



Visual and language semantic hybrid enhancement and complementary for video description

Pengjie Tang^{1,2} · Yunlan Tan^{1,2} · Wenlang Luo^{1,2}

Received: 29 March 2021 / Accepted: 2 November 2021 / Published online: 20 January 2022
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2021

Abstract

It is a fundamental task of computer vision to describe and express the visual content of a video in natural language, which not only highly summarizes the video, but also presents the visual information in description sentence with reasonable pattern, correct grammars and decent words. The task has wide potential application in early education, visual aids, automatic interpretation and human–machine environment development. Nowadays, there are a variety of effective models for video description with the help of deep learning. However, the visual or language semantics is frequently mined alone, and the visual and language information cannot be complemented each other, resulting in that the accuracy and semantics of the generated sentence are difficult to be further improved. Facing the challenge, a framework for video description with visual and language semantic hybrid enhancing and complementary is proposed in this work. In detail, the language and visual semantics enhancing branches are integrated with the multimodal feature-based module firstly. Then a multi-objective jointly training strategy is employed for model optimization. Finally, the output probabilities from the three branches are fused with the weighted average for word prediction at each time step. Additionally, the language and visual semantics enhancing-based deep fusion modules are combined together with the same jointly training and sequential probabilities fusion for further performance improving. The experimental results on MSVD and MSR-VTT2016 datasets demonstrate the effectiveness of the proposed models, with the performance of proposed models outperforming the baseline model Deep-Glove (which is denoted as E-LSC for simplification and comparison) greatly and achieving competitive performance compared to the state-of-the-art methods. In particular, the BLEU4 and CIDEr reach 52.4 and 81.5, respectively, on MSVD with the proposed HE-VLSC[#] model.

Keywords Video description · Visual semantics enhancing · Language enhancing · LSTM · CNN

1 Introduction

Video description is to translate video content including person/object, action, scene and even relationship into natural language with reasonable sentence pattern, correct grammars and decent words. It belongs to the task of high-level visual understanding and possesses broad prospect such as early education, automatic interpretation, visual aids and human–machine intelligent environment development. However, the task is really challenging since the model has to bridge the vision and language which belong to different modal information and there is great semantic gap. Facing the challenge, the object detection, action recognition techniques in computer vision are usually employed to extract the visual semantic entities or regions firstly, and then, they are filled into a pre-designed template

✉ Yunlan Tan
tanyunlan@jgsu.edu.cn

Pengjie Tang
tangpengjie@jgsu.edu.cn

Wenlang Luo
wen8102011@126.com

¹ Electronics and Information Engineering College, Jingtangshan University, Ji'an 343009, People's Republic of China

² Jiangxi Engineering Laboratory of IoT Technology for Crop Growth, Jingtangshan University, Ji'an 343009, People's Republic of China

such as “S–O–V” for sentence [1, 2], or the detected visual semantic concepts are composed of sentence with special rules [3]. However, the generated sentences are frequently with unsatisfying accuracy, coherence and flexibility caused by the limitations of inflexibility template or rule and weak representative ability of handcraft visual feature.

Recently, the deep convolutional neural networks (DCNN) are usually employed to extract visual deep feature which often possesses a higher level of abstraction and more powerful representation. And a number of DCNN models like AlexNet [4], VGGNet [5], GoogLeNet [6] and ResNet [7] are developed and applied on various visual tasks including image classification [4–7], action recognition [8, 9] as well as image captioning [10–13]. Also, the breakthrough of deep learning offers an opportunity to video description for performance improvement [14–19]. Generally speaking, the “encoding–decoding” pipeline for machine translation is learned and employed for the video description. The 2D/3D DCNN models pre-trained on large-scale dataset such as ImageNet [20] are often used to extract video feature firstly. Then the visual feature is usually fed into recurrent neural networks (RNN) for motion feature encoding and language modeling. The generated sentences are generally more accuracy and richer semantics in the help of deep feature and flexible language decoder. Hereafter, the attention mechanism [21–24] and variants (*e.g.*, Transformer) [25, 26], visual attributes/concepts [27, 28] are employed for visual feature selection and optimization. Additionally, advanced optimization strategies such as reinforcement learning [29, 30] and adversarial learning [31] are also applied to generate a description of videos under different circumstances.

However, the visual semantics or language feature is frequently mined and used alone in popular works, leading to the insufficient available information for performance improving. In order to solve the problem, a framework with hybrid enhancing and complementary of visual and language semantics is proposed in this work, where two extra branches including visual and language semantics enhancing are appended on a multimodal model for compensation of visual and language information. In addition, the multi-objective optimization strategy is employed to provide more regularization information for the model training by adding objective functions on the aforementioned three branches. During testing, the output probabilities of all the branches are fused by weight average for word prediction at each time step for further improving sentence quality. For simplification, if the two extra visual and language branches are fused with the multimodal module by concatenation, the model is denoted as HE-VLSC. And if the element-wise addition operation is employed for the fusion, the model is abbreviated as HE-VLSA. Besides, a variant of the proposed HE-VLSC/A is

designed to further boost performance, where the module based on deep fusion and module based on visual semantics enhancing are integrated. Two objective functions are designed to optimize the model jointly. Similar to HE-VLSC/A, the sequential probability fusion is employed during the test stage with weight average. As an example, the proposed HE-VLSA framework is presented in Fig. 1.

Experiments are conducted on two public video description datasets including MSVD [2, 32] and MSR-VTT2016 [33] and better performance is achieved compared to the baseline model, which reveals the effectiveness of the proposed model. Also, competitive results are obtained compared to the other state-of-the-art methods on a few evaluation metrics. The contributions of this work can be concluded as below.

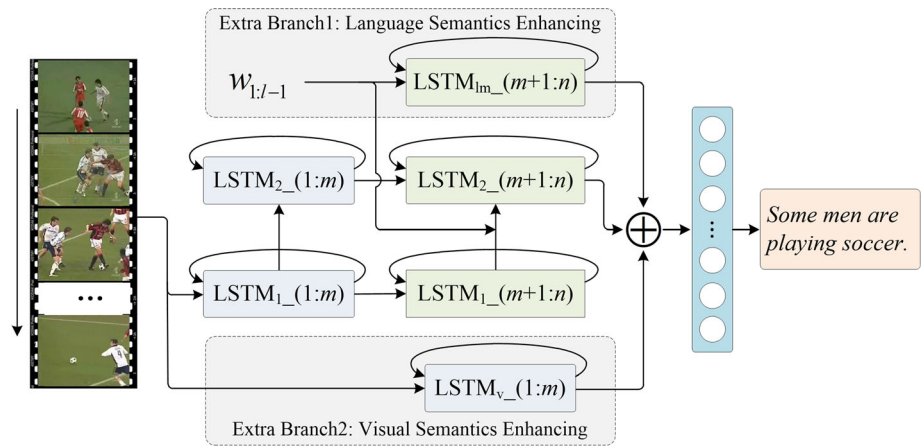
- A model based on visual semantics enhancing for video description is proposed in this work, where the visual information is modeled separately for further mining of motion feature, and then, the module is appended on a multimodal-based module for improving the accuracy of word prediction.
- A framework based on hybrid enhancing and complementary of visual and language semantics is developed for video description, in which two branches for visual and language semantics enhancing are integrated with the multimodal-based module for more powerful representation, and multi-objective optimization and sequential late fusion are employed for model training and sentence generation.
- Two deep fusion-based modules including visual and language semantics enhancing are combined into a wide model for video description, where joint optimization and sequential late fusion are also employed for training and testing, respectively. And competitive results are achieved compared to the state-of-the-art methods on two public evaluation datasets.

The rest of this work is organized as follows: Sect. 2 introduces the related works about video description briefly. Section 3 describes the motivation, proposed models and related formulas of this work in detail. The experiments and result discussion are provided in Sect. 4. And Sect. 5 concludes all this work.

2 Related works

The task of video description has been researched for decades. In early days, the techniques of computer vision such as object detection, action recognition are usually employed to extract visual semantic entities, and then, the related words or phrases are filled into templates designed in advance or reconstituted into the sentence in line with

Fig. 1 Overview of the proposed HE-VLSA framework. Two extra branches including visual semantics enhancing and language semantics enhancing are appended on a multimodal model for comprehensive information capturing



rules. Nagel et al. generate description for vehicle's condition according to composing the detected motion track and type to phrases by a few rules [34]. Gupta et al. propose a model to generate long description for a video, where the actions or events are detected firstly, following that the visual entities are described as simple sentences, and then, they are recomposed to detail description by the relationships among the actions or events [3]. Kojima et al. construct the mapping relationships between semantic objects (*e.g.*, object, action) and specific concepts, and fill them into the templates for sentences according to corresponding syntactic components [1].

Inspired by the pipeline of machine translation, the “encoder–decoder” framework is employed to generate descriptions for videos, where the videos remained to be described are treated as “source language” and the generated sentences are regarded as “target language.” Rohrbach et al. propose a model for video description with flexible structure, where the conditional random field (CRF) is employed to model the relationships of different detected objects [35]. However, the handcraft feature and traditional object detection method in the work limit the performance improvement caused by the insufficient accuracy of recognition and detection. The performance breakthrough of visual classification and recognition with deep learning provides another inspiration for researchers to develop advanced models for video description. In the general pipeline, the 2D/3D DCNN is employed to extract more abstract and representative deep features of frames or clips in a video for visual encoding, and then, the RNN is usually applied to generate sentence for decoding.

Venugopalan et al. propose a visual feature mean pooling-based framework for video captioning, in which the sampled frames are fed to a DCNN model for deep features; hereafter, the element-wise mean feature vector is calculated and used as the visual representation to send to a recurrent neural network named long short-term

memory (LSTM) for words prediction [36]. The method makes use of the information of all sampled frames, but the motion and semantic structure information in the video may be destroyed. Afterward, they learn from the sequence to sequence method in machine translation and propose S2VT model [14], where the sampled frames are given to a CNN model in sequence, and then, the extracted feature sequence is fed to two stacked LSTM network for motion information capturing. When all the features of the frame are exhausted, the final visual representation is transmitted to the next time step in the same LSTM network for decoding. The model forces the motion feature encoding and language decoding to share the same network and parameters, and hence, it possesses concise architecture and good interpretability. Additionally, they develop a Deep-Glove model for further discover language information [37]. In the model, language branch pre-trained on large-scale corpus is added on S2VT to improve the accuracy and semantics of the generated sentences. Shen et al. borrow the fully convolutional network (FCN) [38] to extract CNN representation of visual semantic regions in frames of a video. Then the motion of the visual objects is tracked and a description is generated [39]. Besides, the event is usually employed as the elementary semantic object and description unit in a few works [40–42], and a number of sentences are generated to describe a video in more detail. Mun et al. argue that different events in a video are often interrelated, and thus, the independent event descriptions should be recomposed to paragraph with reasonable logic and the redundant sentences will be truncated [43]. In this work, the S2VT [14] and Deep-Glove [37] are investigated and employed to implement the proposed idea in consideration of the simple architecture and excellent performance for video captioning and description.

In addition to the sequence to sequence with RNN for visual feature encoding, the 3D CNN is another solution for visual representation for serving video description. In general, the method is integrated with RNN, attention

mechanism, hierarchical modeling, advanced memory unit and multi-objective joint optimization for taking advantage of visual information. Yao et al. extract local temporal structure features of videos with a 3D CNN model. The visual feature is then fed to LSTM for decoding. In the process, an attention mechanism is employed to assign different weights to different 3D CNN features at different time steps for guiding sentence generation [16]. Pan et al. employ the C3D model to extract visual features for clips in a video, then they are sent to LSTM network with the 2D CNN mean feature of all frames. Finally, the visual module and language module are jointly training for model optimization [15, 28]. Wang et al. present a hierarchical framework based on reinforcement learning and 3D CNN for video fine-grained description [30]. Additionally, a few other researchers pay attention to visual objects and their corresponding relationships in videos, and develop a series of effective models, in which 2D and 3D visual features are usually both employed [44–46], to complete the task of video captioning. Different from the aforementioned works, Park et al. adopt an adversarial learning strategy and develop a group of discriminators to evaluate the accuracy, relevance of candidate sentences from the generator [47].

The popular works always focus on a single aspect of visual semantics or language feature mining separately for better performance but ignore the use of both and the mutual supplement between them during training and testing. As an example, the S2VT model [14] improves the flexibility of generated sentences, but both the visual semantic information and language are not fully exploited. In Deep-Glove model [37], the language information is further discovered by an extra branch. However, the language branch is difficult to be pre-trained since the large-scale corpus is required. Also, the visual semantic information is not further mined and put into full use. In this work, a framework based on hybrid enhancement and complementary of visual and language semantics is proposed for video description. For comprehensive visual and language information, the visual semantics enhancing and language enhancing branches are designed and appended on a multimodal-based module, making the three parts can be optimized cooperatively and complementarily. During training, besides the main objective function for the multimodal-based module, the other two extra objective functions for visual semantics enhancing branch and language enhancing branch are designed to provide more regularization information and hence improve the generalization ability of the whole model.

3 Proposed methods

Both visual semantics and language information are required in the task of video description. In most popular works, the visual and language information is usually fed to a sequential model to form multi-model representation and map the two different modalities data into a unified feature space. Then the RNN is employed as a language decoder and generates a description sentence for the video. However, more noises including vision and language may be introduced into the model if the multimodal feature is used, leading to interference to word prediction. Additionally, a few significant visual and language information may be lost during multiple nonlinear transformation of feature as the going on of time steps in RNN and other variants (*e.g.*, LSTM) since the activation functions (*i.e.*, Sigmoid and Tanh) may be in a saturated state once the value exceeds the sensitive interval. For richer language information and reducing the influence of noises on the model, a single language branch on a large-scale corpus is pre-trained and then integrated with S2VT by fusing the two sequential features at each time step in [37]. However, the extra domain corpus should be constructed for the language branch optimization. Moreover, the further mining and discovery of visual semantic information are ignored, and the semantics of generated sentences cannot be further improved.

Facing the problem, the visual semantic branch is designed based on S2VT [14] for richer visual information capturing in this work, which is learned from the idea in [37]. Then the branch is added on Deep-Glove [37], and the output features of the three parts are combined with early sequential fusion for comprehensive and complementary representation for sentence generation, where the features are fused before they are fed to the word predictor at each time step. Additionally, the multi-objective optimization strategy is employed to prevent the model to stick to overfitting caused by the increase of parameters in visual and language branches. In detail, each branch of visual semantics enhancing, language semantics enhancing and the multimodal-based module deploys an objective function. Because there is no enough available modal information in the extra added branches (*e.g.*, the visual information is missing in language enhancing branch), the larger errors between ground truth and generated sentence may be produced, as well as the disturbance during model optimization, which can be treated as extra regularization for the whole model. At test time, the late sequential fusion is employed for word prediction, where the probabilities from the predictors in the three modules are fused with the weighted average.

Suppose there are m frames in a video, and the frame sequence is $\langle f_1, f_2, \dots, f_m \rangle$. The corresponding feature sequence extracted from CNN model is $\langle v_1, v_2, \dots, v_m \rangle$. The word sequence in one of the corresponding sentences denoted as $\langle w_1, w_2, \dots, w_k \rangle$, where k is the length of the sentence. During training, the object of the model is to predict the word sequence $\langle w'_1, w'_2, \dots, w'_k \rangle$ under the condition of $\langle v_1, v_2, \dots, v_m \rangle$. Afterward, the errors between the output sequence and reference are calculated and back-propagated through time (BPTT) for updating and optimizing the parameter set Θ . The objective function can be formulated as:

$$\arg \min_{\Theta} \mathcal{L}(\langle w'_1, w'_2, \dots, w'_k \rangle | \langle v_1, v_2, \dots, v_m \rangle, \Theta), \tag{1}$$

where $\mathcal{L}(\cdot)$ is the loss function, where the cross-entropy strategy is usually employed. And for different models, the function may possess different formulas (e.g., Eq. (2) and Eq. (7) in Sects. 3.1 and 3.3, respectively). As an important module in the backbone (Deep-Glove [37]) of the proposed model, the S2VT is presented in Fig. 2. The visual feature sequence $\langle v_1, v_2, \dots, v_m \rangle$ is fed to the first LSTM layer whether at the training stage or the testing stage. As for the second LSTM, it receives $\langle pad \rangle$ for the following word inputting alignment. When all visual features are exhausted, the word sequence $\langle w_1, w_2, \dots, w_k \rangle$ is fed to the second LSTM with the output from the first LSTM for modeling language. During training, each word will be given to the language model and the corresponding output word at each time step will be employed to calculate gradients with the input word, while

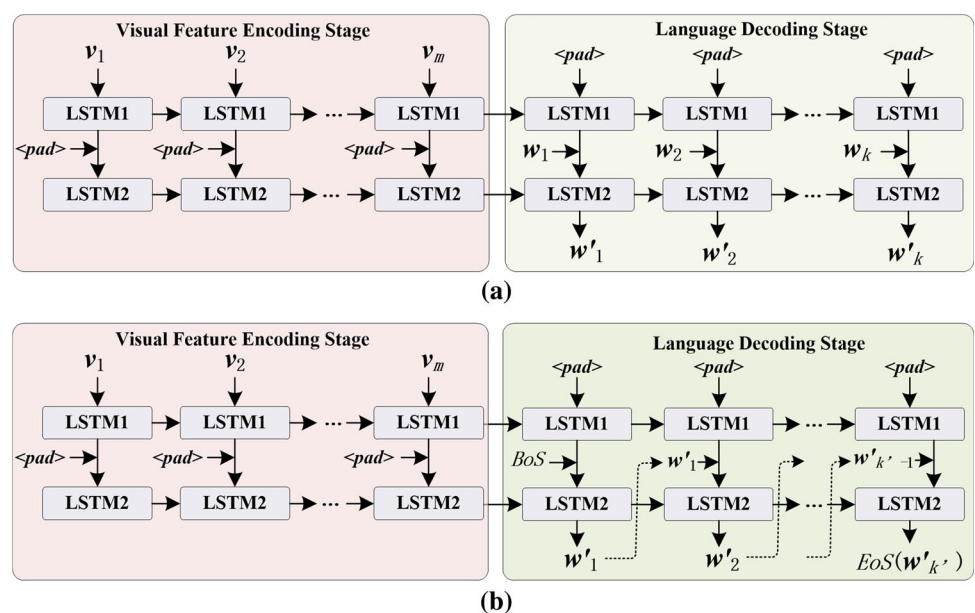
at testing stage, the output words are predicted one by one but each of them has to depend on the previous word sequence.

3.1 The visual semantics enhancing-based model

It is the basis of visual features for video description. In the multimodal-based model, the visual features of frames are extracted from a CNN model pre-trained on large-scale dataset, then they are fed to LSTM by sequence for motion feature encoding. The final visual representation is delivered to the following time steps for language decoding when all the visual features are exhausted. In fact, the visual and language features are combined together in the pipeline. However, the visual feature may be compromised as the language feature dominates the word prediction at each time step in the decoding stage. Particularly, more and more visual details will be lost during nonlinear transformation of visual feature as the time step advances in LSTM, leading to increasing of cumulative error and decreasing of accuracy and semantics of the generated sentence. For richer language information, Venugopalan et al. [37] develop an extra single branch to enhance language semantics and then integrate it with S2VT [14] for more powerful representation and the following improvement of generated sentence. In this work, we learn from the practice but add a single branch for further visual semantics enhancing and fuse the output feature with that from S2VT to enhance the final representation and improve the accuracy of word prediction.

As shown in Fig. 3, it is the architecture overview of E-VSA. The visual CNN features are fed to the bottom LSTM

Fig. 2 Overview of the S2VT framework, which belongs to “encoding–decoding” pipeline. The **a** is training stage, where the visual feature sequence $\langle v_1, v_2, \dots, v_m \rangle$ is fed to the first LSTM during visual encoding, and the word sequence $\langle w_1, w_2, \dots, w_k \rangle$ is fed to the second LSTM with the output from the first LSTM. And the **b** is for testing, where the predicted word at current time step depends on the previous generated word sequence



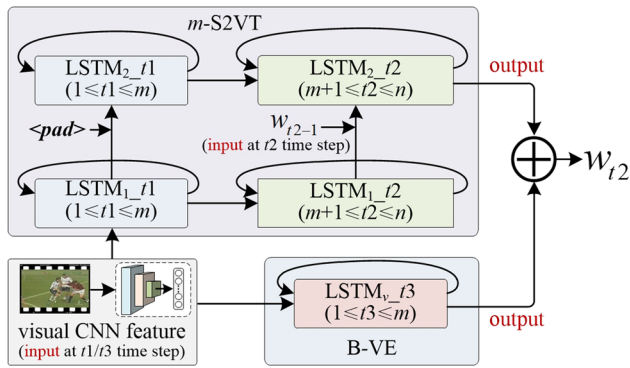


Fig. 3 Visual semantics enhancing-based model for video description (E-VSA). The sequence of visual feature is fed to both LSTM_v and LSTM_l for visual motion feature modeling, and the outputs of LSTM_v and LSTM₂ are fused for word prediction

in multimodal-based module (the module is marked as m-S2VT for simplification) and the single visual semantics enhancing branch (the branch is denoted as B-VE for simplification) at the same time. At test time, the outputs of B-VE and m-S2VT are fused by the early sequential fusion method. For m-S2VT, there are two stages including visual “encoding” and language “decoding,” and the number of all time steps in the LSTM model is denoted as n . During visual encoding, suppose the number of sampled frames in a video is m , the time step is denoted as $t1 (1 \leq t1 \leq m)$. At the decoding stage, the length of the corresponding reference sentence is $n - m + 1$, and the time step is $t2 (m + 1 \leq t2 \leq n)$. As for the extra visual semantics enhancing branch, the time step is denoted by $t3$. It is worth noting that $1 \leq t3 \leq m$ in B-VE since the branch models the motion feature of the same video. If $m \equiv n - m + 1$, the outputs of B-VE and m-S2VT can be fused completely at each time step. However, when $m < n - m + 1$, the $\langle pad \rangle$ is used as the missing output from B-VE of rest time steps, while if $m > n - m + 1$, let $m = n - m + 1$, and the extra frame features will be truncated. In practice for feature fusing, two methods are employed, including feature concatenation (E-VSC) and element-wise addition (E-VSA). When E-VSC is employed, the model focuses on the balance of information including vision and language. In particular, the generated sentence can be further improved in accuracy and semantics if the used CNN feature possesses weaker representative ability (e.g., GoogLeNet [7] feature). On the contrary, when the more abstract feature (e.g., ResNet152 [6] feature) is used, the corresponding sensitivity of useful information can be enhanced when E-VSA is employed.

For E-VSC/A, the loss function is denoted as $\mathcal{L}_{fusion}^{vs}$, which can be calculated by

$$\mathcal{L}_{fusion}^{vs}(V, W, \Theta_{vs}) = \frac{1}{|V|} \sum_{i=1}^{|V|} \sum_{t=1}^{k'} \log \times \mathcal{P}(w'_t | (h_{m+t-1}^{vs_fusion}, w'_{1:t-1}); \Theta_{vs}), \quad (2)$$

where $V = \langle v_1^1, \dots, v_{m_1}^1 \rangle, \langle v_1^2, \dots, v_{m_2}^2 \rangle, \dots, \langle v_1^{|V|}, \dots, v_{m_{|V|}}^{|V|} \rangle$ is the set of visual feature vector for the model in one iteration, and the $|V|$ is the size of the set. The $m_i (i \in \{1, 2, \dots, |V|\})$ is the number of sampled frames in the i^{th} video. And $\Theta_{vs} = (\Theta_{m-s2vt}, \Theta_v)$ is the parameter set, where Θ_{m-s2vt} and Θ_v stand for the parameter set of m-S2VT and B-VE, respectively. The $h_{m+t-1}^{vs_fusion}$ denotes the fusion state of outputs from the hidden state of LSTM in B-VE and that from the top LSTM in m-S2VT at $m + t - 1$ time step, which can be written as

$$h_{m+t-1}^{vs_fusion} = \mathcal{F}(h_{m+t-1}^2, h_{m+t-1}^v). \quad (3)$$

In the above equation, the h_{m+t-1}^2 is the output of the top LSTM in m-S2VT, while h_{m+t-1}^v is the output from LSTM in B-VE branch. The function $\mathcal{F}(\cdot)$ is fusion operation. Take element-wise addition as an example, the formulation is

$$\mathcal{F}(h_{m+t-1}^2, h_{m+t-1}^v) = h_{m+t-1}^2 \oplus h_{m+t-1}^v. \quad (4)$$

Afterward, the fusion feature is fed to the classification layer for word prediction. At $m + t - 1$ time step, the output probabilities can be calculated by a Softmax function $\mathcal{P}(\cdot)$ with the formula as

$$\mathcal{P}(w'_t | h_l^{vs_fusion}) = \frac{\exp(W_{t2} \cdot h_l^{vs_fusion})}{\sum_{i=1}^{|VOC|} \exp(W_{t2} \cdot h_i^{vs_fusion})}, \quad (5)$$

where $l = m + t - 1$, $t2 \in [1, n - m - 1]$ and $t2 = t$. The W_t denotes weight vector of the hidden layer, and $|VOC|$ is the size of the vocabulary. Hereafter, the word in vocabulary corresponding to the maximum probability is picked as the predicted output at this time step.

3.2 Hybrid enhancement and complementary of visual and language semantics-based model

The proposed E-VSC/A ignores the further mining of language information though richer visual details and semantics can be achieved. For comprehensive information, the E-VSC/A and Deep-Glove framework are used for reference, and a more effective model with richer language and visual semantics is developed. Specifically, the B-VE in E-VSC/A is integrated with Deep-Glove framework [37] where an extra language enhancing branch (which is abbreviated as B-LE (Branch for Language semantics Enhancing) for simplification) is added on m-S2VT, to

form a wide and comprehensive architecture for visual and language complementary. The output features from B-VE, m-S2VT and B-LE are fused together by early sequential fusion strategy to enhance representative ability of the final feature at the training stage. During the testing, the output probability from the m-S2VT is used for word prediction. The model is named HE-VLSC/A for convenience.

Figure 4 shows the overview of HE-VLSA. The previous predicted word w_{t-1} will be fed to the top LSTM (LSTM_{2-t2}) in m-S2VT module and language enhancing branch (B-LE) at the $t2$ time step. During training, the w_{t-1} is the previous word from the reference sentence. And at test time, it is the previous predicted word. For the three branches including B-VE, m-S2VT and B-LE, the output features from respective hidden states are fused with concatenation or element-wise addition at each time step, similar to the practice in B-VSC/A. Note that the inputs for B-LE and m-S2VT can correspond to each other for all time steps since they share the same word sequence (sentence).

It is similar to E-VSC/A model, the loss function of HE-VLSC/A is denoted as $\mathcal{L}_{fusion}^{lvs}(V, W, \Theta_{lvs})$, where $\Theta_{lvs} = (\Theta_{m-s2vt}, \Theta_v, \Theta_{lm})$. The output state of fusion $h_{m+t-1}^{lvs\text{-fusion}}$ can be computed by

$$h_{m+t-1}^{lvs\text{-fusion}} = \mathcal{F}(h_{m+t-1}^2, h_{m+t-1}^v, h_{m+t-1}^{lm}), \tag{6}$$

where h_{m+t-1}^2 , h_{m+t-1}^v and h_{m+t-1}^{lm} represent the outputs from the top LSTM in m-S2VT, hidden states in B-VE and B-LE at $m + t - 1$ time step, respectively. As for $\mathcal{F}(\cdot)$, it is the function for fusion with the operation of feature concatenation or element-wise addition.

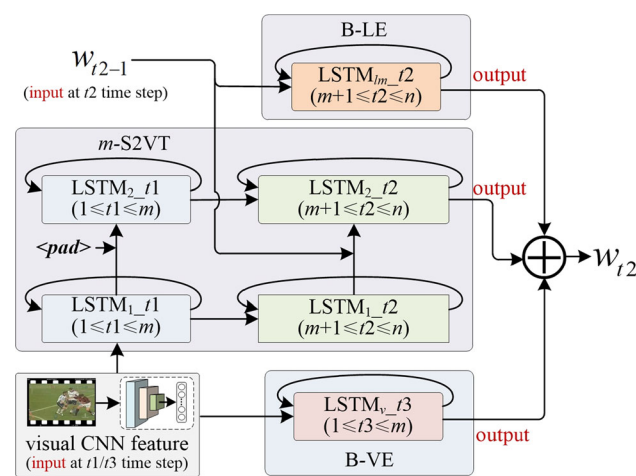


Fig. 4 Framework of hybrid enhancement and complementary of visual and language semantics (here it is an example of HE-VLSA). The output features from LSTM_v, LSTM_{lm} and LSTM₂ in m-S2VT are fused for word prediction

3.3 HE-VLSC/A with multi-objective optimization

In HE-VLSC/A model, the parameter scale will be increased caused by the two added B-VE and B-LE branches visual and language semantic enhancement, which is easy to result in over-fitting for the model. Additionally, the errors are backpropagated not only to m-S2VT, but also to the branches of B-VE and B-LE for parameter updating and optimizing. However, the inputs of B-VE and B-LE are not the complete multimodal feature (*i.e.*, only visual feature for B-VE, while only language information for B-LE), which may lead to insufficient optimization of the whole model.

In this work, two extra objective functions are appended on B-VE and B-LE branches in addition to the objective function with fusion feature. As shown in Fig. 5, it is the architecture overview of multi-objective-based HE-VLSA. Besides the L_{fusion} which is treated as the main loss function, the B-VE and B-LE are implemented their own loss functions of L_{lm} and L_v , respectively. During training, the errors from L_{fusion} are propagated back to the three modules including m-S2VT, B-VE and B-LE for parameter updating. Simultaneously, the errors of L_{lm} and L_v are fed back to their respective branches for parameter optimizing twice in them. For difference to HE-VLSC/A, the model is marked as HE-VLSC*/A*.

Because there are no corresponding language features and visual features in B-VE and B-LE, respectively, the gradients with L_{lm} and L_v are usually bigger than that with L_{fusion} . This may be detrimental to the optimization of the two branches, but the bigger errors can be treated as extra regularization information for the whole model and improve the generalization ability. In Fig. 6, the loss trend and performance on CIDEr of HE-VLSA and HE-VLSA* are presented. Figure 6a, b shows that the two models possess a similar loss trend whether the GoogLeNet [6] feature or ResNet152 [7] feature is employed, but the errors of HE-VLSA* are higher than that with HE-VLSA on the whole, which indicates that the multi-objective optimization-based HE-VLSA* is less likely to stick into the state of over-fitting. On the other hand, the performance of HE-VLSA* is superior to HE-VLSA on CIDEr [48] metric (shown in Fig. 6c, d), which reveals that HE-VLSA* enhances the generalization ability effectively and improves semantics of the generated sentences.

When the sentence is generated, the output probabilities from the fusion branch based on m-S2VT, B-VE and B-LE are sequentially fused by weight average for cooperative word prediction. In practice, the output probability from the fusion branch dominates the word selection, with the assistance of output probabilities from branches of B-VE

Fig. 5 Model of multi-objective-based HE-VLSA (HE-VLSA*). Three objective functions are employed for model training. The L_{lm} and L_v are for language semantics enhancing branch (i.e., B-LE) and visual feature enhancing branch (i.e., B-VE), respectively, while the L_{fusion} is for the multimodal model (i.e., m-S2VT)

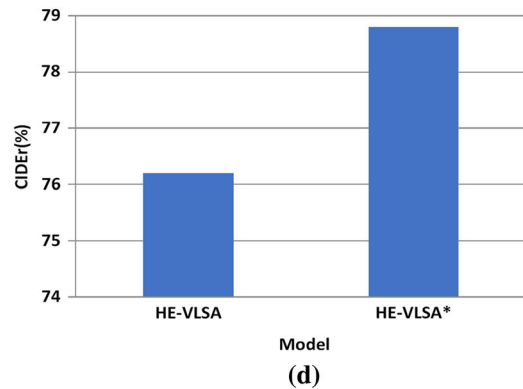
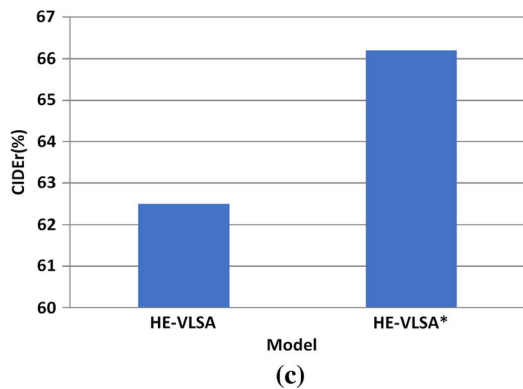
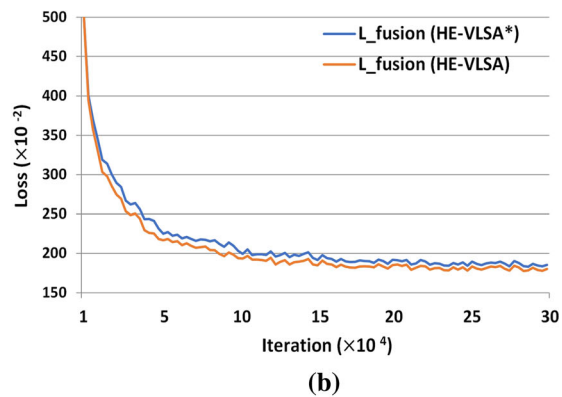
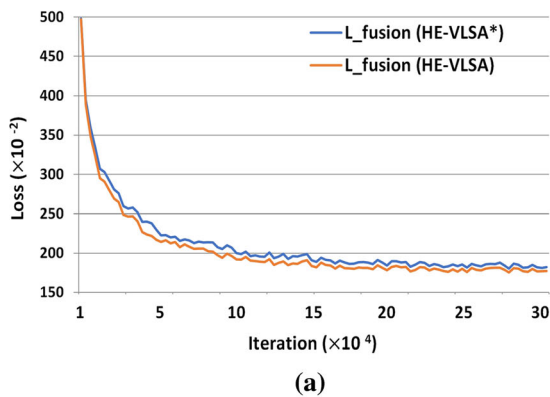
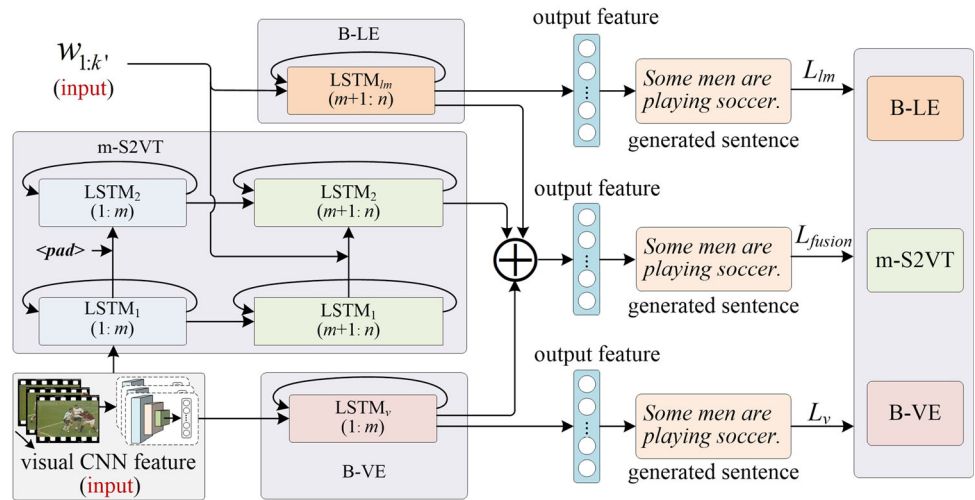


Fig. 6 Comparison of the loss trend and performance on CIDEr with HE-VLSA and HE-VLSA* (for a and c, the GoogLeNet feature is employed in model, while the ResNet152 feature is employed in the model for b and d)

and B-LE. The fusion factors can be achieved empirically in general.

For HE-VLSC*/A* model, there are three loss functions including $\mathcal{L}_{fusion}^{lvs}(V, W, \Theta_{lvs})$ for fusion branch, $\mathcal{L}^v(V, W, \Theta_v)$ for visual semantics enhancing branch B-VE and $\mathcal{L}^{lm}(W, \Theta_{lm})$ for language enhancing branch B-LE. Regarding $\mathcal{L}^{lm}(W, \Theta_{lm})$, it is blind to visual information but just provides gradients to the B-LE branch and

updating the parameters only by language information. The formulation is

$$\mathcal{L}^v(V, W, \Theta_v) = \frac{1}{|V|} \sum_{i=1}^{|V|} \log \mathcal{P}(w'_i | (h_t^v, w'_{1:t-1}); \Theta_v), \quad (7)$$

where h_t^v is the output of hidden state in B-VE at the t time step, and the previous input is CNN feature of the $t - 1$ frame. The three objective functions in HE-VLSC*/A* are

independent of each other, but the objective function in the fusion branch contains the information from B-VE and B-LE, and hence, the parameters in the two auxiliary branches are also optimized by the objective function in the fusion branch.

At test time, the output probabilities of the three branches at the t time step are fused with

$$\mathcal{P}_t^{\text{fusion}} = \alpha \mathcal{P}_t^{m-s2vt} + \beta \mathcal{P}_t^{lm} + \gamma \mathcal{P}_t^v, \tag{8}$$

where $\mathcal{P}_t^{\text{fusion}}$ is the fused probability vector, and \mathcal{P}_t^{m-s2vt} , \mathcal{P}_t^{lm} and \mathcal{P}_t^v stand for the probability vectors of fusion branch, B-LE and B-VE, respectively, at t time step. And α , β and γ are the fusion factors of the three branches, respectively, where they meet the constraint of $\alpha + \beta + \gamma \equiv 1$, and the weight assigned in practice follows experimental experience.

Besides the described HE-VLSC*/A* above, another variant with multi-objective-based model is also investigated. Specifically, the backbone of Deep-Glove is employed as one of modules for language enhancing in the framework (for convenience and distinction, it is renamed as E-LSC#/A#). And the model E-VSC/A is employed as another module for visual semantics enhancing (for convenience and distinction, it is renamed as E-VSC#/A#). Then the two modules are integrated into a wider framework, where the two modules share the same language embedding layer and visual feature reduction layer, which reduces the “one-hot” word feature into embedding language representation and is to map the CNN feature to visual representation with lower dimension, respectively. During training, for separate E-VSC#/A# and E-LSC#/A#, the two modules are optimized independently with their respective objective functions. However, it is worth noting that the two modules are actually trained jointly on the whole in that the parameter updating and optimization of the word embedding layer and visual feature reduction layer depend on both of the two objective functions. At the test stage, the late sequential fusion method is employed to fuse the output probabilities from the two modules. The fusion weights are assigned to be equal. For comparison convenience, the variant is denoted as HE-VLSC#/A#, and the architecture is shown in Fig. 7.

The optimization of HE-VLSC#/A# is similar to HE-VLSC*/A*, and the gradients from the E-LSC#/A# and E-VSC#/A# are fed back not only their own modules but to each other modules according to the language embedding layer and reduction layer, with complementary to each other. During testing, the output probabilities are fused with weight average too, where the fusion factors (which are marked as $\alpha^\#$ and $\beta^\#$) are set to be equal empirically.

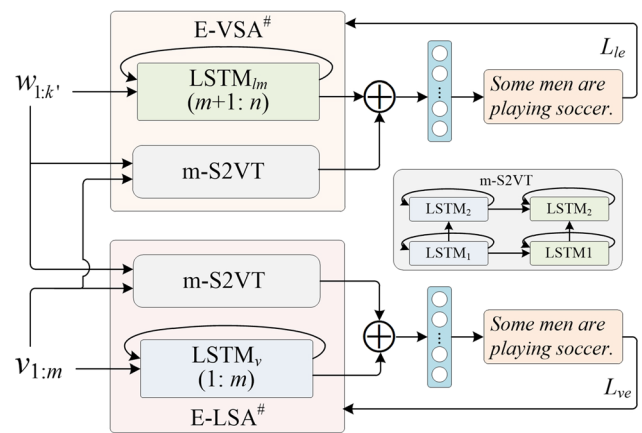


Fig. 7 Architecture of HE-VLSCA#. Two independently designed modules (i.e., language semantics enhancing branch-based module vs visual information enhancing branch-based module) are integrated for comprehensive representation. During training, the two objective functions optimize two modules jointly according to the word embedding layer. At test time, the outputs from the two modules are fused with sequential weight average fusion for final word prediction at each time step

4 Experiments

For evaluation of the proposed model, experiments are conducted on two public datasets including MSVD [2, 32] and MSR-VTT2016 [33]. In this section, the employed dataset and evaluation metric are described in detail in the first place. Then the performance of the proposed model and comparable models are presented and discussed. Concretely, the ablation study is carried on to reveal the improvement of the proposed methods on performance. Besides, the performance comparison with the popular state-of-the-art methods is conducted to show the superiority of the proposed model. Finally, a few examples including sentences generated by the proposed model and other comparable models are presented and discussed to further reveal the effectiveness of the proposed model.

4.1 Evaluation dataset and metric

Two popular datasets including MSVD [2, 32] and MSR-VTT2016 [33] are employed to evaluate the proposed methods. For the MSVD dataset [2, 32], there are 1997 simple video clips (e.g., cooking, exercise) and corresponding 80827 reference sentences in total. According to the using protocol, 1200 videos and related 48774 references are for training, and 100 videos and related 4290 references are for model validation and hyperparameter discovery. The rest video–sentence pairs including 677 videos and related sentences are for the model test. The dataset contains limited training samples and simple video–sentence pairs. Comparatively, MSR-VTT2016 [33]

possesses relatively larger-scale and more complex videos and corresponding descriptions. In the dataset, 10000 videos are included and each video corresponds to 20 description sentences. In general, 6513 videos and related references are for model training, while 497 video–sentence pairs are for validation, and the rest 2990 videos are for model evaluation.

Different from the task of classification or object detection, the evaluation of video description is more complicated since more factors require to be taken into account including accuracy, coherence and semantics. For more comprehensive evaluation of the generated sentences, the metrics for machine translation including BLEU [49], METEOR [50] and ROUGE_L [51] are usually lent to the video description. As for BLEU [49], it assesses the quality of generated sentences according to measuring the matching degree of n -grams (where $n \in \{1, 2, 3, 4\}$) in references and candidate sentences. Generally speaking, the larger the n and the higher the BLEU- n , the better the quality of generated sentences (for simplicity, the metric is denoted as “B- n ”). However, the BLEU metric focuses on the precision of word or phrase prediction but ignores the recall. By comparison, precision and recall are both in consideration in METEOR [50]. Specifically, three matching alignment strategies including precise matching, synonym matching and stem matching are employed to build the matching alignment set of references and candidate sentences. Then the ratio between the set size and candidate sentence length is treated as the precision, while the ratio between the set size and reference sentence length is treated as the recall. Finally, the harmonic mean of the precision and recall is calculated as the evaluation score. (The metric is marked as “M” for convenience.) Different from METEOR, a concept of longest common subsequence (LCS) is defined for ROUGE_L [51]. The ratio between the two lengths of LCS and candidate sentence is used as the precision, and the ratio between those of LCS and reference sentence is used as the recall (it is abbreviated as “R” for convenience). Besides, for more targeted evaluation of visual description, Vedantam et al. develop a novel evaluation metric of CIDEr [48]. The distribution of n -gram in all reference sentences of the image/video remained to be described is statistically analyzed by the idea of “consensus,” and each n -gram is assigned a different weight according to its frequency. Then the similarity of references and candidate sentences is calculated as the score. (The metric is abbreviated as “C” for convenience.)

In fact, an evaluation metric often focuses on one of the aspects of the generated sentences, and thus, it is difficult to assess the ability of a model comprehensively with a single metric. In this work, we follow the popular practice and employ all the metrics of BLUE [49], METEOR [50],

ROUGE_L [51] and CIDEr [48] to measure the quality of generated sentences.

4.2 Experimental setting

All our experiments are implemented on 4 GPUs (TITAN X Pascal, 12GB) to speed model convergence. And the popular deep learning framework Caffe [52] is employed to conduct model configuration, training and testing. In order to reduce redundancy and make full use of information in a video, a frame is sampled every 10. As for the visual features, two CNN models including GoogLeNet [6] and ResNet152 [7] are employed to extract feature from each sampled frame for verifying the performance of the proposed model comprehensively. The two CNN models are not only pre-trained on the large-scale classification dataset ImageNet [4], but also fine-tuned on an image captioning dataset MSCOCO2014 [53]. During fine-tuning, the LRCN framework [11] is employed as the backbone where the CNN model for visual encoding and LSTM for language decoding are jointly optimized. The solver of stochastic gradient descent is used, and the batch size is set to 16. As for the learning rate, it is initially set to 0.01 and then is gradually decreased by multiplying 0.1 in step size manner after each of 20000 iterations. The CNN model with 490000 iterations is picked to extract visual feature of each frame in videos since the model is in convergence and possesses the best performance on the validation set of MSCOCO2014.

The practice forces the CNN model to learn the category information of objects in the first pre-training stage. Then, the CNN model is further fine-tuned at the second stage, where it enables the model to perceive sufficient information in advance for visual description since there are enough image–sentence pairs in MSCOCO2014. Consequently, the two-stage pre-training strategy makes the model sensitive to frequently used words/phrases and sentence patterns in the following video description task. Then the extracted CNN feature sequence of sampled frames in a video is fed to the proposed model and comparable models in order for motion encoding and the following language decoding. During training, the number of time steps in m-S2VT is set to 80, which is learned from the practice in S2VT [14]. When the sum of sampled frames and words in the corresponding reference breaks the limitation, all the words should be fed to the model, while the extra frames will be truncated. Regarding the visual semantics enhancing branch, the number of time steps is equal to that in the part of m-S2VT for language decoding. However, the limitation will be removed at test time, which indicates that all the sampled frames will be fed to LSTM for visual motion encoding.

For further improving performance, the beam search is employed to expand the search space of words, which is also following the practice in most popular works. However, when the proposed models except for HE-VLSC[#]/A[#] are implemented on MSVD [2, 32], the pool of beam size is set to 1 empirically, which means the beam search is not working. As for all the remaining experiments, the pool of beam search is set to 3. In HE-VLSC^{*}/A^{*} model, the factors including α , β and γ for probability fusion are achieved by experimental experience. In general, when the ratio is assigned to 8:1:1 or 6:2:2, the best performance of the model can be obtained. (The discussion and possible reasons are provided in Sect. 4.3.2.)

4.3 Experimental result and discussion

4.3.1 Ablation study

In order to evaluate the effectiveness of the proposed methods including visual semantics enhancing, multi-objective optimization-based hybrid enhancing and complementary of visual and language semantics, ablation experiments are conducted on MSVD [2, 32] and MSR-VTT2016 [33]. Concretely, the Deep-Glove framework [37] is employed as the backbone to construct the baseline model. However, in the original Deep-Glove [37], the extra language semantics branch is pre-trained on other large-scale corpora for richer language semantics. And then the whole model is trained on video captioning dataset (e.g., MSVD [2, 32] and MST-VTT2016 [33]) with the extra language branch fine-tuning and the m-S2VT full training. We abandon this strategy but train the model on video captioning dataset directly for fair comparison. (The baseline model is denoted as E-LSC for convenience and distinction.) In addition, the proposed E-VSC/A, HE-VLSC/A and HE-VLSC^{*}/A^{*} are implemented on the two datasets.

The performances of the baseline and proposed models on MSVD dataset with GoogLeNet and ResNet152 features are shown in Table 1. When GoogLeNet feature is in use, it can be observed that both the proposed models E-VSC and E-VSA perform better than the baseline model (E-LSC) on MSVD from the results. More specifically, the performances on B-4 and CIDEr are 45.3 and 66.5, respectively, with E-VSC, outperforming the baseline model by 3.6 and 5.5. Note that the performance of E-VSA is inferior to E-VSC a little on all metrics, though the model is superior to the baseline model. This comparison trend reveals the effectiveness of the proposed visual semantics enhancing method. As for the proposed HE-VLSC and HE-VLSA, the HE-VLSC/A generally performs better on a few metrics compared to the baseline model and

E-VSC/A. In particular, the CIDEr of HE-VLSC reaches 67.2, with 0.7 points higher than the best E-VSC. However, the B-4, METEOR and ROUGE_L with HE-VLSC are not effectively improved but declined a little. It indicates that the B-LE branch has the ability to enhance the semantics of generated sentences but is not good at coherence improvement. On the contrary, the HE-VLSA concerns more with coherence which is reflected on BLEU, but yields to the entire semantics corresponding to CIDEr. The reason is that the GoogLeNet feature possesses weaker abstractness and representative ability, and more visual noises may be introduced into the model since the further mining of the visual semantics in the model with an extra branch. What is more, the visual noises are further amplified with element-wise addition operation for feature fusion, hindering the improvement of model performance.

As for HE-VLSC^{*}/A^{*}, it outperforms other comparable models on the whole with GoogLeNet feature. Take HE-VLSC^{*} as an example, it performs better than HE-VLSC on all evaluation metrics but CIDEr. Additionally, the HE-VLSA^{*} is also superior to HE-VLSA on each metric. In particular, the CIDEr of HE-VLSA^{*} outperforms HE-VLSA by 4.6. However, the comparison trend of the two models is opposite to that of HE-VLSA and HE-VLSC on CIDEr. It reveals that the element-wise addition has the ability to enhance the response of features to semantic concepts to a certain extent with multi-objective optimization.

Furthermore, when the ResNet152 feature is employed, the performance of E-VSC/A is inferior to the baseline model (E-LSC) in which the language enhancing branch is used. However, the performance of HE-VLSA in which the hybrid enhancing of visual and language semantics is implemented, the performance is improved greatly. Particularly, the B-4 and CIDEr outperform the baseline model by 2.4 and 1.4, respectively. Moreover, the performance of HE-VLSA^{*} on B-4 and CIDEr surpasses the baseline by 4.2 and 4.4, demonstrating the effectiveness and superiority of the proposed methods sufficiently. Another remarkable performance trend is that the model with element-wise addition (i.e., HE-VLSA and HE-VLSA^{*}) performs better than that with concatenation (i.e., HE-VLSC and HE-VLSC^{*}) on most metrics. The possible reason is that the ResNet152 feature is enough abstract compared to GoogLeNet, and the extra visual semantics enhancing branch cannot offer more effective information for the whole model.

The experimental results of the baseline model (E-LSC) and the proposed models on MSR-VTT2016 [33] with GoogLeNet and ResNet152 features are shown in Table 2. If the GoogLeNet feature is employed, the performance improvement of each proposed model on MSR-VTT2016 is not as obvious as that on MSVD dataset [2, 32]

Table 1 Performance comparison of each proposed model on MSVD dataset with GoogLeNet and ResNet152 feature, respectively (the “M,” “R” and “C” are the abbreviations of “METEOR,” “ROUGE_L” and “CIDEr,” respectively, and the E-LSC is actually the baseline model)

Feature	Model	B-1	B-2	B-3	B-4	M	R	C
GoogLeNet	E-LSC	76.9	63.6	52.6	41.7	30.8	67.0	61.0
	E-VSC	78.2	65.6	55.3	45.3	32.1	68.9	66.5
	E-VSA	77.5	64.5	53.7	43.1	31.7	68.1	64.2
	HE-VLSC	78.6	65.5	55.1	45.0	31.7	68.7	67.2
	HE-VLSA	77.9	64.9	54.2	43.7	31.6	68.3	62.5
	HE-VLSC*	80.6	67.9	57.1	46.2	32.5	69.6	66.4
	HE-VLSA*	79.1	66.3	55.7	45.0	32.5	69.2	67.1
ResNet152	E-LSC	79.9	67.5	56.7	45.8	33.5	70.0	74.8
	E-VSC	79.9	67.8	57.1	46.1	33.6	70.8	73.6
	E-VSA	80.5	68.4	57.8	46.8	33.3	70.5	73.0
	HE-VLSC	80.0	69.0	59.0	48.6	33.2	70.1	74.0
	HE-VLSA	81.1	69.3	58.9	48.2	33.8	71.1	76.2
	HE-VLSC*	82.3	69.7	59.3	48.9	33.9	71.3	76.5
	HE-VLSA*	82.4	70.9	61.0	50.9	34.5	72.1	79.2

Table 2 Performance comparison of each proposed model on MSR-VTT2016 dataset with GoogLeNet and ResNet152 feature, respectively

Feature	Model	B-1	B-2	B-3	B-4	M	R	C
GoogLeNet	E-LSC	76.5	61.6	48.2	36.8	26.3	58.2	39.6
	E-VSC	74.9	60.7	47.2	35.6	25.5	56.4	39.8
	E-VSA	75.0	60.4	46.2	34.1	25.2	55.9	37.9
	HE-VLSC	76.1	61.7	48.5	37.1	26.4	58.2	40.6
	HE-VLSA	77.1	62.8	49.7	38.1	26.8	59.1	41.6
	HE-VLSC*	76.4	62.1	48.6	36.7	26.1	57.8	40.6
	HE-VLSA*	77.6	63.2	49.7	37.7	26.6	58.8	40.8
ResNet152	E-LSC	78.2	63.8	50.2	38.5	27.4	59.5	44.6
	E-VSC	76.3	61.9	47.6	35.3	25.7	56.9	41.6
	E-VSA	76.9	62.7	48.6	36.3	25.7	56.9	41.7
	HE-VLSC	78.4	63.3	49.1	37.0	27.0	58.8	43.8
	HE-VLSA	78.9	64.7	51.1	39.1	27.2	59.8	45.1
	HE-VLSC*	78.8	64.4	50.5	38.5	26.9	59.2	44.6
	HE-VLSA*	79.1	64.9	51.0	38.6	27.2	59.8	44.3

compared to the baseline model. As an example in Table 2, the baseline model E-LSC commonly performs better than E-VSC/A on most of the metrics. This is rooted in that the sentences in MSR-VTT2016 are usually longer and with richer semantics compared to that in the MSVD dataset, and more language information can be captured by B-LE branch. As for HE-VLSC/A and HE-VLSC*/A*, they achieve better performance than baseline model and proposed E-VSC/A on the whole, but the best HE-VLSA model only outperforms the baseline model by 1.3 and 2.0 on B-4 and CIDEr, respectively. Also, the better HE-VLSA* compared to HE-VLSC* is superior to the baseline by 0.9 and 1.2. On the other hand, the performance of HE-VLSC*/A* is inferior to HE-VLSC/A generally,

which goes the opposite of the performance trend on MSVD. The possible reason is that the MSR-VTT2016 is relatively clean and there are fewer visual and language noises compared to MSVD, and the model is difficult to be disturbed with the errors from B-LE and B-VE, limiting the generalization ability of the model. At the test stage, the lower prediction accuracy of the two branches will weaken the fusion performance of the final word prediction.

And if the ResNet152 feature is employed, the performance trend is similar to that GoogLeNet feature is employed from the comparison between baseline model E-LSC and E-VSC/A. Another remarkable observation is that the HE-VLSC* performs higher than that of HE-VLSA* when GoogLeNet feature is the visual representation, while

the results are opposite if ResNet152 feature is in use. The trend indicates that the performance improvement of concatenation of the more abstract visual feature (*i.e.*, ResNet152) is limited compared to the operation of element-wise addition.

Besides the aforementioned model evaluation and discussion, the performances of HE-VLSC[#]/A[#] on MSVD [2, 32] and MSR-VTT2016 [33] with GoogLeNet feature and ResNet152 feature are presented in Table 3 and Table 4, respectively. For comparison, the E-VSC[#]/A[#] and E-LSC[#]/A[#] are evaluated, where the output probability from only visual semantics enhancing-based module or language enhancing-based module is used to word prediction, respectively, during the testing. From the results in Table 3, it is obvious that the performance of HE-VLSC[#]/A[#] is greatly improved compared to E-VSC[#]/A[#] or E-LSC[#]/A[#] on all evaluation metrics on MSVD. As an example, the B-4 and CIDEr of HE-VLSA[#] reach 47.8 and 69.4, respectively, on MSVD with GoogLeNet feature. And when ResNet152 feature is employed, the performance on the two metrics achieves to 52.4 and 81.5, respectively, outperforming that the E-LSC[#] or E-VSC[#] is used alone. Additionally, the proposed HE-VLSC[#]/A[#] also performs well on MSR-VTT2016 regardless of using GoogLeNet or ResNet152 feature, and the performance of the model is improved greatly in comparison with E-LSC[#]/A[#] and E-VSC[#]/A[#]. (The results are presented in Table 4.)

In addition, it can be found that the models with different fusion methods perform quite different from experimental results in Table 3 and Table 4. In general, the models with the fusion method of feature concatenation possess better performance than that with element-wise

addition. The reason can be attributed to that there is no loss of feature with concatenation operation, and the representation is relatively more complete and comprehensive. However, the element-wise addition improves the sparsity of features, retains the dimension and speeds up iteration, but certain available information may be obliterated.

4.3.2 Example of generated sentence and discussion

The weight assigning of α , β and γ belonging to m-S2VT, B-LE and B-VE, respectively, is investigated. The performance trends under different ratios of $\alpha : \beta : \gamma$ on MSVD dataset are presented in Fig. 8, where Fig. 8a, b shows the performance trends with GoogLeNet feature, while Fig. 8c, d shows those with ResNet152 feature. From the comparison, the models including HE-VLSC* and HE-VLSA* generally get the best performance on B-4 when the ratio of $\alpha : \beta : \gamma$ is 6:2:2 or 4:3:3 is employed. However, if the ratio of 4:3:3 is employed, the performance on CIDEr is not satisfactory when the GoogLeNet feature is used. The possible reason is that the modules of B-LE and B-VE can be treated as auxiliary branches for visual and language semantics enhancing and compensation since incomplete information is used as their input. On the contrary, the module m-S2VT receives the complete multimodal information and fuses the three modules outputs for word prediction, and it behaves better than the above two modules. And hence, the α is usually given bigger weights than β and γ . The ratio of $\alpha : \beta : \gamma$ is set to 8:1:1 or 6:2:2 empirically in our work.

In addition, a few examples of reference and generated sentences with the baseline and proposed models are presented in Fig. 9, which are from MSVD and MSR-VTT2016, respectively. From the examples, the generated

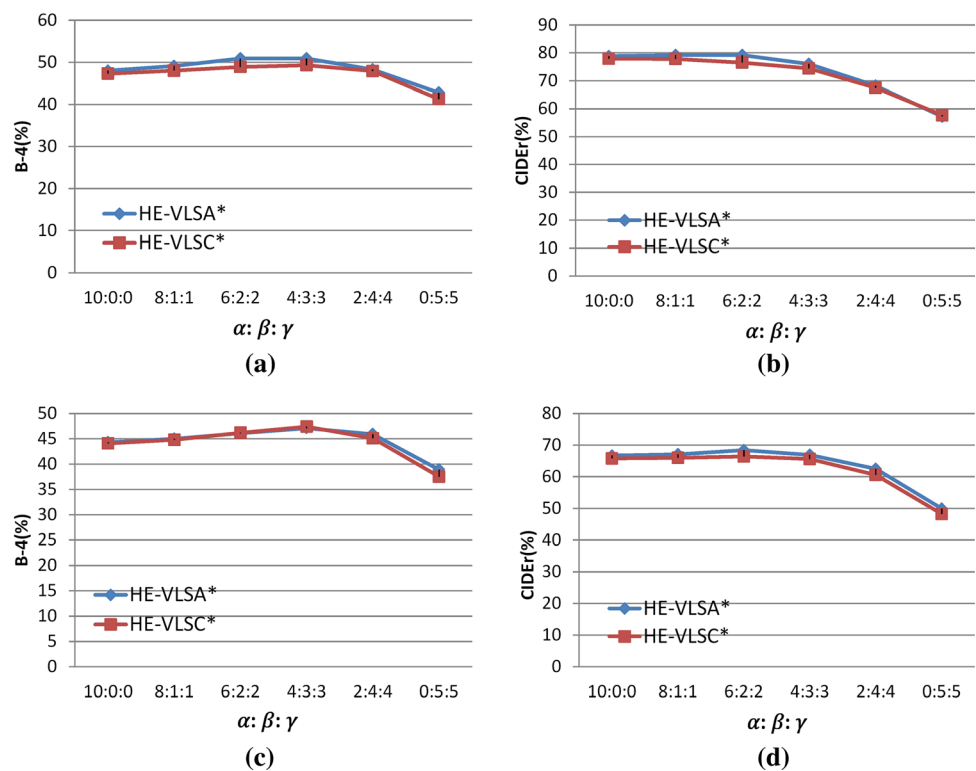
Table 3 Performance comparison of E-VSC[#]/A[#], E-LSC[#]/A[#] and HE-VLSC[#]/A[#] on MSVD dataset with GoogLeNet and ResNet152 features, respectively

Feature	Model	B-1	B-2	B-3	B-4	M	R	C
GoogLeNet	E-VSC [#]	79.1	67.0	56.5	46.2	31.2	67.7	65.9
	E-LSC [#]	79.0	66.0	55.2	44.4	31.8	68.5	64.2
	E-VSA [#]	79.1	67.0	56.3	45.7	31.5	67.8	66.8
	E-LSA [#]	79.1	66.4	55.9	45.2	31.6	68.0	65.9
	HE-VLSC [#]	79.8	67.9	57.7	47.5	32.1	69.3	67.9
	HE-VLSA [#]	80.5	68.7	58.4	47.8	32.7	69.3	69.4
ResNet152	E-VSC [#]	81.0	69.8	59.4	48.4	32.0	68.5	73.9
	E-LSC [#]	81.6	70.3	60.3	49.9	33.7	70.9	77.3
	E-VSA [#]	81.9	71.2	61.0	50.7	33.7	70.5	78.8
	E-LSA [#]	81.5	70.2	60.1	49.0	33.9	71.1	77.4
	HE-VLSC [#]	83.0	72.4	62.8	52.4	34.4	71.6	81.5
	HE-VLSA [#]	82.3	71.6	61.9	51.6	34.1	71.6	79.4

Table 4 Performance comparison of E-VSC[#]/A[#], E-LSC[#]/A[#] and HE-VLSC[#]/A[#] on MSR-VTT2016 dataset with GoogLeNet and ResNet152 features, respectively

Feature	Model	B-1	B-2	B-3	B-4	M	R	C
GoogLeNet	E-VSC [#]	77.1	62.5	48.7	36.6	26.2	58.5	40.6
	E-LSC [#]	77.2	62.5	48.9	36.9	26.4	58.8	40.5
	E-VSA [#]	76.7	61.9	48.3	36.5	26.0	58.2	40.1
	E-LSA [#]	76.9	62.4	48.9	37.1	26.6	58.7	41.0
	HE-VLSC [#]	78.4	64.1	50.7	38.7	26.9	59.6	42.4
	HE-VLSA [#]	77.7	63.3	49.8	38.0	26.8	59.2	42.1
ResNet152	E-VSC [#]	78.1	63.7	49.6	37.4	26.5	58.8	43.9
	E-LSC [#]	79.4	65.3	51.3	38.9	27.3	59.8	45.4
	E-VSA [#]	78.5	64.1	50.0	37.6	26.7	58.9	43.9
	E-LSA [#]	79.3	65.1	51.5	39.2	27.3	60.1	44.7
	HE-VLSC [#]	79.6	65.8	51.8	39.4	27.4	60.2	46.1
	HE-VLSA [#]	80.0	66.4	52.8	40.4	27.6	60.7	46.0

Fig. 8 Performance trend comparison on B-4 and CIDEr under different ratios of $\alpha : \beta : \gamma$ on MSVD dataset (**a** and **b** are with GoogLeNet feature, while **c** and **d** are with ResNet152 feature)

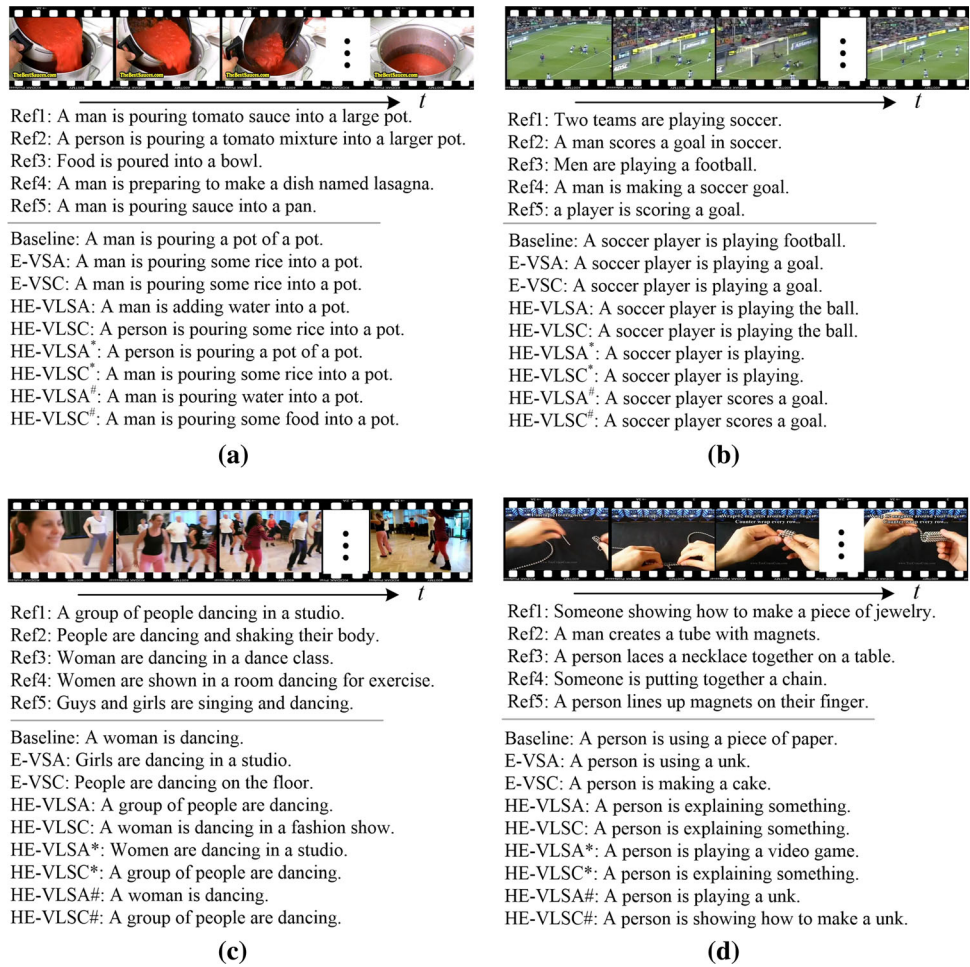


sentences with the proposed models possess better quality than the baseline model and E-VSC/A on the whole. Particularly, the sentences generated with HE-VLSC[#] are superior to those with other comparable models in accuracy and semantics. As an example in Fig. 9a which is from MSVD, the visual content is described comprehensively and accurately in the sentence generated with HE-VLSC[#]. In contrast, both the grammar and logic are not correct in the sentence with the baseline model. As for other comparable models, the wrong words are usually predicted

(e.g., “rice,” “water”). Similarly, the used sentence elements of “A group of people” in the generated sentence with HE-VLSC[#] is more accurate than “A woman” in that with baseline model according to the video to be described in Fig. 9c which is from MSR-VTT2016.

From Fig. 9, an interesting fact can be observed that different models may be sensitive to different videos. Take Fig. 9b for example, the baseline model flatly describes the video, while the generated sentences with HE-VLSC/A and HE-VLSC[#]/A[#] are generally coarser. However, the

Fig. 9 Examples of reference, generated sentence with different models (the ResNet152 is employed in all models, and the samples of **a** and **b** are from MSVD dataset, while the **c** and **d** are from MSR-VTT2016 dataset. The “Ref” is the selected references of the corresponding videos)



sentences with HE-VLSC/A and HE-VLSA* behave more appropriately in Fig. 9d, by contrary, the description is ambiguous in sentences with HE-VLSA# (“unk” is used, which stands for “unknown”). Most important of all, there is still a great gap to be bridged for generated sentences compared to references in flexibility and semantics.

The average time cost (GPU time) comparison of the baseline model and the proposed models for one sentence

generation is given in Table 5. From the results, it is obvious that the E-VSC and E-VSA cost less time than the baseline model (E-LSC) whether the GoogLeNet feature or ResNet152 feature is employed. And from comparison between the time cost of HE-VLSC and HE-VLSA, the HE-VLSC takes more time than HE-VLSA since the element-wise addition operation does not extend the dimension of the feature for word prediction. However, when

Table 5 Average time cost (ms) (GPU time) of each model for one sentence generation with GoogLeNet feature and ResNet152 feature

Model	Time cost (GoogLeNet)	Time cost (ResNet152)
Baseline (E-LSC)	48.82	71.99
E-VSC	48.24	56.40
E-VSA	44.10	53.52
HE-VLSC	81.51	63.65
HE-VLSA	46.49	50.82
HE-VLSC*	140.05	146.51
HE-VLSA*	125.34	120.58
HE-VLSC#	150.00	112.64
HE-VLSA#	107.01	118.79

HE-VLSC*/A* is implemented, the time cost increases significantly. The reason is that each branch (module) such as B-LE, B-VE and m-S2VT in HE-VLSC*/A* has to calculate the probability vector which will be fused for final word prediction. Additionally, similar to the trend of HE-VLSC/A, the HE-VLSA* possesses a lower time cost than HE-VLSA*, which also indicates the concatenation operation consumes more time than element-wise addition operation.

As for the parameter scale of each module, there are 21.81 million and 0.51 million parameters for word embedding and visual feature dimension reduction, respectively (suppose the GoogLeNet feature is employed). In m-S2VT, about 12.60 million parameters are contained. And for B-LE and B-VE, the parameter scales are about 6.30 million and 8.40 million, respectively. Regarding the output module, it possesses a relatively large scale generally, with about 74.40 million parameters for one branch. Therefore, the parameter scales of HE-VLSA and HE-VLSC are about 124.02 million and 198.42 million, respectively, while when the multi-objective strategy is employed, the scales increase sharply to 272.82 million and 347.22 million in the two models (HE-VLSA* and HE-VLSC*). Additionally, the HE-VLSC# has the largest scale, with about 359.82 million parameters. By contrast, the scale is only about 211.02 million in HE-VLSA#. The comparison details are presented in Table 6.

4.3.3 Comparison with the state-of-the-art methods

Performance comparison to the popular models is made, and the results are shown in Tables 7 and 8. It is obvious that the proposed HE-VLSA*(R), HE-VLSA#(R) and HE-VLSC#(R) (where the “R” represents that the ResNet152 feature is used in the model) achieve competitive performance on MSVD compared to the state of the art. Especially in the HE-VLSC#(R) model, it outperforms

Table 6 Parameter scale comparison of each model

Model	Parameter scale ($\times 10^6$)
Baseline (E-LSC)	189.51
E-VSC	170.31
E-VSA	95.91
HE-VLSC	198.42
HE-VLSA	124.02
HE-VLSC*	347.22
HE-VLSA*	272.82
HE-VLSC#	359.82
HE-VLSA#	211.02

Table 7 Performance comparison to the other popular models on MSVD dataset

Model	B-1	B-2	B-3	B-4	M	C
FGM [55]	–	–	–	13.7	23.9	–
LSTM-YT [36]	–	–	–	33.3	29.1	–
S2VT [14]	–	–	–	–	29.8	–
ReBiLSTM [56]	79.0	60.5	48.4	37.3	30.3	–
HRNE [17]	79.2	66.3	55.1	43.8	33.1	–
SA [16]	80.0	64.7	52.6	41.9	29.6	51.7
LSTM-E [15]	78.8	66.0	55.4	45.3	31.0	–
LSTM-GAN [31]	–	–	–	42.9	30.4	–
GRU-RCN [57]	–	–	–	43.3	31.6	67.8
h-RNN [58]	81.5	70.4	60.4	49.9	32.6	65.8
ASTAR [59]	–	–	–	51.7	36.4	72.2
hLSTMat [60]	82.9	72.2	63.0	53.0	33.6	73.8
aLSTMs [61]	81.8	70.8	61.1	50.8	33.3	74.8
M3-IC [62]	82.5	72.4	62.8	52.8	33.3	–
MS-RNN [63]	82.9	72.6	63.5	53.3	33.8	74.8
SCN-LSTM [64]	–	–	–	51.1	33.5	77.7
GRU-EVE _{hft+sem} [54]	–	–	–	47.9	35.0	78.1
TDCovED [65]	–	–	–	53.3	33.8	76.4
HE-VLSA*(R)	82.4	70.9	61.0	50.9	34.5	79.2
HE-VLSA#(R)	82.3	71.6	61.9	51.6	34.1	79.4
HE-VLSC#(R)	83.0	72.4	62.8	52.4	34.4	81.5

Table 8 Performance comparison to the other popular models on MSR-VTT2016 dataset

Model	B-4	M	R	C
ReBiLSTM [56]	33.9	26.2	–	–
LSTM-GAN [31]	36.0	26.1	–	–
aLSTMs [61]	38.0	26.1	–	–
M3-IC [62]	38.1	26.6	–	–
MS-RNN(R) [63]	39.8	26.1	59.3	40.9
RecNet _{local} [66]	39.1	26.6	59.3	42.7
ruc-uva [67]	38.7	26.9	58.7	45.9
VideoLAB [61]	39.1	27.7	60.6	44.1
Aalto [68]	39.8	26.9	59.8	45.7
v2t navigator [69]	40.8	28.2	60.9	44.8
PickNet [29]	41.3	27.7	–	44.1
TDCovED [65]	39.5	27.5	–	42.8
GRU-EVE _{hft+sem} [54]	38.3	28.4	–	48.1
HE-VLSA(R)	39.1	27.2	59.8	45.1
HE-VLSA#(R)	40.4	27.6	60.7	46.0
HE-VLSC#(R)	39.4	27.4	60.2	46.1

the popular GRU-EVE_{hft+sem} [54] by 3.4 on the CIDEr metric. On the MSR-VTT2016 dataset [33], the proposed HE-VLSA(R), HE-VLSA[#](R) and HE-VLSC[#](R) also possess better performance than most of the popular works. However, the proposed model is inferior to GRU-EVE_{hft+sem} [54] on CIDEr, though they perform better than GRU-EVE_{hft+sem} [54] on B-4 metric, which indicates that the proposed methods remains to be further improved on a relatively large and complex dataset.

5 Conclusion

A framework with hybrid visual and language semantics enhancing and complementary is proposed in this work, which aims to mine and use visual information efficiently, and compensate each other of vision and language information. Inspired by Deep-Glove model, a visual semantics enhancing branch is developed and appended on a multi-modal-based module with concatenation or element-wise addition operation to further take advantage of visual information. On the basis, the language and visual semantics enhancing branches, and multimodal-based module are integrated into a wider and more effective model. Additionally, the multi-objective optimization strategy is employed and developed a new framework to further optimize the proposed model and boost performance. Experiments on MSVD and MSR-VTT2016 datasets are conducted, and competitive results are achieved. From the comparison, the proposed models are more effective and superior compared to not only the baseline model but also the other popular methods. Specifically, if the more abstract visual feature (*e.g.*, ResNet152) is employed, the visual semantics enhancing method can effectively supplement visual information to the model and better performance can be obtained compared to language semantics enhancing. Alternatively, the model with a language semantics enhancing branch performs better than that with visual semantics enhancing branch. And when both visual and language semantics enhancing methods are employed, the performances are further improved generally. Particularly, the variant HE-VLSC[#] possesses the competitive performance compared to the state-of-the-art methods on both datasets.

The results indicate that the language semantics enhancing is better for the situation that the reference sentences are clean and with rich semantics, while the visual semantics enhancing is more adaptable when there are more extra noises in sentences. Naturally, the integration of the two methods facilitates to complementary of visual and language information and further performance improvement. Furthermore, the concatenation operation for

the feature is more sensitive to complete information in each branch and performance balance of different modules in that the HE-VLSC[#] performs more excellent than other models. However, the accuracy and semantics of the generated sentences remain to be further improved. Actually, the proposed methods can be introduced into other popular and powerful frameworks to further boost performance. In the future work, more prior knowledge such as visual concept and attribute, the attention unit and related variant (*e.g.*, Transformer), and reinforcement learning strategy will be introduced into the proposed framework for further performance improvement.

Acknowledgements This work was supported in part by National Natural Science Foundation of China (No. 62062041, 61961023), Jiangxi Provincial Natural Science Foundation (No. 20212BAB202020), Ph.D. Research Initiation Project of Jingtangshan University (No. JZB1923, JZB1807), Bidding Project for the Foundation of College's Key Research on Humanities and Social Science of Jiangxi Province (No. JD17082).

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Kojima A, Tamura T, Fukunaga K (2002) Natural language description of human activities from video images based on concept hierarchy of actions. *Int J Comput Vis* 50(2):171–184
2. Guadarrama S, Krishnamoorthy N, Malkarnenkar G, Venugopalan S, Mooney R, Darrell T, Kate S (2013) Youtube2text: recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In: *IEEE international conference on computer vision (ICCV)*, Sydney, Australia, Jan 2013, IEEE, pp 2712–2719
3. Gupta A, Srinivasan P, Shi J, Davis LS (2009) Understanding videos, constructing plots learning a visually grounded storyline model from annotated videos. In: *IEEE conference on computer vision and pattern recognition (CVPR)*, Miami, USA, Jun 2009, IEEE, pp 2012–2019
4. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: *Annual conference on neural information processing systems (NIPS)*, Quebec, Canada, Jul 2012, MIT, pp 1097–1105
5. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. In: *International conference on learning representations (ICLR)*, Banff, Canada
6. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: *IEEE conference on computer vision and pattern recognition (CVPR)*, Boston, USA, June 2015, IEEE, pp 1–9
7. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: *IEEE conference on computer vision and pattern recognition (CVPR)*, Las Vegas, USA, June 2015, IEEE, pp 770–778
8. Simonyan K, Zisserman A (2014) Two-stream convolutional networks for action recognition in videos. In: *Annual conference*

- on neural information processing systems (NIPS), Montreal, Canada, Dec 2014, MIT, pp 568–576
9. Wang L, Qiao Y, Tang X (2015) Action recognition with trajectory-pooled deep-convolutional descriptors. In: IEEE conference on computer vision and pattern recognition (CVPR), Boston, USA, June 2015, IEEE, pp 4305–4314
 10. Mao J, Xu W, Yang Y, Wang J, Yuille A (2014) Deep captioning with multimodal recurrent neural networks (m-RNN). In: International conference on learning representations (ICLR), Banff, Canada, Apr 2014
 11. Donahue J, Hendricks LA, Guadarrama S, Rohrbach M, Venugopalan S, Saenko K, Darrell T (2015) Long-term recurrent convolutional networks for visual recognition and description. In: IEEE conference on computer vision and pattern recognition (CVPR), Boston, USA, June 2015, IEEE, pp 2625–2634
 12. Vinyals O, Toshev A, Bengio S, Erhan D (2015) Show and tell: A neural image caption generator. In: IEEE conference on computer vision and pattern recognition (CVPR), Boston, USA, June 2015, IEEE, pp 3156–3164
 13. Tang P, Wang H, Kwong S (2018) Deep sequential fusion lstm network for image description. *Neurocomputing* 312:154–164
 14. Venugopalan S, Rohrbach M, Donahue J, Mooney R, Darrell T, Saenko K (2015) Sequence to sequence—video to text. In: IEEE international conference on computer vision (ICCV), Santiago, Chile, Dec 2015, IEEE, pp 4534–4542
 15. Pan Y, Mei T, Yao T, Li H, Rui Y (2016) Jointly modeling embedding and translation to bridge video and language. In: IEEE conference on computer vision and pattern recognition (CVPR), Las Vegas, USA, June 2016, IEEE, pp 4594–4602
 16. Yao L, Torabi A, Cho K, Ballas N, Pal C, Larochelle H, Courville A (2015) Describing videos by exploiting temporal structure. In: IEEE international conference on computer vision (ICCV), Santiago, Chile, Dec 2015, IEEE, pp 4507–4515
 17. Pan P, Xu Z, Yang Y, Wu F, Zhuang Y (2016) Hierarchical recurrent neural encoder for video representation with application to captioning. In: IEEE conference on computer vision and pattern recognition (CVPR), Las Vegas, USA, June 2016, IEEE, pp 1029–1038
 18. Yu Y, Ko H, Choi J, Kim G (2017) End-to-end concept word detection for video captioning, retrieval, and question answering. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, USA, Jul., IEEE, pp 3261–3269
 19. Zheng Q, Wang C, Tao D (2020) Syntax-aware action targeting for video captioning. In: In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Online (Virtual), June 2020, IEEE, pp 13096–13105
 20. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z et al (2015) Imagenet large scale visual recognition challenge. *Int J Comput Vis* 115(3):211–252
 21. Xu K, Ba JL, Kiros R, Cho K, Courville A, Salakhutdinov R, et al (2015) Show, attend and tell: Neural image caption generation with visual attention. In: International conference on machine learning (ICML), Lille, France, July 2015, ACM, pp 2048–2057
 22. Cho K, Courville A, Bengio Y (2015) Describing multimedia content using attention-based encoder-decoder networks. *IEEE Trans Multimedia* 17(11):1875–1886
 23. Bin Y, Yang Y, Shen F, Xie N, Shen H, Li X (2019) Describing video with attention based bidirectional lstm. *IEEE Trans Cyber* 49(7):2631–2641
 24. Gao L, Guo Z, Zhang H, Xu X, Shen H (2017) Video captioning with attention-based lstm and semantic consistency. *IEEE Trans Multimedia* 19(9):2045–2055
 25. Jin T, Huang S, Chen M, Li Y, Zhang Z (2020) Sbat: Video captioning with sparse boundary-aware transformer. In: International joint conference on artificial intelligence (IJCAI), Online (Virtual), Jan 2020, Morgan Kaufmann, pp 630–636
 26. Pan Y, Li Y, Luo J, Xu J, Yao T, Mei T (2020) Auto-captions on gif: a large-scale video-sentence dataset for vision-language pre-training. [arXiv:2007.02375](https://arxiv.org/abs/2007.02375)
 27. Wu Q, Shen C, Liu L, Dick A, Hengel A (2016) What value do explicit high level concepts have in vision to language problems? In: IEEE conference on computer vision and pattern recognition (CVPR), Las Vegas, USA, Jun 2016, IEEE, pp 203–212
 28. Pan Y, Yao T, Li H, Mei T (2017) Video captioning with transferred semantic attributes. In: IEEE international conference on computer vision (ICCV), Venice, Italy, Oct 2017, IEEE, pp 984–992
 29. Chen Y, Wang S, Zhang W, Huang Q (2018) Less is more: picking informative frames for video captioning. In: European conference on computer vision (ECCV), Munich, Germany, Sept 2018, IEEE, pp 367–384
 30. Wang X, Chen W, Wu J, Wang YF, Wang WY (2018) Video captioning via hierarchical reinforcement learning. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, USA, Jun 2018, IEEE, pp 4213–4222
 31. Yang Y, Zhou J, Ai J, Bin Y, Hanjalic A, Shen H (2018) Video captioning by adversarial lstm. *IEEE Trans Image Process* 27(11):5600–5611
 32. Chen DL, Dolan WB (2011) Collecting highly parallel data for paraphrase evaluation. In: The 49th annual meeting of the association for computational linguistics (ACL). Portland, USA, Jun 2011, ACL, pp. 190–200
 33. Xu J, Mei T, Yao T, Rui Y (2016) MSR-VTT: A large video description dataset for bridging video and language. In: IEEE conference on computer vision and pattern recognition (CVPR), Las Vegas, USA, Jun 2016, IEEE, pp 5288–5296
 34. Nagel HH (1994) A vision of xxx vision and language xxx comprises action: an example from road traffic. *Artif Intell Rev* 8(2):189–214
 35. Rohrbach M, Qiu W, Titov I, Thater S, Pinkal M, Schiele B (2013) Translating video content to natural language descriptions. In: IEEE international conference on computer vision (ICCV), Sydney, Australia, Jan 2013, IEEE, , pp 433–440
 36. Venugopalan S, Xu H, Donahue J, Rohrbach M, Mooney R, Saenko K (2015) Translating videos to natural language using deep recurrent neural networks. In: The 2015 annual conference of the north american chapter of the ACL (NAACL), Denver, USA, May, ACL, pp 1494–1504
 37. Venugopalan S, Hendricks LA, Mooney R, Saenko K (2016) Improving lstm-based video description with linguistic knowledge mined from text. In: Conference on empirical methods in natural language processing (EMNLP), Austin, USA, Nov 2016, ACL, pp 1961–1966
 38. Johnson J, Karpathy A, Li FF (2016) DenseCap: Fully convolutional localization networks for dense captioning. In: IEEE conference on computer vision and pattern recognition (CVPR), Las Vegas, USA, Jun 2016, IEEE, pp 4565–4574
 39. Shen Z, Li J, Su Z, Li M, Chen Y, Jiang YG, Xue X (2017) Weakly supervised dense video captioning. In: IEEE conference on computer vision and pattern recognition (CVPR), Honolulu, USA, Jul 2017, IEEE, pp 1916–1924
 40. Wang J, Jiang W, Ma L, Liu W, Xu Y (2018) Bidirectional attentive fusion with context gating for dense video captioning. In: IEEE conference on computer vision and pattern recognition (CVPR), Salt Lake City, USA, Jun 2018, IEEE, pp 7190–7198
 41. Zhou L, Zhou Y, Jason JC, Richard S, Xiong C (2018) End-to-end dense video captioning with masked transformer. In: IEEE conference on computer vision and pattern recognition (CVPR), Salt Lake City, USA, Jun 2018, IEEE, pp 8739–8748
 42. Krishna R, Hata K, Ren F, Li FF, Niebles JC (2017) Dense-captioning events in videos. In: IEEE international conference on

- computer vision (ICCV), Venice, Italy, Oct 2017, IEEE, pp 706–715
43. Mun J, Yang L, Ren Z, Xu N, Han B (2019) Streamlined dense video captioning. In: IEEE conference on computer vision and pattern recognition (CVPR), Long Beach, USA, Jun 2019, IEEE, pp 6588–6597
 44. Zhang Z, Shi Y, Yuan C, Li B, Wang P, Hu W, Zha Z (2020) Object relational graph with teacher-recommended learning for video captioning. In: IEEE conference on computer vision and pattern recognition (CVPR), Online (Virtual), Jun 2020, IEEE, pp 13278–13288
 45. Zhang J, Peng Y (2019) Object-aware aggregation with bidirectional temporal graph for video captioning. In: IEEE conference on computer vision and pattern recognition (CVPR), Long Beach, USA, Jun 2019, IEEE, pp 8327–8336
 46. Zhang J, Peng Y (2019) Hierarchical vision-language alignment for video captioning. In: International conference on multimedia modeling (MMM), Thessaloniki, Greece, Jan 2019, Springer, pp 42–54
 47. Park JS, Rohrbach M, Darrell T, Rohrbach A (2019) Adversarial inference for multi-sentence video description. In: IEEE conference on computer vision and pattern recognition long beach, USA, Jun 2019, IEEE, pp 6598–6608
 48. Vedantam R, Zitnick CL, Parikh D (2015) CIDEr: Consensus-based image description evaluation. In: IEEE conference on computer vision and pattern recognition (CVPR), Boston, USA, Jun 2015, IEEE, pp 4566–4575
 49. Papineni K, Roukos S, Ward T, Zhu WJ (2002) BLEU: A method for automatic evaluation of machine translation. In: Annual meeting of the association for computational linguistics (ACL), Philadelphia, USA, Jul 2002, ACL, pp 311–318
 50. Banerjee S, Lavie A (2005) METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In: Annual meeting of the association for computational linguistics workshop (ACLW), Ann Arbor, USA, Jun 2005, ACL, pp 65–72
 51. Lin CY, Och FJ (2004) Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In: Annual meeting of the association for computational linguistics (ACL), Barcelona, Spain, Jul 2004, ACL, pp 21–26
 52. Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R, Guadarrama S, Darrell T (2014) Caffe: Convolutional architecture for fast feature embedding. In: ACM international conference on multimedia (ACM MM), Orlando, USA, Nov 2014, ACM, pp 675–678
 53. Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollr P, Zitnick CL (2014) Microsoft coco: Common objects in context. In: European conference on computer vision (ECCV), Zurich, Switzerland, Sept 2014, Springer, pp 740–755
 54. Aafaq N, Akhtar N, Liu W, Gilani SZ, Mian A (2019) Spatio-temporal dynamics and semantic attribute enriched visual encoding for video captioning. In: IEEE conference on computer vision and pattern recognition (CVPR), Long Beach, USA, Jun 2019, IEEE, pp 12487–12496
 55. Thomason J, Venugopalan S, Guadarrama S, Saenko K, Mooney R (2014) Integrating language and vision to generate natural language descriptions of videos in the wild. In: Proceedings of international conference on computational linguistics (COLING), Dublin, Ireland, Aug 2014, ACL, pp 1218–1227
 56. Baraldi L, Costantino G, Rita C (2017) Hierarchical boundary-aware neural encoder for video captioning. In: IEEE conference on computer vision and pattern recognition (CVPR), Honolulu, USA, Jul 2017, IEEE, pp 3185–3194
 57. Ballas N, Yao L, Pal C, Courville A (2015) Delving deeper into convolutional networks for learning video representations. In: International conference on learning representations (ICLR), San Diego, USA, May 2015
 58. Yu H, Wang J, Huang Z, Yang Y, Xu W (2016) Video paragraph captioning using hierarchical recurrent neural networks. In: IEEE conference on computer vision and pattern recognition (CVPR), Las Vegas, USA, Jun 2016, IEEE, pp 4584–4593
 59. Pu Y, Min MR, Gan Z, Carin L (2016) Adaptive feature abstraction for translating video to language. arXiv preprint [arXiv:1611.07837](https://arxiv.org/abs/1611.07837)
 60. Gao L, Li X, Song J, Shen HT (2020) Hierarchical LSTMs with adaptive attention for visual captioning. *IEEE Trans Pattern Anal Mach Intell* 42(5):1112–1131
 61. Ramanishka V, Abir D, Huk PD, Subhashini V, Anne HL, Marcus R, Kate S (2016) Multimodal video description. In: ACM conference on multimedia conference (ACM MM), Amsterdam, Netherlands, Oct 2016 ACM, pp 1092–1096
 62. Wang J, Wang W, Huang Y, Wang L, Tan T (2018) M3: Multimodal memory modelling for video captioning. In: IEEE conference on computer vision and pattern recognition (CVPR), Salt Lake City, USA, Jun 2018, IEEE, pp 7512–7520
 63. Song J, Guo Y, Gao L, Li X, Alan H, Shen H (2019) From deterministic to generative: multimodal stochastic rnns for video captioning. *IEEE Trans Neural Netw Learn Syst* 30(10):3047–3058
 64. Gan Z, Gan C, He X, Pu Y, Tran K, Gao J, Carin L, Deng L (2017) Semantic compositional networks for visual captioning. In: IEEE conference on computer vision and pattern recognition (CVPR), Honolulu, USA, Jul 2017, IEEE, pp 5630–5639
 65. Chen J, Pan Y, Li Y, Yao T, Chao H, Mei T (2019) Temporal deformable convolutional encoder-decoder networks for video captioning. In: AAAI conference on artificial intelligence (AAAI), Honolulu, USA, Jan 2019, AAAI, pp 8167–8174
 66. Wang B, Ma L, Zhang W, Liu W, Reconstruction network for video captioning. In: IEEE conference on computer vision and pattern recognition (CVPR), Salt Lake City, USA, Jun 2018, IEEE, pp 7622–7631
 67. Dong J, Li X, Lan W, Huo Y, Snoek CG (2016) Early embedding and late reranking for video captioning. In: ACM conference on multimedia conference (ACM MM), Amsterdam, Netherlands, Oct 2016, ACM, pp 1082–1086
 68. Shetty R, Laaksonen J (2016) Frame- and segment-level features and candidate pool evaluation for video caption generation. In: ACM conference on multimedia conference (ACM MM), Amsterdam, Netherlands, Oct 2016, ACM, pp 1073–1076
 69. Jin Q, Chen J, Chen S, Xiong Y, Hauptmann A (2016) Describing videos using multimodal fusion. In: ACM conference on multimedia conference (ACM MM), Amsterdam, Netherlands, Oct 2016, ACM, pp 1087–1091

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.