

Wanting Ji [a], Ruili Wang [b,*], Yan Tian [b], Xun Wang [b]

[a] School of Information, Liaoning University, Shenyang, China
[b] School of Computer Science and Information Engineering, Zhejiang Gongshang University, Hangzhou, China

ARTICLE INFO

Article history:
Received 22 December 2020
Received in revised form 24 November 2021
Accepted 12 December 2021
Available online 21 December 2021

Keywords:
Attention mechanism
Deep neural network
Dual learning
Encoder–decoder
Video captioning

ABSTRACT

Video captioning aims to generate sentences/captions to describe video contents. It is one of the key tasks in the field of multimedia processing. However, most of the current video captioning approaches utilize only the visual information of a video to generate captions. Recently, a new encoder–decoder–reconstructor architecture was developed for video captioning, which can capture the information in both raw videos and the generated captions through dual learning. Based on this architecture, this paper proposes a novel attention based dual learning approach (ADL) for video captioning. Specifically, ADL is composed of a caption generation module and a video reconstruction module. The caption generation module builds a translatable mapping between raw video frames and the generated video captions, i.e., using the visual features extracted from videos by an Inception-V4 network to produce video captions. Then the video reconstruction module reproduces raw video frames using the generated video captions, *i.e.*, using the hidden states of the decoder in the caption generation module to reproduce/synthesize raw visual features. A multi-head attention mechanism is adopted to help the two modules focus on the most effective information in videos and captions, and a dual learning mechanism is adopted to fine-tune the performance of the two modules to generate final video captions. Therefore, ADL can minimize the semantic gap between raw videos and the generated captions by minimizing the differences between the reproduced and the raw videos, thereby improving the quality of the generated video captions. Experimental results demonstrate that ADL is superior to the state-of-the-art video captioning approaches on benchmark datasets.

1. Introduction

Video captioning aims to generate sentences/captions to describe video contents [1–3]. It has received increasing attention in the fields of video understanding [4], natural language processing [5–7], and computer vision [7–9]. In the real world, video captioning based applications, such as video captioning based transcriptions and blind navigation, are widely used in surveillance systems, healthcare, and smart cities, and demonstrate their enormous scientific and commercial potential in these applications [4,5].

Compared to other captioning tasks (*e.g.*, image captioning [10–12]), video captioning is more challenging. This is because a video contains more complicated information (*e.g.*, actions, objects, and scenes) than a still image [13–15]. The existing video captioning approaches are mainly based on two types of models: (i) template based language models and (ii) sequence learning based models.

Early efforts on video captioning mainly focused on template based language models. The template based language models [16–18] predefine a series of language templates, and map video features to words using object detection methods. Then the detected words are placed on a predefined template to form a video caption that follows specific grammatical rules to describe the video content. Thus, each part of the generated sentence can be aligned with the video content based on the predefined templates. However, since such captions are composed of the detected words, template based language models only describe the detected video contents, *i.e.*, part of the video contents. Furthermore, since the syntactical structure of a caption is predefined by the templates, the generated caption is rather 'robotic', *i.e.*, not quite like a natural language sentence [13].

Recently, various deep learning techniques have obtained significant success in the fields of image processing and speech processing [19–22]. These techniques have also been introduced to the video captioning task. The video captioning models that are achieved through these deep learning techniques are named sequence learning based models, also known as the encoder–decoder models [13].

A sequence learning based model usually includes two stages: an encoding stage and a decoding stage. In the encoding stage,

* Corresponding author.
E-mail address: prof.ruili.wang@gmail.com (R. Wang).

https://doi.org/10.1016/j.asoc.2021.108332
1568-4946/

convolutional neural networks (CNNs) are often used as an encoder to convert a video into a compact vector to extract video features from videos. After that, the extracted video features are fed into a recurrent neural network (RNN) based decoder for video caption generation. Compared with the video captions generated by the template based language models, the video captions generated by the sequence based learning models can have more flexible syntactical structures.

Furthermore, since the encoder–decoder models allow the training process to work in an end-to-end manner, it has become the mainstream of current video captioning models. However, the encoder–decoder models have a limitation in generating video captions. Specifically, in the training process of an encoder–decoder model, the previous ground-truth word is often used as the input of the decoder at each time step. However, in the test process, this input is replaced by the previously generated word that is generated by the decoder [23]. This exposure bias may lead to error accumulation during the test process. In other words, during the test process, once a "bad" word is generated by the model, this error will be propagated and accumulated as the length of the sequence increases.

To overcome the aforementioned problem, a reconstruction network (RecNet) was proposed in [1] with a new encoder–decoder–reconstructor architecture. The proposed network generated video captions through dual learning on two flows (a video-to-sentence flow and a sentence-to-video flow). Specifically, the video-to-sentence flow encoded video semantic features to produce video captions, and the sentence-to-video flow reconstructed the video features using the output of the video-to-sentence flow. A soft attention mechanism was used in both flows to capture key information from video features and generated captions. However, this simple temporal attention mechanism cannot capture the internal relationships of key information [24].

To overcome the above problem, this paper proposes a novel attention based dual learning approach (ADL) for video captioning. Based on the similar architecture in [1], a multi-head attention mechanism replaces the soft attention mechanism to capture key information in raw videos and generated video captions. Specifically, two modules (*i.e.*, a caption generation module and a video reconstruction module) are contained in the proposed approach. The caption generation module is developed to generate video captions using the visual features extracted from videos through an Inception-V4 network [25]. The video reconstruction module reproduces/synthesizes the raw video feature sequences (*i.e.*, the raw video frames) using the hidden states of the decoder in the caption generation module. A multi-head attention mechanism is adapted to the two modules to help them focus on the most effective information in raw videos and video captions, and a dual learning mechanism is used to fine-tune the two modules. Therefore, the proposed approach can minimize the semantic gap between raw videos and generated captions by minimizing the differences between the reproduced and the raw videos, thereby enhancing the quality of the generated video captions. Experimental results demonstrate that, on benchmark datasets, our approach is superior to the state-of-the-art video captioning approaches.

The rest of this paper is organized as follows. The related work on video captioning is discussed in Section 2. Section 3 details our proposed ADL approach for video captioning. The experimental setting details and results are discussed in Section 4. The conclusion and our future work are presented in Section 5.

2. Related work

Generating descriptive captions for videos is a key task, which has received extensive attention in the fields of video understanding, natural language processing, and computer vision in recent years. The existing video captioning approaches can be classified into two categories: template based language models and sequence learning based models. In this section, Sections 2.1 and 2.2 briefly introduce the two types of video captioning models, and Section 2.3 reviews the applications of dual learning.

2.1. Template based language model

Early work [16–18] for video captioning mainly rely on template based language models, which predefine a set of language templates for caption generation. Specifically, a sentence is separated into several phases (*e.g.*, subject, verb, and object) based on the predefined templates with specific grammar rules [13]. By using object detection methods, each word can be aligned with a part of video information, and then all detected words can be placed in different phases of a template to generate a video caption. To detect objects in a video, Kojima et al. [16] developed a human activity description method based on concept hierarchies of actions. However, the generation of their method was limited to narrow domains and small vocabularies of actions. To describe arbitrary activities in videos, Guadarrama et al. [9] developed a method named zero-shot recognition to recognize activities in a video and described the recognized activities using semantic hierarchies.

On a different tack, Rohrbach et al. [17] developed a video captioning approach that introduced a conditional random field to simulate the connections/relationships between objects and activities in a video. Thus, in their approach, both visual features and semantic features were used for generating video captions. Further, Xu et al. [18] developed a video captioning framework that contained a joint embedding module, a deep video module, and a semantic language module, to generate video captions from videos.

However, since the template based language models were incapable of textualizing everything in videos, *i.e.*, mapping all video information to words, the sentences generated by these models only described part of video contents. In addition, since the templates predefined the syntactic structures of video captions, the generated sentences were based on simple and uniform syntactic structures, which were somewhat robotic in some cases. Thus, the sequence learning based model has been developed for video captioning.

2.2. Sequence learning based model

Recent achievements in deep learning techniques have significantly enhanced the performance of video captioning approaches. Compared to template based language models, sequence learning based models can generate video captions with more flexible syntactical structures. This is because the model can learn the probability distributions of video contents and natural language sentences in a common space.

A typical architecture of the sequence learning based video captioning model is to combine CNNs and RNNs, where CNNs are utilized to extract compact representational vectors from the input videos, and RNNs are utilized to construct a language model that operates on the extracted vectors for video caption generation. Venugopalan et al. [26] computed video representation vectors by averaging the features of each video frame extracted by CNNs, and then these vectors were fed into a long short term memory (LSTM) network [26] for caption generation.

To capture the temporal dynamics of video sequences, Venugopalan et al. [27] developed the well-known Sequence to Sequence Video to Text (S2VT) approach, which utilized the optical flow to extract temporal information, and used LSTMs on both the encoder and the decoder. Zhang and Tian [28] proposed a

two-stream neural network to exploit both spatial and temporal information for video captioning.

Later, Pan et al. [29] proposed a video captioning approach named LSTM-E with an encoder–decoder architecture. In this approach, video representations were produced by mean pooling over the frame features extracted by a 2D/3D CNN, and an LSTM was used as the decoder. Ballas et al. [30] proposed a video captioning approach named GRU-RCN, which learned the spatio-temporal features in videos from intermediate visual representations using a gated-recurrent-unit recurrent network (GRU). This approach calculated the convolutional maps of video frames and fed them into a convolutional GRU-RNN (i.e., GRU-RCN) at different time steps.

HRNE [31] is a video captioning approach, which could exploit long-term temporal information in videos. Specifically, this approach generated video representations with an emphasis on temporal modeling. This reduced the length of input information flow and exploited temporal transitions with multiple granularities. h-RNN [32] is a hierarchical-RNN framework to transfer videos to natural sentences. Specifically, it consisted of two generators. A sentence generator was based on a spatiotemporal attention mechanism to produce a short sentence that described a specific short video interval. Then a paragraph generator was used to capture the inter-sentence dependency utilizing the output of the sentence generator. LSTM-LS [33] is a video captioning framework with listwise supervision. It generated a sentence ranking list for each video. In other words, by using a nearest-neighbor search, sentences could be associated with videos. Then a set of rank triplets was used to collect ranking information and measure the quality of the ranking list. Thus, the quality of the generated video captions could be improved by maximizing the ranking quality.

Furthermore, attention mechanisms were introduced to the video captioning models, which have been proven as an effective way to enhance the performance of video captioning models with the encoder–decoder structure. Yao et al. [34] assigned weights to video frame features by an attention mechanism and then fused them according to the attentive weights. Yan et al. [3] proposed a spatial–temporal attention mechanism for video captioning, which captured information from the spatial–temporal structures in a video and selected the significant regions from the most relevant video segments to generate captions. However, this approach only considered visual information for caption generation. The aLSTMs [35] is an attention-based LSTM framework for video captioning. Specifically, it developed a new attention mechanism, based on which LSTM could capture the significant structural features of the input videos as well as the correlation between multimodal features (such as textual and visual features), to ensure the semantic consistency between the visual contents of videos and the sentence descriptions.

Recently, multimodal learning was also introduced to video captioning models to improve the quality of the generated captions, since a video contains multiple modalities, such as visual modality, audio modality, and textual modality. Wang et al. [2] proposed a Multimodal Memory Model for video captioning based on textual and visual modalities to solve the visual-textual alignment problem. They developed a visual and textual shared memory that modeled long-term visual-textual dependency and guided visual attention for video caption generation by interacting with videos and captions.

2.3. Dual learning approaches

Dual learning has been effectively applied for many machine learning applications, such as machine translation [36–39], image-to-image transformation [40–42], sentiment analysis [43], image segmentation [44], etc. The main idea of dual learning is very intuitive: leveraging the duality between two related tasks as a feedback signal to boost the performances of both tasks [45,46].

Usually, a dual learning framework contains two agents (a primal model and a dual model) to utilize such duality. The primal model maps an x from one domain to another, while the dual model maps it back. The mapping functions between these two domains are trained simultaneously so that one function is close to the inverse of the other. For example, when using dual learning in machine translation, if we translate a sentence from Chinese to English and then translate the obtained English sentence back to Chinese, the same sentence or a very similar one can be obtained.

He et al. [36] first proposed dual learning and applied it to machine translation. They updated two dual translators in a reinforcement learning manner and utilized the reconstructed distortion as the feedback signal. After that, Wang et al. [37] and Xia et al. [47] considered the joint distribution constraint in dual learning. They have proved that the joint distribution of samples over two domains is invariant when computing from either domain. Xia et al. [48] proposed a model-level dual learning approach, which shared components between the primary model and the dual model.

In addition, Zhao et al. [23] proposed a cross-domain image captioning approach using dual learning to overcome the problem of lack of image-text pairs in the training set. Wang et al. [45] proposed a multi-agent dual learning framework, which consisted of multiple primal and dual models, for machine translation and image translation.

In this paper, our proposed approach utilizes attention based dual learning for video captioning. Unlike the existing encoder–decoder model which only contains a video-to-caption forward flow, we also build a caption-to-video backward flow. In other words, by fully considering bidirectional training between videos and captions, our proposed approach is able to further enhance the accuracy of video captioning.

3. Framework

This paper proposes a novel ADL approach for video captioning. As illustrated in Fig. 1, ADL includes two modules: a caption generation module and a video reconstruction module. The caption generation module constructs the forward flow from videos to captions by learning a translatable mapping between video frames and captions. The backward flow from captions to videos is formed by the video reconstruction module, which is able to synthesize raw video feature sequences based on the hidden state sequences of the decoder. A multi-head attention mechanism is used in the two modules, helping them focus on the most effective information for video captioning. The two modules are fine-tuned via dual learning, and the whole approach is trained in an end-to-end fashion.

In this section, a brief introduction of RNN and LSTM is provided in Section 3.1, and the two modules are presented in Sections 3.2 and 3.3, respectively. The loss function of the proposed approach is presented in Section 3.4 for training.

3.1. Long short term memory network

RNN is a class of deep neural networks extended from the feedforward neural network by adding feedback connections. RNNs have shown extraordinary ability in dealing with sequence learning. It is because it contains a specially designed recurrent operation that models sequence information by maintaining the historical sequential information inside hidden units.

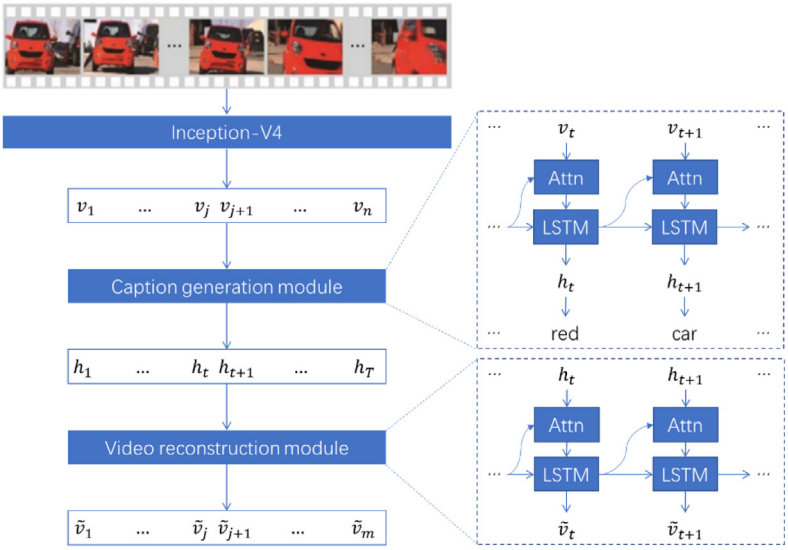

Fig. 1. An illustration of the ADL approach for video captioning. Specifically, an Inception-V4 network is used to extract visual information from a video. The caption generation module generates video captions based on the extracted features. The output of the caption generation module will be fed into the video reconstruction module for video synthesizing/reproducing. A multi-head attention mechanism is used in the two modules. Thus, the proposed approach can improve the quality of the generated video captions by minimizing the differences between the synthesized/reproduced and raw videos.

An RNN is usually used to process the video captioning task. In a video captioning method, the input of RNN can be a video composed of n video frames, *i.e.*, $\mathcal{V} = \{v_1, v_2, \ldots, v_n\}$, and the output of RNN is a descriptive sentence composed of T words $S = \{s_1, s_2, \ldots, s_T\}$. Mathematically, the output sequence $\{s_1, s_2, \ldots, s_T\}$ can be calculated by the input vectors $\{v_1, v_2, \ldots, v_n\}$ according to the following equations:

$$h_t = \tanh\left(W_h v_t + U_h v_{t-1} + b_h\right), \tag{1}$$

$$s_t = softmax\left(U_s h_t + b_s\right), \tag{2}$$

where h_t is the hidden state at time step t ($t = 1, \ldots, T$); matrices W_* and U_* denote the weights to be learned; b_* denotes a bias term. Thus, the input v_t and the previous hidden layer's state h_{t-1} can be utilized to obtain the current hidden layer's state h_t and the current hypothesis s_t The historical information of a sequence is transmitted throughout the whole sequence and affects the output at each time step.

However, standard RNNs have difficulties in dealing with long-term temporal information in some cases due to the gradient exploding or vanishing problem. Thus, a variant of the standard RNN, LSTM network was proposed.

Compared with the standard RNN, LSTM is equipped with an additional memory cell to selectively remember the previous inputs. The scale of historical information that a network can forget or remember is controlled by the memory cell, thereby overcoming the gradient exploding or vanishing problem. Thus, LSTM is more efficient than the standard RNN when dealing with tasks that require very deep structures.

In LSTM, the memory cell c_t and the hidden state h_t can be calculated by the following equations:

$$f_t = \sigma\left(W_f v_t + U_f h_{t-1} + b_f\right), \tag{3}$$

$$i_t = \sigma\left(W_i v_t + U_i h_{t-1} + b_i\right), \tag{4}$$

$$o_t = \sigma\left(W_o v_t + U_o h_{t-1} + b_o\right), \tag{5}$$

$$g_t = \tanh\left(W_g v_t + U_g h_{t-1} + b_g\right), \tag{6}$$

$$c_t = f_t c_{t-1} + i_t g_t, \tag{7}$$

$$h_t = o_t * \tanh\left(c_t\right), \tag{8}$$

where $\sigma(\cdot)$ is a sigmoid activation function; i_t, f_t and o_t are the three gates in the memory cell, and $*$ is the multiplication operator.

In LSTM, the input gate i_t and the forget gate f_t control whether to remember the current input v_t or forget the previous memory c_{t-1}, and the output gate o_t determines which history information in the memory cell c_t can be transported to the hidden state h_t. Thus, the collaboration of these three gates allows LSTM to perform or model long-term sequence information.

3.2. Caption generation module

The purpose of video captioning is to produce a descriptive sentence $S = \{s_1, s_2, \ldots, s_T\}$, which is able to depict the content of a video $\mathcal{V} = \{v_1, v_2, \ldots, v_n\}$. Conventional encoder–decoder structures usually establish models for the caption generation probability word-by-word:

$$P\left(S|V\right) = \prod_{t=1}^{T} P\left(s_t|s_{<t}, \mathcal{V}; \theta\right), \tag{9}$$

where T denotes the length of the sentence S; $s_{<t}$ denotes the partial caption that has been generated, *i.e.*, $\{s_1, s_2, \ldots, s_{t-1}\}$; θ denotes the parameters in an encoder–decoder model.

In the encoding stage: To generate reliable video captions, some visual features, which contain the high-level semantic information of a video, need to be captured (*i.e.*, the process of feature extraction). Previous approaches usually leverage CNNs (*e.g.*, AlexNet [26], VGG19 [49], and GoogLeNet [39]) for feature extraction since these networks can convert each video frame into a fixed-length video representation that contains high-level semantic information.

Considering that we need a deeper network to extract video representation, in this paper, the Inception-V4 [25] is introduced as an encoder to extract features from raw videos. Thus, a given video $\mathcal{V}$ can be encoded into a sequence $\{v_1, v_2, \ldots, v_n\}$ as video representation, where n denotes the total frame number of a video.

In the decoding stage: The decoder generates captions word-by-word according to the video representation. Usually, LSTM, which is capable of modeling long-term temporal dependencies,

is utilized as a decoder to convert the video representation into video captions. Moreover, to further exploit the most salient regions in videos, attention mechanisms are often introduced into the decoder, which is used to select the key video frames for captioning. In this paper, LSTM is utilized as a decoder to convert video representations into video captions, and a multi-head dot product attention (MHDPA) [50] is employed to help the decoder to exploit the most salient regions in videos.

During the process of video captioning, the word prediction at the time step t is performed as:

$$P\left(s_t|s_{<t}, \mathcal{V}, \theta\right) \propto exp\left(\varphi\left(s_{t-1}, h_t, e_t; \theta\right)\right), \tag{10}$$

where $\varphi(\cdot)$ denotes the LSTM inference; h_t denotes the LSTM hidden state calculated at the time step t; e_t denotes the context vector calculated by MHDPA at the time step t.

Moreover, since we utilize MHDPA to assign attention weight α_j^t to the video representation of each frame $\{v_1, v_2, \ldots, v_n\}$ at the time step t, the tth context vector can be calculated as follows:

$$e_t = \sum_{j=1}^{n} \alpha_j^t v_j, \tag{11}$$

where n denotes the frame number of a video.

As demonstrated in [34], the attention mechanisms encourage the decoder to choose a subset of key video frames to produce the most appropriate word at each time step. In other words, all currently generated words are summarized (or memorized) in the $t-1$th hidden state h_{t-1}. Then the correlations between the jth feature in the video sequence and all currently produced words can be reflected by the attention weight α_j^t at the time step t. Thus, e_t replaces v_t as the input of LSTM in the caption generation module of our proposed approach.

MHDPA is a self-attention mechanism proposed in [50]. Specifically, it utilizes matrices Q, K, and V to respectively store all queries, keys, and values. All these queries, keys, and values can be built by using a linear projection:

$$Q = W_q M, \tag{12}$$

$$K = W_k M, \tag{13}$$

$$V = W_v M, \tag{14}$$

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V. \tag{15}$$

$$head_i = Attention\left(QW_i^Q, KW_i^K, VW_i^V\right), \tag{16}$$

$$MultiHead(Q, K, V) = concat(head_1, \ldots, head_m)W^o, \tag{17}$$

where W_*^* denotes weight matrices; d_k denotes the dimensionality of key vectors; M denotes the input of the attention mechanism, which has the same dimensions as the hidden layer of the LSTM network; and $softmax(\cdot)$ denotes a softmax function.

In this paper, the proposed caption generation module is jointly trained by minimizing the negative log-likelihood to generate accurate natural language sentences for the given videos. Mathematically,

$$\min_{\theta} \sum_{t=1}^{T} \left\{-\log P\left(S^t|\mathcal{V}^t; \theta\right)\right\}. \tag{18}$$

3.3. Video reconstruction module

As shown in Fig. 1, the proposed video reconstruction module is used to reproduce videos. In other words, it is to generate vectors that can represent the content of video frames according to the hidden state sequence of the decoder. However, it is difficult to directly reproduce video frames using the hidden states in the caption generation module due to the high dimension and diversity of raw video frames. Thus, in this paper, the proposed video reconstruction module takes the hidden states sequence of the decoder $H = \{h_1, h_2, \ldots, h_T\}$ as input to reproduce the video representations created by the encoder.

The benefits of building such a module is two-fold: (i) with such a video reconstruction process, more useful information can be extracted from raw video sequences by the decoder; (ii) the proposed video reconstruction module is able to be trained in an end-to-end fashion. Thus, the relationships between the raw videos and the generated video captions are able to be further enhanced, so as to improve the accuracy of video captioning.

The proposed video reconstruction module is composed of LSTM and MHDPA. During the process of video reconstruction, the reproduced video frame at the time step t is performed as:

$$P\left(\tilde{v}_t|\tilde{v}_{<t}, S, \theta\right) \propto exp\left(\varphi\left(\tilde{v}_{t-1}, h_t, \mu_t; \theta\right)\right), \tag{19}$$

where $\varphi(\cdot)$ denotes the LSTM inference; h_t denotes the LSTM hidden state calculated at the time step t; μ_t denotes the context vector calculated by MHDPA at the time step t. Thus, for each frame, the video representation can be reproduced by the key hidden states of the decoder which is chosen by MHDPA:

$$\mu_t = \sum_{j=1}^{m} \beta_j^t h_j, \tag{20}$$

where β_j^t denotes the attention weight calculated by MHDPA for the jth hidden state at time step t. Thus, the correlations between the jth hidden state in the generated captions and all currently reconstructed video representations $\{\tilde{v}_1, \tilde{v}_2, \ldots, \tilde{v}_{t-1}\}$ can be measured by β_j^t. This helps the proposed video reconstruction module to selectively process the hidden states according to the attention weight β_j^t and dynamically generate contextual information μ_t at each time step. Moreover, both the generated context μ_t and the hidden state h_t are used as the input of LSTM in the video reconstruction module. Therefore, the proposed video reconstruction module is able to further employ the word composition and the temporal dynamics of the whole video caption to close the relationships between the raw videos and the generated video captions.

3.4. Loss function

In this paper, since the video representation is produced frame by frame, we define the reconstruction loss function as:

$$L_{rec} = \frac{1}{m}\sum_{j=1}^{m} \psi\left(\tilde{v}_j, v_j\right), \tag{21}$$

where $\tilde{v}_j$ denotes the hidden states of the proposed video reconstruction module; $\tilde{v}_j$ denotes the video representation; $\psi(\cdot)$ is the Euclidean distance measure function.

As shown in Eq. (21), the proposed ADL approach is trained by minimizing the whole loss function. The whole loss function contains two phases: one is a video-to-sentence phase that is calculated by the forward likelihood; the other is a sentence-to-video phase that is calculated by the backward loss function. Thus, the loss function of the proposed approach can be defined as:

$$L\left(\theta, \theta_{rec}\right) = \sum_{j=1}^{n}\left(-\log P\left(S^j|\mathcal{V}^j; \theta\right) + \lambda L_{rec}\left(\mathcal{V}^j, Z^j; \theta_{rec}\right)\right). \tag{22}$$

where the generation loss $-\log P\left(S^i|\mathcal{V}^i; \theta\right)$ can be calculated by Eq. (18); the reconstruction loss $L_{rec}\left(\mathcal{V}^i, Z^i; \theta_{rec}\right)$ can be calculated by Eq. (20); the hyper-parameter λ is introduced to find

Algorithm 1: ADL training algorithm

Input: Training pairs <*video* $\mathcal{V}$, *ground-truth caption*>
Output: Generated captions S
Randomly initialize parameters;
Extract $\{v_1, v_2, \ldots, v_n\}$ from videos using the Inception-V4 network;
for *each epoch* **do**
  Caption generation module:
  while θ has not converged **do**
    Generate $\{s_1, s_2, \ldots, s_T\}$ based on forward likelihood using Eq. (10);
  end while
  Video reconstruction module:
  while θ has not converged **do**
    Generate reconstructed videos $\{\widetilde{v}_1, \widetilde{v}_2, \ldots, \widetilde{v}_{t-1}\}$ using Eq. (22);
  end while
  Calculate loss function $L(\theta, \theta_{rec})$;
end for

a compromise between the proposed caption generation module and the proposed video reconstruction module. The larger the difference between the generated results and the ground truth, the greater the gradient of the loss function, and the faster the convergence rate.

As shown in Algorithm 1, the training of the proposed ADL approach can be separated into two phases:

(i) In the first phase, we train the proposed caption generation module based on the forward likelihood, which terminates the training process according to the early stopping strategy.
(ii) In the second phase, we utilize the whole loss function to jointly train the video reconstruction module and fine-tune the caption generation module. Both the hidden state sequence and the video frame feature sequence are used to calculate the video reconstruction loss function.

4. Experiments

We evaluate the proposed ADL approach on two benchmark datasets: Microsoft Research video to text (MSR-VTT) [49] dataset and Microsoft Research Video Description Corpus (MSVD) [51]. To demonstrate the effectiveness of ADL, we utilize the popular evaluation metrics including METEOR [52], BLEU-4 [53], ROUGE-L [54], and CIDEr [55] with the codes released on the Microsoft COCO evaluation server [56].

4.1. Datasets and experimental setting

The details of the two benchmark datasets are shown below:
MSVD: MSVD consists of 1970 YouTube video clips, each of which describes one single activity, of length between 10 and 25 s. Each video clip was annotated with approximately 40 English captions. In this paper, similar to [1], 1200 video clips are used as the training set, 100 video clips are used as the validation set, and 670 video clips are used as the test set.
MSR-VTT: MSR-VTT is one of the largest datasets for video captioning so far. In this paper, the initial version of MSR-VTT (*i.e.*, MSR-VTT-10K) is utilized for experiments. MSR-VTT-10K consists of 10K video clips from 20 categories. Approximately 20 sentences are used to annotate a video clip. In summary, MSR-VTT-10K consists of a total of 29,316 unique words and 200K clip-sentence pairs. Similar to [1], 6513 video clips are used as the training set, 497 video clips are used as the validation set, and 2990 video clips are used as the test set.

For the sentences in datasets, we removed punctuations, separated the sentences with blank spaces, and then transformed all words into lowercase. We truncated the sentences longer than 30 words. For each word, the word embedding size is set as 468.

For the proposed caption generation module, all frames in a video clip are fed into Inception-V4 pre-trained on the ILSVRC2012-CLS classification dataset [57]. In this way, all video frame features can be reshaped to the standard size 299×299, so that the semantic feature of each video frame can be extracted from the last pooling layer with 1536 dimensions.

Inspired by [34], for each video clip, 28 equally spaced features are selected. When the number of features is less than 28, zero vectors are used for filling. Moreover, we set the input dimension of the decoder to 468, which is equal to the dimension of the word embedding. In addition, there are 512 units contained in the hidden layer.

In the video reconstruction module, the hidden state of the decoder is taken as the input, the dimension of which is set to 512. To simplify the calculation of the reconstruction loss function, we set the size of the hidden layer to the same size as the video presentation, *i.e.*, 1536 dimensions.

Wang et al. [1] have verified that for a dual learning based approach, the hyper-parameter λ can balance the contributions of the two modules (*i.e.*, the caption generation module and the video reconstruction module). Thus, the selection of the hyper-parameter λ is crucial. Although they have shown that adding the reconstruction loss is able to enhance the performance of video captioning, too large λ may cause an obvious decrease in the performance of video caption generation. Therefore, in this paper, we test the proposed approach with λ values of 0.1 and 0.3, and set λ to 0.1 based on the experimental results.

We utilized AdaDelta [58] to optimize the training process. Furthermore, the training process was stopped when the CIDEr value on the validation set stopped growing for the next 20 consecutive epochs. Then in the test process, we used a beam search with size 5 to generate the final video captions. The activation functions used for LSTM include the sigmod function and tanh function, the batch size is set to 20, the learning rate is set to 5e−5 on the MSVD dataset and 2e−4 on the MSR-VTT dataset, and the rnn_dropout is set to 0.5.
Hardware and Software Environment: The experiments in this paper are executed on a deep learning workstation with an Intel Core i9 CPU, four GTX 1080 Ti GPUs, and 128 GB RAM. We implement the proposed approach in Python. The deep learning framework we use is TensorFlow2.0.
METEOR: METEOR is a popular evaluation metric that has been widely used to measure the similarity between sentences [52]. METEOR has been shown to produce the closest results to human judgments when only a few sentence references are given.

Table 1
Experimental results of different video captioning approaches in terms of METEOR, BLEU-4, ROUGE-L, and CIDEr scores on MSVD (%).

Approaches	METEOR	BLEU-4	ROUGE-L	CIDEr
MP-LSTM (AlexNet) [1]	29.1	33.3	–	–
$\text{RecNet}_{\text{local}}$ (S2VT) [1]	32.7	43.7	68.6	69.8
$\text{RecNet}_{\text{local}}$ (SA-LSTM) [1]	34.1	52.3	69.8	80.3
LSTM-E [29]	31.0	45.3	–	–
GRU-RCN [30]	31.6	43.3	–	68.0
HRNE [31]	33.1	43.8	–	–
h-RNN [32]	32.6	49.9	–	65.8
aLSTMs [35]	33.3	50.8	–	74.8
LSTM-LS (VGG19) [33]	31.2	46.5	–	–
ADL (AlexNet)	32.7	46.8	69.6	71.8
ADL (VGG19)	33.3	48.8	69.2	75.6
ADL (Inception-V4)	**35.7**	**54.1**	**70.4**	**81.6**

Table 2
Experimental results of different video captioning approaches in terms of METEOR, BLEU-4, ROUGE-L, and CIDEr scores on MSR-VTT (%).

Approaches	METEOR	BLEU-4	ROUGE-L	CIDEr
MP-LSTM (AlexNet) [1]	23.4	32.3	–	–
MP-LSTM (GoogLeNet) [1]	24.6	34.6	–	–
MP-LSMT (VGG19) [1]	24.7	34.8	–	–
SA-LSTM (AlexNet) [1]	23.8	34.8	–	–
SA-LSTM (GoogLeNet) [1]	25.2	35.2	–	–
SA-LSTM (VGG19) [1]	25.4	35.6	–	–
SA-LSTM (Inception-V4) [1]	25.5	36.3	58.3	39.9
$\text{RecNet}_{\text{global}}$ (SA-LSTM) [1]	26.2	38.3	59.1	41.7
$\text{RecNet}_{\text{local}}$ (SA-LSTM) [1]	26.6	39.1	59.3	42.7
ADL (AlexNet)	25.9	37.1	59.0	41.9
ADL (VGG19)	26.3	38.1	59.0	43.1
ADL (Inception-V4)	**26.6**	**40.2**	**60.2**	**44.0**

BLEU: BLEU was originally proposed as an evaluation metric for machine translation [53]. It focuses on the accuracy of machine translation. In dense video captioning, it is concerned with how many n-grams in the generated captions are the same as the n-gram in the ground truth. That is, it compares the overlapping degree between the generated sentence and the ground truth.
ROUGE-L: In contrast to BLEU, ROUGE-L focuses on the recall rate [54]. In dense video captioning, it calculates the longest common subsequence length of the generated captions and the ground truth. The longer the length of the common subsequence, the higher the ROUGE-L score. A high ROUGE-L score means that the generated caption is similar to the ground truth.
CIDEr-D: CIDEr-D considers each sentence in the captions as a document, and then calculates the cosine angle of its TF–IDF vector [55]. In this way, CIDEr-D can obtain the similarity between the generated captions and the ground truth.

4.2. Experimental results

We test the performance of ADL on two benchmark datasets for video captioning. Tables 1 and 2 show the quantitative experimental results on these two datasets. On the two datasets, we compare our ADL approach with several classical encoder–decoder approaches and the state-of-the-art approaches, including MP-LSTM [1], SA-LSTM [1], RecNet [1], LSTM-E [29], GRU-RCN [30], HRNE [31], h-RNN [32], aLSTMs [35], and LSTM-LS [33], for video captioning.

MP-LSTM [1] is a video captioning approach with an encoder–decoder architecture. It utilized a CNN (i.e., AlexNet, GoogLeNet, or VGG19) as the encoder, an LSTM network as the decoder, and a mean pooling technique to aggregate the features captured from video frames. SA-LSTM [1] is also a video captioning approach, which utilized a CNN (i.e., AlexNet, GoogLeNet, VGG19, or Inception-V4) as the encoder and an LSTM network as the decoder. Similarly, we also verify the performances of AlexNet, VGG19, and Inception-V4 for feature extraction.

Differing from MP-LSTM [1], SA-LSTM [1] contained an attention mechanism to aggregate the features captured from video frames. RecNet [1] is a video captioning approach with an encoder–decoder–reconstructor architecture. It consisted of an encoder–decoder structure to generate sentences from videos and a decoder–reconstructor structure to reproduce videos using the generated sentences. Two types of attention mechanisms (i.e., local attention mechanism and global attention mechanism) were used to capture information from the generated sentences.

On the MSVD dataset, we compare the proposed ADL approach with several classical encoder–decoder approaches and state-of-the-art approaches, such as MP-LSTM [1] and RecNet [1], for video captioning. As shown in Table 1, on the MSVD dataset, the proposed approach achieves the best performance of the reference approaches. Compared with the classical encoder–decoder approaches, e.g., MP-LSTM [1], ADL is able to obtain better evaluation results since it contains an additional video reconstruction module to improve the captioning accuracy. Furthermore, although the training time/convergence rate of the proposed ADL approach is similar to RecNet [1], the performance of ADL is better than RecNet [1], which also contained a video reconstruction module, for video captioning. This is because the attention mechanism in the proposed ADL approach can capture important information from videos to generate accurate video captions.

On the MSR-VTT dataset, we compared the proposed ADL approach with several classical encoder–decoder approaches and the state-of-the-art approaches, including MP-LSTM [1], SA-LSTM [1], and RecNet [1], for video captioning. Table 2 illustrates the quantitative experimental results of these approaches on the MSR-VTT dataset. When using the same encoder (such as AlexNet), the performance of SA-LSTM is better than MP-LSTM. This is because MP-LSTM utilized mean-pooling for frame feature aggregation/fusion, while SA-LSTM was based on an attention mechanism for feature fusion. Furthermore, compared to other SA-LSTMs that utilized AlexNet, GoogLeNet, or VGG19 as the encoder, SA-LSTM that utilized Inception-V4 as the encoder produced the best captioning results. This is because Inception-V4 is deeper than other networks, and is good at extracting advanced semantic features from videos. Therefore, our ADL approach with Inception-V4 is superior to other reference approaches.

Compared with the results of other reference approaches, our proposed ADL approach leverages the strength of dual learning in video caption generation to improve all evaluation scores. Therefore, our approach can generate accurate captions from videos. Fig. 2 shows qualitative examples of video captions generated by our approach. We compare the generated captions with the ground truths (GT). According to the comparison results, our approach can generate keywords describing the content of a video and combine these keywords into a complete sentence. In other words, our approach can generate video captions to describe video content accurately and concisely.

5. Conclusion

Video captioning aims to produce natural language sentences from videos, which has been widely used to solve real-world problems, such as explaining a movie plot to the blind. This paper proposes a novel attention based dual learning approach (ADL) for video captioning. ADL consists of two modules: a caption generation module that can generate video captions from raw video frames, and a video reconstruction module that can reproduce raw video frames based on the generated video captions. A multi-head attention mechanism is adapted to the two modules

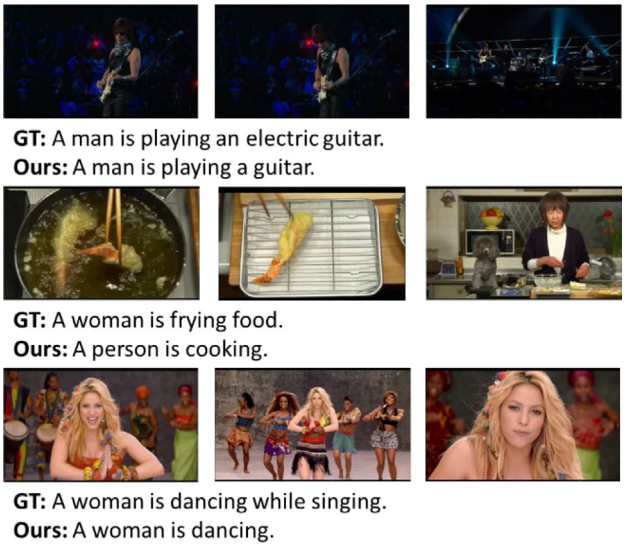

Fig. 2. Qualitative examples of video captions generated by our approach. We compared the generated captions with the ground truths (GT).

to capture the most effective information, and a dual learning mechanism is used to fine-tune the two modules. Therefore, the proposed approach is able to minimize the semantic gap between raw videos and generated captions by decreasing the differences between the reproduced and raw video features.

We compare the proposed approach with several video captioning approaches on two benchmark datasets. Experimental results demonstrate the superiority of our approach over the state-of-the-art video captioning approaches. Our research also verifies the effectiveness of multi-head attention based dual learning for generating high-quality video captions.

The proposed approach also can be further improved. For example, we simply use multi-head attention rather than developing a new attention mechanism for capturing information from videos and captions. For future work, we intend to develop more appropriate attention mechanisms and better video captioning approaches for video caption generation. Based on our previous research [59–62], we will develop new deep neural networks. We will also explore the use of other information such as audio and semantic information for video caption generation. In addition, we intend to apply video captioning to a wider application field for solving real-world problems.

CRediT authorship contribution statement

Wanting Ji: Conceptualization, Methodology, Formal analysis, Writing – original draft preparation, Writing – review & editing. **Ruili Wang:** Supervision, Investigation, Writing – review & editing. **Yan Tian:** Visualization, Data curation, Writing - review & editing. **Xun Wang:** Resources, Writing - review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported in part by the National Key Research and Development Program of China (2018YFB1404102), National Natural Science Foundation of China (61976188, 61972351, 62111530300), and the Natural Science Foundation of Zhejiang Province, China (LY18F020008).

References

[1] B. Wang, L. Ma, W. Zhang, W. Liu, Reconstruction network for video captioning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7622–7631.
[2] J. Wang, W. Wang, Y. Huang, L. Wang, Tieniu. Tan, M3: Multimodal memory modelling for video captioning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7512–7520.
[3] C. Yan, Y. Tu, X. Wang, Y. Zhang, X. Hao, Y. Zhang, Q. Dai, STAT: SPatial–temporal attention mechanism for video captioning, IEEE Trans. Multimed. (2019) http://dx.doi.org/10.1109/TMM.2019.2924576.
[4] J. Li, Y. Wong, Q. Zhao, M.S. Kankanhalli, Video storytelling, 2018, arXiv preprint arXiv:1807.09418.
[5] J. Li, Y. Wong, Q. Zhao, M.S. Kankanhalli, Video storytelling: Textual summaries for events, IEEE Trans. Multimed. (2019) http://dx.doi.org/10.1109/TMM.2019.2930041.
[6] F. Hou, R. Wang, J. He, Y. Zhou, Improving entity linking even further through semantic reinforced entity embeddings, in: The Annual Conference of the Association for Computational Linguistics, 2020, pp.1–8.
[7] Z. Liu, Z. Li, R. Wang, M. Zong, W. Ji, Spatiotemporal saliency-based multi-stream networks with attention-aware LSTM for action recognition, Neural Comput. Appl. 32 (18) 14593–14602.
[8] H. Zheng, R. Wang, W. Ji, M. Zong, W.K. Wong, Z. Lai, H. Lv, Discriminative deep multi-task learning for facial expression recognition, Inform. Sci. (2020) http://dx.doi.org/10.1016/j.ins.2020.04.041.
[9] S. Guadarrama, N. Krishnamoorthy, G. Malkarnenkar, S. Venugopalan, R. Mooney, T. Darrell, K. Saenko, Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition, in: Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 2712–2719.
[10] F. Xiao, X. Gong, Y. Zhang, Y. Shen, J. Li, X. Gao, DAA: Dual LSTMs with adaptive attention for image captioning, Neurocomputing 364 (2019) 322–329.
[11] H. Wang, H. Wang, K. Xu, Evolutionary recurrent neural network for image captioning, Neurocomputing 401 (2020) 249–256.
[12] R. Li, H. Liang, Y. Shi, F. Feng, X. Wang, Dual-CNN: A convolutional language decoder for paragraph image captioning, Neurocomputing 396 (2020) 92–101.
[13] Z. Wu, T. Yao, Y. Fu, Y. Jiang, Deep learning for video classification and captioning, 2016, arXiv preprint arXiv:1609.06782.
[14] M. Varges Da Silva, A.N. Marana, Human action recognition in videos based on spatiotemporal features and bag-of-poses, Appl. Soft Comput. 95 (2020) 106513.
[15] J.F. Connolly, E. Granger, R. Sabourin, Dynamic multi-objective evolution of classifier ensembles for video face recognition, Appl. Soft Comput. 13 (6) (2013) 3149–3166.
[16] A. Kojima, T. Tamura, K. Fukunaga, Natural language description of human activities from video images based on concept hierarchy of actions, Int. J. Comput. Vis. 50 (2) (2002) 171–184.
[17] M. Rohrbach, W. Qiu, I. Titov, S. Thater, M. Pinkal, B. Schiele, Translating video content to natural language descriptions, in: Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 433–440.
[18] R. Xu, C. Xiong, W. Chen, J.J. Corso, Jointly modeling deep video and compositional text to bridge vision and language in a unified framework, in: The 29th AAAI Conference on Artificial Intelligence, 2015, pp. 1–7.
[19] W. Ji, R. Wang, A multi-instance multi-label dual learning approach for video captioning, ACM Trans. Multimid. Comput. Commun. Appl. 17 (2s) (2021) 1–18.
[20] Y. Wu, X. Ji, W. Ji, Y. Tian, H. Zhou, CASR: A Context-aware residual network for single-image super-resolution, Neural Comput. Appl. (2019) http://dx.doi.org/10.1007/s00521-019-04609-8.
[21] M. Zong, R. Wang, Z. Chen, M. Wang, X. Wang, J. Potgieter, Multi-cue based 3D residual network for action recognition, Neural Comput. Appl. (2020) 1–12.
[22] T. Jin, Y. Li, Z. Zhang, Recurrent convolutional video captioning with global and local attention, Neurocomputing 370 (2019) 118–127.
[23] W. Zhao, W. Xu, M. Yang, J. Ye, Z. Zhao, Y. Feng, Y. Qiao, Dual learning for cross-domain image captioning, in: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, 2017, pp. 29–38.
[24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems, 2017, pp. 5998–6008.
[25] C. Szegedy, S. Ioffe, V. Vanhoucke, A.A. Alemi, Inception-V4, inception-resnet and the impact of residual connections on learning, in: The 31st AAAI Conference on Artificial Intelligence, 2017, pp. 1–7.
[26] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, K. Saenko, Translating videos to natural language using deep recurrent neural networks, 2014, arXiv preprint arXiv:1412.4729.
[27] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, K. Saenko, Sequence to sequence-video to text, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 4534–4542.

[28] C. Zhang, Y. Tian, Automatic video description generation via LSTM with joint two-stream encoding, in: The 23rd International Conference on Pattern Recognition, 2016, pp. 2924–2929.
[29] Y. Pan, T. Mei, T. Yao, H. Li, Y. Rui, Jointly modeling embedding and translation to bridge video and language, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4594–4602.
[30] N. Ballas, L. Yao, C. Pal, A. Courville, Delving deeper into convolutional networks for learning video representations, 2015, arXiv preprint arXiv:1511.06432.
[31] P. Pan, Z. Xu, Y. Yang, F. Wu, Y. Zhuang, Hierarchical recurrent neural encoder for video representation with application to captioning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1029–1038.
[32] H. Yu, J. Wang, Z. Huang, Y. Yang, W. Xu, Video paragraph captioning using hierarchical recurrent neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4584–4593.
[33] Y. Liu, X. Li, Z. Shi, Video captioning with listwise supervision, in: The 31st AAAI Conference on Artificial Intelligence, 2017, pp. 1–7.
[34] L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, A. Courville, Describing videos by exploiting temporal structure, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 4507–4515.
[35] L. Gao, Z. Guo, H. Zhang, X. Xu, H.T. Shen, Video captioning with attention-based LSTM and semantic consistency, IEEE Trans. Multimed. 19 (9) (2017) 2045–2055.
[36] D. He, Y. Xia, T. Qin, L. Wang, N. Yu, T. Liu, W. Ma, Dual learning for machine translation, in: Advances in Neural Information Processing Systems, 2016, pp. 820–828.
[37] Y. Wang, Y. Xia, L. Zhao, J. Bian, T. Qin, G. Liu, T. Liu, Dual transfer learning for neural machine translation with marginal distribution regularization, in: The 32nd AAAI Conference on Artificial Intelligence, 2018, pp. 1–7.
[38] G. Lample, A. Conneau, L. Denoyer, M.A. Ranzato, Unsupervised machine translation using monolingual corpora only, 2017, arXiv preprint arXiv:1711.00043.
[39] M. Artetxe, G. Labaka, E. Agirre, K. Cho, Unsupervised neural machine translation, 2017, arXiv preprint arXiv:1710.11041.
[40] Z. Yi, H. Zhang, P. Tan, M. Gong, Dualgan: Unsupervised dual learning for image-to-image translation, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2849–2857.
[41] T. Kim, M. Cha, H. Kim, J.K. Lee, J. Kim, Learning to discover cross-domain relations with generative adversarial networks, in: Proceedings of the 34th International Conference on Machine Learning, vol. 70, 2017, pp. 1857–1865.
[42] J. Zhu, T. Park, P. Isola, A.A. Efros, Unpaired image-to-image translation using cycle-consistent adversarial networks, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2223–2232.
[43] Y. Xia, J. Bian, T. Qin, N. Yu, T. Liu, Dual inference for machine learning, in: International Joint Conferences on Artificial Intelligence, 2017, pp. 3112–3118.
[44] P. Luo, G. Wang, L. Lin, X. Wang, Deep dual learning for semantic image segmentation, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2718–2726.
[45] Y. Wang, Y. Xia, T. He, F. Tian, T. Qin, C. Zhai, T. Liu, Multi-agent dual learning, in: Proceedings of the International Conference on Learning Representations, 2019, pp. 1–15.
[46] Z. Zhao, Y. Xia, T. Qin, T. Liu, Dual learning: Theoretical study and algorithmic extensions, in: Proceedings of the International Conference on Learning Representations, 2019, pp. 1–16.
[47] Y. Xia, T. Qin, W. Chen, J. Bian, N. Yu, T. Liu, Dual supervised learning, in: Proceedings of the 34th International Conference on Machine Learning, vol. 70, 2017, pp. 3789–3798.
[48] Y. Xia, X. Tan, F. Tian, T. Qin, N. Yu, T. Liu, Model-level dual learning, in: International Conference on Machine Learning, 2018, pp. 5383–5392.
[49] J. Xu, T. Mei, T. Yao, Y. Rui, MSR-VTT: A large video description dataset for bridging video and language, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 5288–5296.
[50] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems, 2017, pp. 5998–6008.
[51] D.L. Chen, W.B. Dolan, Collecting highly parallel data for paraphrase evaluation, in: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, vol. 1, 2011, pp. 190–200.
[52] S. Banerjee, A. Lavie, 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, 2005, pp. 65–72.
[53] K. Papineni, S. Roukos, T. Ward, W. Zhu, BLEU: A method for automatic evaluation of machine translation, in: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, 2002, pp. 311–318.
[54] C. Lin, Rouge: A package for automatic evaluation of summaries, in: Text Summarization Branches Out, 2004, pp. 74–81.
[55] R. Vedantam, C.L. Zitnick, D. Parikh, Cider: Consensus-based image description evaluation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 4566–4575.
[56] X. Chen, H. Fang, T. Lin, R. Vedantam, S. Gupta, P. Dollár, C.L. Zitnick, Microsoft coco captions: Data collection and evaluation server, 2015, arXiv preprint arXiv:1504.00325.
[57] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, et al., ImageNet large scale visual recognition challenge, Int. J. Comput. Vis. 115 (3) (2015) 211–252.
[58] M.D. Zeiler, ADADELTA: AN adaptive learning rate method, 2012, arXiv preprint arXiv:1212.5701.
[59] H. Wang, Y. Gao, Y. Shi, R. Wang, Group-based alternating direction method of multipliers for distributed linear classification, IEEE Trans. Cybern. 47 (11) (2016) 3568–3582.
[60] P. Shamsolmoali, M. Zareapoor, R. Wang, H. Zhou, J. Yang, A novel deep structure u-net for sea-land segmentation in remote sensing images, IEEE J. Sel. Top. Appl. Earth Obs. Remote (2020).
[61] Z. Chen, R. Wang, Z. Zhang, H. Wang, L. Xu, Background-foreground interaction for moving object detection in dynamic scenes, Inform. Sci. 483 (2019) 65–81.
[62] J. Guo, P. Yi, R. Wang, Q. Ye, C. Zhao, Feature selection for least squares projection twin support vector machine, Neurocomputing 144 (2014) 174–183.