Hindawi
Mathematical Problems in Engineering
Volume 2018, Article ID 3125879, 8 pages
https://doi.org/10.1155/2018/3125879

Research Article

Multimodal Feature Learning for Video Captioning

Sujin Lee and Incheol Kim

Department of Computer Science, Kyonggi University, San 94-6, Yiui-dong, Youngtong-gu, Suwon-si 443-760, Republic of Korea

Correspondence should be addressed to Incheol Kim; kic@kgu.ac.kr

Received 6 October 2017; Revised 16 January 2018; Accepted 24 January 2018; Published 19 February 2018

Academic Editor: Daniel Zaldivar

Video captioning refers to the task of generating a natural language sentence that explains the content of the input video clips. This study proposes a deep neural network model for effective video captioning. Apart from visual features, the proposed model learns additionally semantic features that describe the video content effectively. In our model, visual features of the input video are extracted using convolutional neural networks such as C3D and ResNet, while semantic features are obtained using recurrent neural networks such as LSTM. In addition, our model includes an attention-based caption generation network to generate the correct natural language captions based on the multimodal video feature sequences. Various experiments, conducted with the two large benchmark datasets, Microsoft Video Description (MSVD) and Microsoft Research Video-to-Text (MSR-VTT), demonstrate the performance of the proposed model.

1. Introduction

As video data increases, there has been a recent surge of interest in automatic video content analysis. Furthermore, technological advancement in computer vision, natural language processing, and machine learning has resulted in an increase of interest in complex intelligence problems relating to the simultaneous understanding of natural language and video clips. Video-based complex intelligence problems typically include video captioning and video question answering. As illustrated by the example shown in Figure 1, video captioning refers to the task of generating a natural language sentence that explains the content of the input video clip.

Video captioning process generally comprises feature extraction from input video clips and caption generation based on the extracted features. In many related works, video captioning was addressed using an encoder-decoder framework [1–3]. In these frameworks, features are first extracted by the encoder, followed by caption generation using the decoder. A convolutional neural network (CNN) like ResNet [4], VGG [5], and C3D [6] is selected as an encoder for such frameworks, whereas a recurrent neural network (RNN) like LSTM [7] is chosen as a decoder. However, they considered frame features of the video equally, without any particular focus. Some subsequent works have attempted to make use of an attention-based mechanism to learn where to focus in the image/video during captioning [8–10]. On the other hand, they still ignore the gap between low-level video feature and sentence descriptions, without clearly representing high-level video concepts. In order to address the above-mentioned problems, recent works add explicit high-level semantic concepts of the input image/video [11–13]. Although significant performance improvements were achieved, integration of semantic concepts into the LSTM-based caption generation process is still constrained in these ways: semantic features are used only (1) for initialization of the first step of the LSTM or (2) for implementing a soft attention mechanism to the LSTM-based caption generation process.

This study proposes a deep neural network model, SeFLA (SEmantic Feature Learning and Attention-Based Caption Generation), for effective video captioning by utilizing both visual and semantic features that describe the video content. In the proposed model, visual features are extracted using ResNet CNN, while semantic features are obtained using LSTM RNN. Moreover, the proposed model adopts an attention-based mechanism that determines which semantic feature to focus on at every time step to generate correct captions effectively based on the multimodal video features. To assess the performance of the suggested model, various

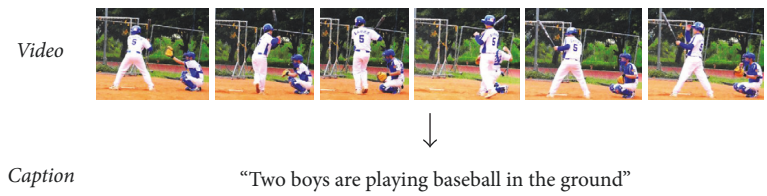


Figure 1: Example of video captioning.

Figure 2: Examples of dynamic and static semantic features.

experiments are run using the Microsoft Video Description (MSVD) [14] and Microsoft Research Video-to-Text (MSR-VTT) [15] datasets, following which the results are discussed.

2. Related Work

Previously, visual content understanding and natural language processing were not correlated with each other. Integrating visual content with natural language learning to generate descriptions for images/videos has been regarded as a challenging task [16, 17]. Video captioning is a critical step towards machine intelligence and many applications such as video retrieval, video understanding, blind navigation, and automatic video subtitling. Inspired by the successful use of the encoder-decoder framework employed in machine translation, many existing works on video captioning employ a convolutional neural network (CNN) as an encoder, obtaining a fixed-length vector representation of a given video. On the other hand, they adopt a recurrent neural network (RNN), typically implemented with long short-term memory (LSTM) [7] as a decoder to generate a natural language caption [1–3]. However, although there is salient part of the video that contribute more to captioning, they considered frame features of the video equally, without any particular focus.

Some recent works attempted to make use of an attention-based mechanism to learn where to focus in the image/video during caption generation [8–10]. Attention mechanism is a standard part of the deep learning toolkit, contributing to impressive results in neural machine translation, visual captioning, and question answering. Attention mechanism applicable to a video clip can be categorized into temporal attention, which indicates the frames to focus on in a video frame sequence and spatial attention, which specifies the key regions in a frame. In a recent work, an adjusted temporal attention mechanism is employed to avoid focusing on non-visual words (e.g., "the" and "a") during caption generation [10]. Although the attention-based approaches mentioned above have achieved excellent results, they still ignore the gap between low-level video feature and sentence descriptions, without clearly representing high-level video concepts.

Furthermore, recent works show that adding explicit high-level semantic concepts of the input image/video can further improve visual captioning [11–13]. In these works, detecting explicit semantic concepts encoded in an image/video and adding this high-level semantic information into the CNN-LSTM framework have improved performance significantly. Specifically, [16, 17] proposed to discover and integrate the rich semantic description, such as objects, scenes, and actions, to benefit the video caption task. Their models jointly learn the dynamics within both visual and textual modalities for video captioning. Although significant performance improvements were achieved, integration of semantic concepts into the LSTM-based caption generation process is still constrained in these ways: semantic features are used only (1) for initialization of the first step of the LSTM or (2) for implementing a soft attention mechanism to the LSTM-based caption generation process. Also, unlike our SeFLA model, previous works using semantic features [11, 12] are limited in that they do not distinguish the dynamic semantic features from the static semantic features. Moreover, they use a relatively simple LSTM model for generating captions.

3. Video Captioning Model

3.1. Model Outline. This study proposes a video captioning model that utilizes semantic features along with visual features that describe video clips for more effective video captioning. Direct linking of visual features extracted by a convolutional neural network (CNN), such as ResNet and VGG, to LSTM-based textual caption generation may ignore the rich intermediate/high-level description, such as objects, scenes, and actions. To address the issue, this study employs additionally two different types of semantic features: dynamic and static semantic features. As shown in Figure 2(a), dynamic semantic feature corresponds to the action taking place within the input video. In contrast, static semantic feature refers to the object, person, and background present in the video, as illustrated in Figure 2(b). In other words, verbs in caption sentence correspond to dynamic semantic feature and nouns to static semantic features.

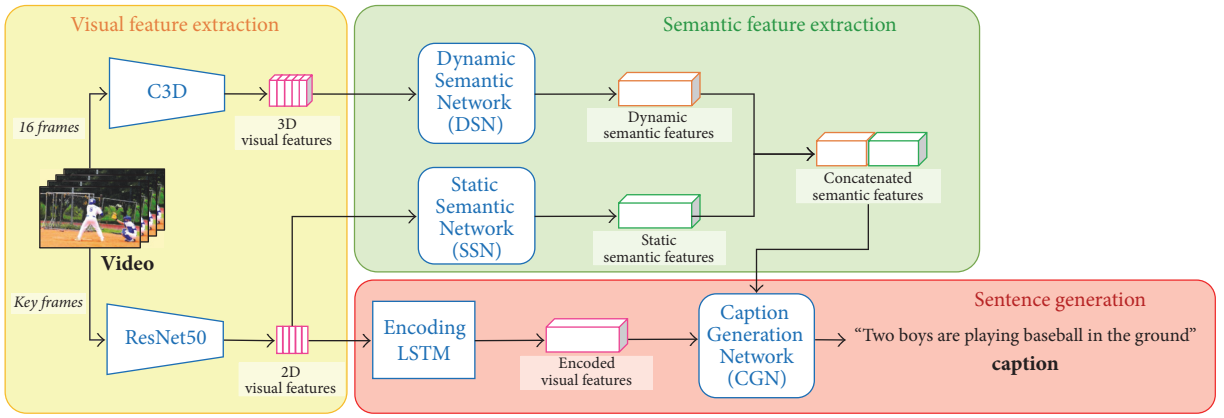


Figure 3: Overall framework of the proposed video captioning model.

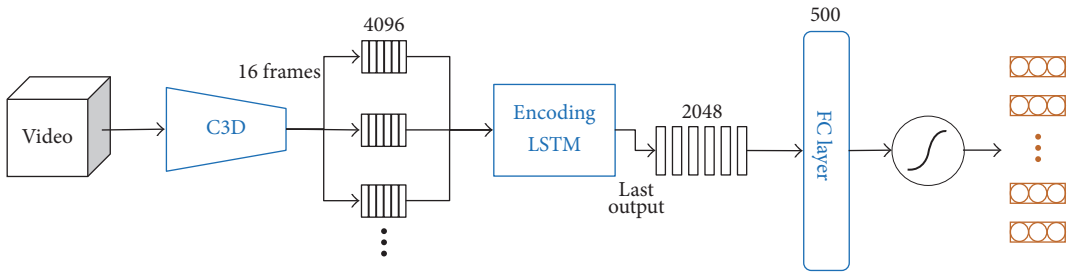


Figure 4: Dynamic semantic feature network (DSN).

The overall framework of the proposed SeFLA model is illustrated in Figure 3. It consists of three main parts: visual feature extraction, semantic feature extraction, and sentence generation. First, visual features required for caption generation are extracted using pretrained ResNet and C3D. The extracted visual features then serve as inputs to the Dynamic Semantic Network (DSN) and static semantic network (SSN), which will be introduced in Section 3.2. In particular, DSN uses visual features of C3D which effectively represents dynamic feature of the video, whereas SSN uses ResNet which represents static feature. Dynamic semantic features and static semantic features are then extracted from each network, which are subsequently concatenated and utilized as inputs to the caption generation network (CGN) introduced in Section 3.3, at each time step. Moreover, CGN applies the attention mechanism on the concatenated semantic features to treat each semantic feature differently at each time step. Visual features extracted via ResNet serve as inputs not only to the SSN, but also to the LSTM that encodes visual features. The final output from the encoding LSTM is given to the initialization step of the CGN. At every time step, the CGN determines the specific semantic feature to focus on and computes the probability distribution of the words. Afterwards, the caption is generated based on the probability distribution of the output words.

3.2. Semantic Feature Learning. To implement caption generation using semantic features, they must first be identified from the input video. As explained previously, semantic features can be categorized into dynamic semantics that illustrate actions and static semantics that denote objects, persons, and backgrounds; clear-cut differences exist between these. Identification of a dynamic semantic feature based on a single frame is hardly possible and requires observation of the video clip for a certain period. On the other hand, a static semantic feature corresponds to an object, person, or background present in a particular moment and, thus, can be identified using a single frame. Hence, extraction of dynamic and static semantic features was carried out separately and treated as a matter of multilabel classification in this study. Dynamic semantic features were extracted based on visual features that effectively illustrated temporal and spatial features of the video, while static semantic features were extracted based on visual features that effectively described the spatial features.

The DSN suggested in this research is shown in Figure 4. First, visual features were extracted in clips, intervals of 16 frames, using a pretrained C3D CNN (see (1)) to exploit the visual features that effectively described the temporal and spatial features of the video. $v_1^i, \ldots, v_{16}^i$ in (1) denotes each single frame in the ith clip and n_v the total number of frames. The extracted visual features (c_i) are then encoded (e) using the LSTM RNN model, as shown in (2). c_t refers to the visual feature corresponding to a single clip encoded at the current time step (t), while h_{t-1} denotes the previous hidden state of the LSTM.

$$c_i = \mathrm{C3D}\left(v_1^i, \ldots, v_{16}^i\right), \quad i \in \left\{0, 1, \ldots, \frac{n_v}{16}\right\} \tag{1}$$

$$e = \mathrm{LSTM}\left(c_t, h_{t-1}\right). \tag{2}$$

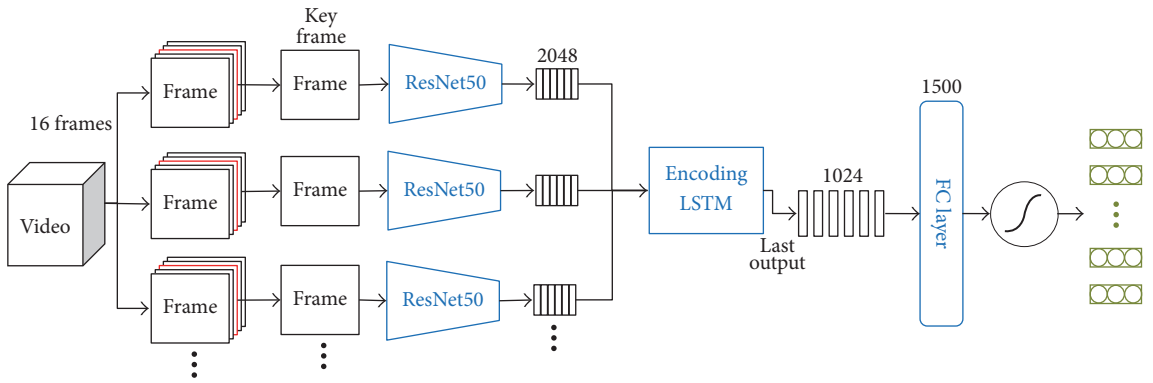


Figure 5: Static semantic feature network (SSN).

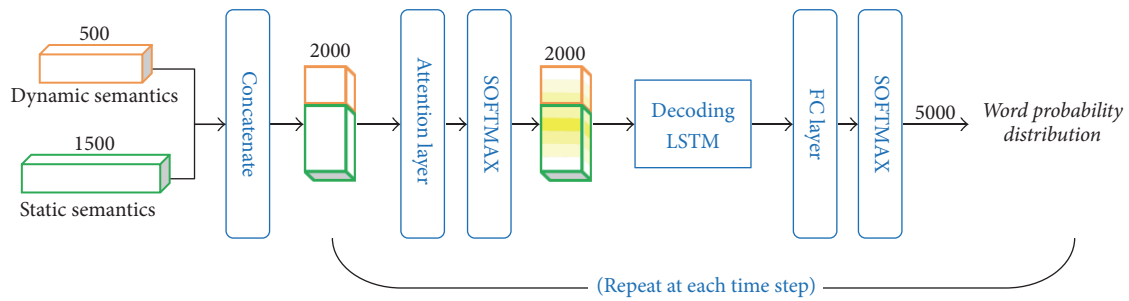


Figure 6: Caption generation network (CGN).

Next, the probability distribution of the dynamic semantic feature (p_d) can be determined from the encoded visual feature (e), fully connected layer, and sigmoid activation function, as shown in (3), where W_d denotes the weight values to be trained and b_d the bias.

$$p_d = \text{sigmoid}\left(W_d \cdot e + b_d\right). \tag{3}$$

The proposed SSN is shown in Figure 5. First, temporal features are extracted from the pretrained ResNet CNN to utilize visual features that effectively describe the spatial features of the video. The video is then divided into clips, intervals of 16 frames each, as expressed by (4), and the visual features (r_i) extracted from the 8th frame (v_8^i) in the ith clip are encoded (e) by LSTM in the manner shown in (5). Subsequently, as shown in (6), fully connected layer and sigmoid activation function are used to determine the probability distribution (p_s) of the SSN.

$$r_i = \text{ResNet}\left(v_8^i\right), \quad i \in \left\{0, 1, \ldots, \frac{n_v}{16}\right\} \tag{4}$$

$$e = \text{LSTM}\left(r_t, h_{t-1}\right) \tag{5}$$

$$p_s = \text{sigmoid}\left(W_s \cdot e + b_s\right). \tag{6}$$

3.3. Attention-Based Caption Generation. This research proposes an attention-based caption generation network (CGN) for effective caption generation using multimodal features, as illustrated in Figure 6.

CGN receives dynamic semantic features and static semantic features as inputs at every time step and identifies the probability distribution. Both dynamic and static semantic features are concatenated and serve as inputs for the attention layer. Conventionally, it is advisable to direct attention to an object within the video if the word to be generated is a noun, and similarly the focus should be on a behavior observed in the video if the word is a verb. In this paper, the attention layer is used to determine the type of semantic feature to focus on at the current time step when implementing a CGN. At the attention layer, a weight value (W_a) that reflects the semantic feature to focus on at a current time step (t) is applied to compute semantic features. The weighted semantic feature (a_t) can be calculated using (7), where s_t refers to the semantic feature given as input and b_a denotes the bias.

$$a_t = \text{softmax}\left(W_a \cdot s_t + b_a\right). \tag{7}$$

The converted semantic features serve as inputs to the decoding LSTM. The decoding LSTM learns sentence structures based on the input semantic features (a_t) and output a status value (h_t) that indicates the word to be generated at the current time step (t), as expressed in (8). The initial hidden state ($h_{t=0}$) of the decoding LSTM is initialized as the final hidden status value of the encoding LSTM that encodes the temporal features.

$$h_t = \text{LSTM}\left(a_t, h_{t-1}\right). \tag{8}$$

The outputs from the decoding LSTM are given as inputs to the fully connected layer. The probability distribution (p_t),

which indicates appropriate words at the current time step (t), is computed in the fully connected layer according to (9), where W_p refers to the weight value to be trained, h_t the inputs given from the decoding LSTM, and b_p the bias.

$$p_t = \text{softmax}\left(W_p \cdot h_t + b_p\right). \tag{9}$$

At each time step, attention values for input semantic features are computed, and the probability distribution of words is output via decoding LSTM and fully connected layer. Then, the output words are strung in order from the first to the keyword denoting the end of statement "⟨EOS⟩" to generate a caption.

4. Performance Evaluation

4.1. Dataset. To train and assess the performance of the CGN suggested in the study, the MSVD dataset and a video caption dataset collected from YouTube videos were used. The MSVD dataset consisted of 1970 YouTube video clips and 80,000 caption statements corresponding to such clips. The sizes of training, cross-validation, and test sets were 1200, 100, and 670, respectively.

On the other hand, the MSR-VTT (Video-to-Text) dataset consists of around 10,000 web video clips. The video clips are classified into 20 categories: music, people, gaming, sports/actions, news/events/politics, education, TV shows, movie/comedy, animation, vehicles/autos, how-to, travel, science/technology, animals/pets, kids/family, documentary, food/drink, cooking, beauty/fashion, and advertisement. They are divided into 6513, 497, and 2990 videos for training, validation, and test sets, respectively. Each video has around 20 natural language captions.

To train the semantic feature networks suggested in the study, training datasets were required. To collect datasets for training, MSVD video caption datasets were used. First, the Part-Of-Speech (POS) tag function in Natural Language Toolkit (NLTK) was used to separate nouns and verbs, while plural nouns and tenses of verbs, past, continuous, and so on, were converted back to their root forms using the lemmatize function in NLTK. Among the extracted verbs, the 500 most frequently appearing words were selected as labelled data for the dynamic semantic features, while 1500 most frequent nouns were chosen as labelled data for static semantic features. A video was labelled with 1 if its caption contained one of the verbs designated as labelled data for dynamic semantic feature, and 0 otherwise. The static dataset was compiled in a similar fashion. Each video contained approximately 7 nouns and 3 verbs present in the datasets. The semantic feature datasets comprised 1200, 100, and 670 examples for training, cross-validation, and test, respectively, like the MSVD caption dataset.

4.2. Model Training. For this research, Keras, a deep learning library in Python, was run in Ubuntu 14.04 LTS environment to implement the proposed models. The hardware specifications for the experiments are as follows: CPU: Intel(R) Core(TM) i7-6700 CPU @ 3.40 GHz, RAM: 32 GB, and GPU: GeForce GTX 1080. Input videos were tailored with uniform sampling such that each video contained 40 clips, and each clip consisted of 16 frames. For the semantic feature networks (SSN and DSN), Adam was used as the model optimization algorithm, and the binary cross-entropy cost function in (10) was used for the loss function. Here, y denotes the actual value, while $\tilde{y}$ indicates the expected value.

$$L_{\text{binary}} = -\left[y \log \tilde{y} + (1 - y) \log (1 - \tilde{y})\right]. \tag{10}$$

Once the semantic feature networks were fully trained, semantic features were extracted from all videos in the caption dataset, which were then used as inputs for the caption generation network (CGN). For the CGN, RMSprop was used as the model optimization algorithm, and the categorical cross-entropy cost function in (11) was selected as the loss function.

$$L_{\text{categorical}} = -\frac{1}{n}\sum_{x}\left[y \log \tilde{y} + (1 - y) \log (1 - \tilde{y})\right]. \tag{11}$$

The batch size and the epoch for learning semantic feature networks (SSN and DSN) were set at 32 and 500, while those for the caption generation network (CGN) were 25 and 50, respectively.

Table 1: Performance of semantic feature networks on MSVD dataset.

Networks	Val-accuracy	Test-accuracy
DSN	99.42%	99.43%
SSN	99.61%	99.64%

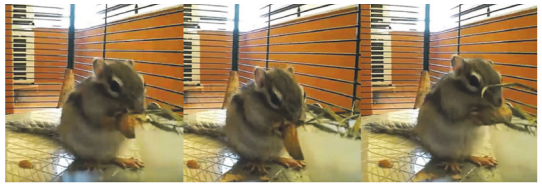
4.3. Experiments. The first experiment was conducted to assess the performance of the semantic feature extraction network suggested in this study. The accuracy for each semantic feature extraction network was calculated with Mean Square Error (MSE) as shown in (12). In (12), n represents the output dimension, y_i the actual value, and $\tilde{y}_i$ the expected value.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \tilde{y}_i\right)^2. \tag{12}$$

The level of performance of each network evaluated using MSE is tabulated in Table 1, where DSN and SSN denote the dynamic and static semantic network, respectively. The recorded values in the table indicate a high accuracy of semantic feature extraction in both networks.

Figure 7 shows the results for the qualitative assessment of both semantic feature networks. As illustrated in the figure, the SSN extracts words that indicate that the subjects are carrying out certain behaviors, whereas the DSN extracts words that describe the behaviors displayed by the subjects.

The aim of the second experiment was to investigate the effects of each semantic feature on caption generation performance. The CGN used in this experiment was kept the same as the selective attention CGN suggested in this study, while the input features were varied. BLEU@N [18] and CIDEr-D [19], which are typical caption generation evaluation metrics, were selected as measures for the performance of CGN. All



Static semantics: squirrel, peanut, chipmunk, nut, rabbit, shell, animal
Dynamic semantics: eat, play, chew

Static semantics: piano, boy, kid, song keyboard, room, music
Dynamic semantics: play, sit, sing

Figure 7: Some examples of semantic features.

Table 2: Comparison of different feature sets on MSVD dataset.

Feature sets	B@1	B@2	B@3	B@4	CIDEr
CGN	66.1	47.8	37.1	26.5	26.4
DSN + CGN	76.0	58.1	45.7	35.8	50.0
SSN + CGN	78.8	63.4	51.4	41.4	77.8
DSN + SSN + CGN	**84.8**	**70.8**	**60.0**	**50.0**	**94.3**

Table 3: Performance comparison with other state-of-the-art models on MSVD dataset.

Models	B@1	B@2	B@3	B@4	CIDEr
SCN [11]	-	-	-	51.1	77.7
LSTM-TSA [12]	82.8	72.0	62.8	52.8	74.0
hLSTMat [10]	82.9	**72.2**	**63.0**	**53.0**	73.8
SeFLA	**84.8**	70.8	60.0	50.0	**94.3**

Table 4: Performance comparison with other state-of-the-art models on MSR-VTT dataset.

Models	BLEU@4
MP-LSTM (V) [1]	34.8
MP-LSTM (C) [1]	35.4
MP-LSTM (V + C) [1]	35.8
SA (V) [2]	35.6
SA (C) [2]	36.1
SA (V + C) [2]	36.6
hLSTMt [10]	37.4
hLSTMat [10]	38.3
SeFLA	**41.8**

evaluation metrics were computed using codes provided by Microsoft COCO evaluation server. CGN in Table 2 depicts the case when captions were generated using solely the visual features, DSN + CGN the case when only DSN was used, SSN + CGN the case when only SSN was used, and finally DSN + SSN + CGN the case when both DSN and SSN were utilized in tandem.

The results in Table 2 indicate that models that utilized semantic feature networks were more effective than the case that only used the CGN. A noteworthy observation is that the DSN + CGN model performed better than the SSN + CGN model. This may be attributed to the effect of the dynamic semantic feature that indicates activity present in the video unlike static semantic feature that can only illustrate objects, persons, and backgrounds. Also, this may be caused by the fact that, in a given caption for a video, there are usually one verb (activity) and multiple nouns (objects). Furthermore, the model incorporating both the DSN and SSN proved to be the most effective, implying that the two semantic feature networks contribute to the caption generation performance independently.

The third experiment was conducted on MSVD dataset for a comparative assessment of the SeFLA caption generation model that was proposed in this study. Table 3 records the performance of SeFLA in comparison with the other models proposed in previous studies. SCN [11] in Table 3 was suggested by Gan et al., while LSTM-TSA [12] and hLSTMat [10] were proposed by Song et al., respectively. SCN and LSTM-TSA incorporate semantic feature networks, while hLSTMat employs an attention-based layered LSTM as the RNN for caption generation. Specifically, both SCN and LSTM-TSA use semantic features as well as visual features. However, unlike our SeFLA, they are limited in that they do not distinguish the dynamic semantic features from the static semantic features.

From Table 3, the performance achieved by SeFLA is observed to be 84.8% and 94.3% on BLEU@1 and CIDEr, respectively. This indicates that SeFLA is more effective by 1.9% and 16.6% than the other models for the respective metrics. However, SeFLA recorded subpar performance in BLEU@2, BLEU@3, and BLEU@4, illustrating that SeFLA, although effective in predicting word by word, is relatively inefficient when consecutively predicting a few words. This observation is also reflective of SeFLA's ineffectiveness in generating prepositional and postpositional particles, in contrast to its superiority in generating nouns or verbs with the help of semantic features. Such a problem might arise due to the lack of datasets to train the CGN on the sentence structures of LSTM. However, in general standards, the caption generating capability of SeFLA using semantic features, as proposed by this study, can be considered efficient.

Table 4 shows the performance comparisons between the SeFLA model and other models on MSR-VTT dataset. (V) denotes that the model uses VGGnet as a CNN model for video encoding, (C) denotes C3D, and (V + C) denotes that the model use both CNN models.

Table 4 shows that the proposed SeFLA model achieved 41.8% BLEU@4 score, that is, 3.5%, better performance than

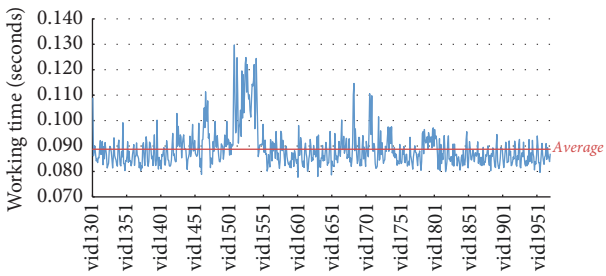


Figure 8: Working time of the SeFLA model on each MSVD test video.

previous studies on MSR-VTT. The result indicates that the SeFLA model has better caption generation performance than previous models with the help of semantic features.

In the fifth experiment, the working time of the SeFLA model, which is the caption generation time, was measured on MSVD test dataset. Note that the feature extraction time is not included in the working time.

Figure 8 shows the results of working time measurement, and the average working time was 0.89 sec. Each working time was affected by the number of words in the generated caption and the length of the input video.

5. Conclusion

This study proposed a deep neural network model capable of effective video captioning. Apart from visual features, the proposed model learns additionally semantic features that describe the video content effectively. In our model, visual features of the input video are extracted using convolutional neural networks such as C3D and ResNet, while semantic features are obtained using recurrent neural networks such as LSTM. In addition, our model includes an attention-based caption generation network to generate the correct natural language captions based on the multimodal video feature sequences. Various experiments, conducted with the two large benchmark datasets: Microsoft Video Description (MSVD) and Microsoft Research Video-to-Text (MSR-VTT), demonstrate the performance of the proposed model. Our future works are as follows. First, a more sophisticated attention mechanism will be incorporated into our SeFLA model for further boosting video captioning. Second, we will investigate how to leverage multimodal features for multiple sentence generation for videos.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the Technology Innovation Program or Industrial Strategic Technology Development Program (10077538, Development of Manipulation Technologies in Social Contexts for Human-Care Service Robots) funded by the Ministry of Trade, Industry & Energy (MOTIE, Korea).

References

[1] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko, "Sequence to sequence - Video to text," in *Proceedings of the 15th IEEE International Conference on Computer Vision, ICCV 2015*, pp. 4534–4542, December 2015.

[2] L. Yao, A. Torabi, K. Cho et al., "Describing videos by exploiting temporal structure," in *Proceedings of the 15th IEEE International Conference on Computer Vision, ICCV 2015*, pp. 4507–4515, December 2015.

[3] Y. Pan, T. Mei, T. Yao, H. Li, and Y. Rui, "Jointly Modeling Embedding and Translation to Bridge Video and Language," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4594–4602, Las Vegas, NV, USA, June 2016.

[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '16)*, pp. 770–778, Las Vegas, Nev, USA, June 2016.

[5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proceedings of the in Proceedings of the International Conference on Learning Representations (ICLR15)*, 2015.

[6] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *Proceedings of the 15th IEEE International Conference on Computer Vision, ICCV 2015*, pp. 4489–4497, December 2015.

[7] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[8] K. Xu, J. Ba, and R. Kiros, "attend and tell: neural image caption generation with visual attention," in *Proceedings of the in Proceedings of International Conference on Machine Learning (ICML15)*, 2015.

[9] M. Zanfir, E. Marinoiu, and C. Sminchisescu, "Spatio-temporal attention models for grounded video captioning," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 10114, pp. 104–119, 2017.

[10] J. Song, L. Gao, Z. Guo, W. Liu, D. Zhang, and H. T. Shen, "Hierarchical LSTM with Adjusted Temporal Attention for Video Captioning," in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, pp. 2737–2743, Melbourne, Australia, August 2017.

[11] Z. Gan, C. Gan, X. He et al., "Semantic Compositional Networks for Visual Captioning," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1141–1150, July 2017.

[12] Y. Pan, T. Yao, H. Li, and T. Mei, "Video captioning with transferred semantic attributes," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 984–992, Honolulu, Hawaii, USA, July 2017.

[13] Y. Yu, H. Ko, J. Choi, and G. Kim, "End-to-end concept word detection for video captioning, retrieval, and question answering," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3261–3269, July 2017.

[14] S. Guadarrama, N. Krishnamoorthy, G. Malkarnenkar et al., "Youtube2text: recognizing and describing arbitrary activities

using semantic hierarchies and zero-shot recognition," in *Proceedings of the 2013 14th IEEE International Conference on Computer Vision, (ICCV '13)*, pp. 2712–2719, December 2013.

[15] J. Xu, T. Mei, T. Yao, and Y. Rui, "MSR-VTT: A large video description dataset for bridging video and language," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 5288–5296, July 2016.

[16] F. Nian, T. Li, Y. Wang, X. Wu, B. Ni, and C. Xu, "Learning explicit video attributes from mid-level representation for video captioning," *Computer Vision and Image Understanding*, 2017.

[17] A. Liu, N. Xu, and Y. Wong, "Hierarchical & multimodal video captioning: discovering and transferring multimodal knowledge for vision to language," *Computer Vision and Image Understanding*, 2017.

[18] K. A. Papineni, S. Roukos, T. Ward, and W. J. Zhu, "BLEU: a method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics (ACL '02)*, pp. 311–318, July 2002.

[19] R. Vedantam, C. L. Zitnick, and D. Parikh, "CIDEr: Consensus-based image description evaluation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (CVPR '15)*, pp. 4566–4575, June 2015.