Incorporating Textual Similarity in Video Captioning Schemes

Konstantinos Gkountakos
Information Technologies Institute
Centre for Research and Technology Hellas
Thessaloniki, Greece
gountakos@iti.gr

Anastasios Dimou
Information Technologies Institute
Centre for Research and Technology Hellas
Thessaloniki, Greece
dimou@iti.gr

Georgios Th. Papadopoulos, Member, IEEE
Information Technologies Institute
Centre for Research and Technology Hellas
Thessaloniki, Greece
papad@iti.gr

Petros Daras, Senior Member, IEEE
Information Technologies Institute
Centre for Research and Technology Hellas
Thessaloniki, Greece
daras@iti.gr

***Abstract*—The problem of video captioning has been heavily investigated from the research community the last years and, especially, since Recurrent Neural Networks (RNNs) have been introduced. Aforementioned approaches of video captioning, are usually based on sequence-to-sequence models that aim to exploit the visual information by detecting events, objects, or via matching entities to words. However, the exploitation of the contextual information that can be extracted from the vocabulary has not been investigated yet, except from approaches that make use of parts of speech such as verbs, nouns, and adjectives. The proposed approach is based on the assumption that textually similar captions should represent similar visual content. Specifically, we propose a novel loss function that penalizes/rewards the wrong/correct predicted words based on the semantic cluster that they belong to. The proposed method is evaluated using two widely-known datasets in the video captioning domain, Microsoft Research - Video to Text (MSR-VTT) and Microsoft Research Video Description Corpus (MSVD). Finally, experimental analysis proves that the proposed method outperforms the baseline approach in most cases.**

***Index Terms*—video captioning, Word2Vec, textual information, encoder-decoder, Recurrent Neural Network (RNN)**

I. Introduction

The number of videos captured on a daily basis and then uploaded on the internet has been increased dramatically due to the wide usage of smart-phone devices. These videos are usually uploaded without a description. However, video captioning approaches aim to generate sentences (captions) that generally describe the visual content of videos. Broadly, video captioning approaches comprise two separate components, a feature extractor that typically extracts the features of the whole video - by sampling among the frames using a fixed number as step - and an encoder-decoder. The second component, that has been inspired from Natural Language Processing (NLP) [1] networks, firstly encodes the visual content - in the form of features - and then assigns it to words that are included in the vocabulary.

In [2], a sequence-to-sequence model that converts the video input to text is proposed. Specifically, the authors make use of a feature extractor in order to extract the features of the videos and then feed them to an encoder-decoder module. The input features are encoded using a Long Short-Term Memory (LSTM) [3] network and then -on the decoding phase- the features are mapped to specific words using also an LSTM network. Additionally, the authors incorporate another modality of information, the optical flow, and they show that can improve the accuracy of the predicted captions.

Based on the aforementioned scheme, a variety of methods have been proposed so far. Recently, methods that make use of bidirectional LSTMs and methods that solve the video captioning problem by incorporating a paragraph module have been proposed. Moreover, attention mechanisms have been widely explored on video captioning domain. More effective feature extractors have been also investigated. Furthermore, reinforcement-learning-based approaches and methods that are based on event detection have been introduced in [4], [5], [6], [7], [8], [9], [10]. From the above analysis, it can be deduced that the video captioning-related literature has in principle focused on visual information analysis, while the respective video captions' similarity has not been investigated, leaving a great potential for further performance improvement.

To address the above issue, a novel method that takes into account the textual similarity of the videos' captions to enhance the training process of the caption architectures is proposed. Based on the hypothesis that textually similar captions describe similar videos, from the visual point of view, a method that penalizes or rewards the predicted captions is proposed in this work. Specifically, the proposed method assigns the words of the vocabulary to specific clusters. Furthermore, a new loss function is introduced in order to penalize or reward the videos that are predicted with a wrong or a correct caption, respectively. The main contributions of the proposed paper are summarized below.

978-1-7281-3401-7/18/$31.00 ©2019 IEEE

- The proposed method takes into account the textual similarity of the captions that have been extracted from the dictionary in the form of cluster vectors, in order to drive the video captioning architectures to encode the visual content and decode it to text in a more effective way.
- The modeling of the proposed method, by adding a penalty-reward function, that makes the architecture agnostic of the feature extractor and the dataset used. Therefore, it can be utilized in conjunction with any baseline architecture.
- The proposed method is evaluated using two video captioning datasets: MSR-VTT [11] and MSVD [12]. After a detailed analysis, it is shown that the proposed method improves significantly the results compared to the baseline approach.

The remainder of the paper, is organized as follows: Related work is discussed in Section II. In Section III, the proposed method is detailed, while experimental results are presented in Section IV. Finally, conclusions are drawn in Section V.

II. Related work

Venugopalan *et al.* [2] proposed a method for video captioning that learns to map a sequence of frames directly to a sequence of words. Specifically, an encoder-decoder LSTM architecture is proposed that not only takes as input the video features, but also incorporates the optical flow modality for generating more accurate captions. More specifically, an architecture that comprises two stacked LSTMs is proposed. The first LSTM network encodes a sequence of frames to a hidden representation while the second one decodes it into a sentence. Methods that are based on attention mechanisms have been also proposed. Gao *et al.* [6] proposed an architecture that incorporates an attention mechanism that makes use of salient features extracted from a Convolutional Neural Network (CNN) [13]. Additionally, they proposed a cross-view model in order to enforce the consistency between the predicted sentences and the visual features. Pu *et al.* [8] proposed an attention-based architecture adaptable on different levels of CNN features.

Bin *et al.* [14] are the first that utilize bidirectional recurrent neural networks in order to explore the temporal structure in video captioning problem. Additionally, Wang *et al.* [15] incorporated a bidirectional model in order to better capture the temporal action proposals from the past, current and future events of the videos. Moreover, they took care of the overlapped events in order to improve the predicted captions. Yao *et al.* [16] pay attention to the feature extractor part. Specifically, they incorporated a 3-D CNN followed by an encoder-decoder for capturing the local spatio-temporal information. Additionally, an attention mechanism is proposed and the whole framework is evaluated on video description domain. Yu *et al.* [4] introduced a hierarchical structure in decoder stage. Specifically, the method consists of two parts: a sentence generator and a paragraph generator. More specifically, the paragraph generator takes as input the embeddings of a sentence and via a recurrent layer the paragraph state is generated. Finally, the output of the paragraph layer is used as the initial state of the sentence generator.

Recently, Shetty *et al.* [17] proposed a method that is using two different kinds of video features, one that consists of features and attributes of objects and one for capturing the motion and the action information. Additionally, the architecture is based on an encoder-decoder scheme and they have also proposed an evaluation model in order to pick the best caption from the pool of candidates generated. Similar to the aforementioned approach, Ma *et al.* [18] proposed a method, named SINet-Caption, that takes into account the interaction among groups of objects. Moreover, the authors explored the effectiveness of coarse-grained and fine-grained information of the key-frames using an attention mechanism.

Hierarchical structures have also been explored on video captioning domain so far. Pan *et al.* [19] proposed a hierarchical recurrent neural encoder in order to exploit the temporal information of videos on encoding stage. Additionally, the proposed method is able to exploit with a more effective way the temporal structure of long videos. Furthermore, actions that are part of a global action can be also exploited. Song *et al.* [20] considered that a caption contains visual and non-visual words, such as articles, and that the second ones can be easily predicted using a natural language model that do not make use of visual features. Specifically, they proposed a hierarchical LSTM framework that can automatically select the frames that describe 'visual' words in order to generate words for video captioning. Finally, Baraldi *et al.* [21] proposed a method that detects the discontinuities in the input video and enables the encoding layer to modify its temporal connectivity by resetting its internal state and memory also.

Reinforcement learning approaches have been also investigated on the video captioning problem. Phan *et al.* [22] proposed a reinforced-based method that in training process the sentences obtained from the annotated captions. Wang *et al.* [9] have also proposed a reinforcement learning approach. Specifically, the proposed architecture consists of two parts. A high level module, called Manager, that learns to design sub-goals and a low-level module, named Worker, that learns to recognize the actions in order to achieve the sub-goal. Moreover, PickNet [23] that has been proposed from Chen *et al.* aims to resolve video captioning problems. Specifically, the architecture consists of an encoder-decoder and, based on reinforcement learning, tries to pick the informative frames. However, to the best of our knowledge, the aforementioned methods do not exploit the frequency of each word in the vocabulary and do not take into account the word context among the vocabulary.

III. Proposed method

In this section, the baseline architecture is previewed and, subsequently, the proposed method is outlined. Additionally, the pre-processing steps are described.

978-1-7281-3401-7/18/$31.00 ©2019 IEEE

A. Pre-processing steps

In this section, the steps in order to transform the data to a suitable form are presented. First of all, each word of the vocabulary is mapped to a word embedding using the word2vec algorithm that has been proposed from Mikolov *et al.* [24]. Specifically, each embedding vector represents a word using a 300-dimensional real-value vector. Due to the fact that the video captioning datasets describe only short-length vocabularies, the usage of generic embeddings is mandatory. Therefore, we make use of the Google news dataset that consists of 1 billion words in order to export more comprehensive word embeddings. More specifically, each word of the dataset's vocabulary is mapped to an embedding from one of the 692K embeddings generated. For words not included in the Google news vocabulary, we perform a string similarity measure, as presented in [25], in order to assign to the most relevant embedding.

As mentioned above, the main goal of the pre-processing steps is to map each word from the vocabulary to a specific cluster. To address this, the clustering algorithm – k-means – that have been proposed by Hartigan *et al.* [26] is adopted. Specifically, the k-means algorithm is repeated for 25 steps. The cosine similarity distance among the clusters' centroids and the word embeddings is taken into account.

B. Proposed approach

As in all cases in the video captioning domain, a baseline architecture that comprises a feature extractor, an encoder and a decoder module has been selected. In order to simplify the implementation of the proposed approach, the method proposed by Venugopalan *et al.* [2], named Sequence to Sequence - Video to Text (S2VT), has been selected as baseline method. As mentioned, the proposed architecture can be applied to any video captioning architecture in the form of an extra penalty/reward function. Due to the fact that each dataset comprises of a different number of words, the pre-processing steps should be performed on each dataset separately.

The proposed approach takes into account the words' context encoded in word2vec embeddings and their frequency of appearance. Each word from the vocabulary is mapped to a specific cluster. This information, in the form of cluster vectors that contain the frequency of appearance for each word, is used as the criterion to the introduced loss function. Equation (1) describes the function that is used in order to decide whether the word x belongs or not to cluster j. The formulation of the predicted ground truth vector is presented in (2). Specifically, j denotes the number of clusters and i denotes the max length of the generated caption while function g is described in (1). Equation (3), similarly to the previous ones, denotes the cluster vector that has been generated from the ground truth caption.

$$g(x) = \begin{cases} 1, x \in C_j \\ 0, x \notin C_j \end{cases}, C = cluster \tag{1}$$

$$PcV_0^j = \sum_0^i (g(w_i)) \tag{2}$$

$$GcV_0^j = \sum_0^i (g(w_i)) \tag{3}$$

$$\|GcV - PcV\| = \frac{\sqrt{(GcV - PcV) * (PcV - GcV)}}{\lambda} \tag{4}$$

Equation (4) denotes the Euclidean distance between the predicted and the ground truth vector, (2) and (3) respectively, while λ declares the effect of the penalty/reward of the proposed loss function. Specifically, (4) calculates the global distance of the predicted caption to the ground truth caption, using as a criterion the distance between the two cluster vectors, the ground truth cluster vector and the predicted one. It should be noted that this value balances the penalty/reward functionality. If the value is < 1 the loss value of the predicted caption is decreased (reward), while if the value is > 1 the loss value is increased (penalty), and, obviously, if the value is equal to 1 there is no penalty/reward and consequently only the cross entropy loss is applied. It should be noted, that the introduced loss function is applied in combination with the cross-entropy loss by a simple multiplication. In Fig. 1 the proposed architecture is presented. Specifically, the basic processing steps of the two videos are depicted. The main modules (feature extractor, encoder, decoder) are depicted on the left. Subsequently, the processing of the vocabulary, the generated clusters and the cluster vectors are depicted on the right. Additionally, the Euclidean distance between the two (ground truth and predicted ones) cluster vectors and the cross entropy loss are placed on the center of the figure.

IV. Results

A. Employed Datasets

In order to evaluate the performance of the proposed approach two widely-used video captioning datasets are used, MSR-VTT [11] and MSVD [12]. MSR-VTT dataset contains 10000 videos clips from 20 categories. Additionally, each clip has been manually annotated with a set of 20 captions. Furthermore, the split-settings proposed by [11] are adopted. 6513 videos comprise the training set, 497 videos the validation set and the remaining 2990 the test set. The vocabulary of MSR-VTT dataset consists of 16860 unique words.

The second dataset that was used during the evaluation is the MSVD. The collection comprises 1970 videos from YouTube while the annotations of the sentences are provided by the owner of the organization. Additionally, the annotation process has been carried out using multilingual workers and the videos have been annotated in more than 20 languages. In this work, only the videos that have been annotated using the English language were used, counted to 1517. Each video is described with an average of 22 captions and their duration is between 10 to 25 seconds. Due to the fact that some videos are no longer available for download, the total number of videos that was used is equal to 931. However, we follow the splits – using same percentage– proposed by Venugopalan *et al.* [27]. Specifically, the training set consists of 60%, the validation

978-1-7281-3401-7/18/$31.00 ©2019 IEEE

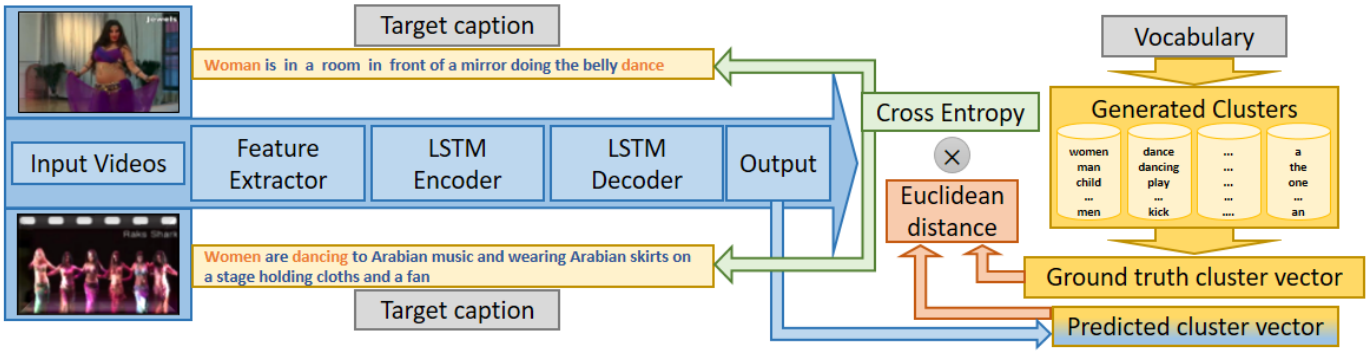

Fig. 1. The proposed architecture of the processing steps of the two videos is depicted. The main modules (feature extractor, encoder, decoder) of the proposed architecture are depicted on the left, while the vocabulary, the generated clusters and the cluster vectors on the right. Additionally, the Euclidean distance between the two cluster vectors –introduced loss – and the cross-entropy loss are placed on the center of the figure.

set 5% and the test set 35%. Finally, for each video 40 frames were sampled while the vocabulary consists of 5821 unique words.

B. Implementation details

For training both the baseline and the proposed approach, the selected number of epochs was 3000. Adam [28] was selected as optimizer with initial learning rate and weight decay 10^{-4} and 10^{-5}, respectively. The learning rate was scheduled to be decayed by 0.8 every 200 epochs. Moreover, the batch size was selected to be equal to 512. We compared the models that achieved the min loss value on the validation set. Finally, all implementations were carried out using PyTorch [29] library on an Nvidia GTX 1070 GPU with 8GB memory.

C. Evaluation metrics

For the evaluation, commonly-used metrics on the video captioning domain were used. Specifically, both baseline and the proposed models were evaluated using METEOR [30], BLUE@1-4 [31], ROUGE-L [32] and CIDEr-D [33] metrics.

D. Evaluation results

In order to transform the visual input to feature vectors, the features have been extracted using two different networks as feature extractors, Inception-v4 [34] and ResNet-152 [35]. The dimensions of the extracted features are 40×2048 when ResNet-152 is used and 40×1536 when Inception-v4 is used. Both networks were pre-trained on the ImageNet [13] dataset. 40 frames are sampled from each video. Due to the fact that each video has a duration of 10 to 30 seconds, at least one frame for each second of the video has been taken into account when the MSR-VTT dataset is processed. The impact of the number of clusters has been also investigated. Specifically, the number of clusters in the experiments that we have conducted was 10, 20 and 100. Furthermore, the value of λ was selected experimentally. More specifically, the experiments were carried out using multiple λ values equal to 1.0, 0.7 and 0.3.

In Table I the results of the different experimental settings using Inception-v4 as feature extractor are presented. From a detailed examination of the provided results, it can be seen that the proposed method performs better compared to the baseline approach. Specifically, the proposed method performs an improvement of 44%, 10%, 12%, 30% when evaluated using Blue@4, METEOR, ROUNGE-L and CIDEr-D respectively. More specifically, the proposed method exhibits better results when the number of clusters is low, 10 or 20. This happens, to the best of our knowledge, because the introduced loss function works as a global penalty/reward in combination with the existing cross-entropy loss. Thus, a small number of clusters leads the proposed model to minimize the introduced loss value that explicitly describes a global assignment of words to clusters. Additionally, in Table I the optimum value of the factor λ can be observed. With λ set to 0.3 the proposed method exhibits significantly better results. This improvement of the performance is expected when a small value of λ is used. As mentioned, the introduced loss function describes a global sentence loss and, therefore, a higher value leads the model to learn more abstract sentences. It should be noted that this fact is penalized during evaluation.

In Table II, the evaluation results using ResNet-152 as feature extractor are depicted. The experiments were carried out using the same configuration as the experiments where the Inception-v4 was used as feature extractor. As it can be seen, the proposed method outperforms the baseline approach. Specifically, when the factor λ is equal to 0.3 and the number of clusters is low, 10 or 20, the proposed approach exhibits a significant improvement compared to the baseline approach. More specifically, the proposed method improves the results of Blue@4, METEOR, ROUNGE-L and CIDEr-D by 37%, 11%, 10%, 30%, respectively.

The results of Tables I and II prove that the proposed method is agnostic to the feature extractor used. Additionally, a low value of the contributing factor λ to the overall cross entropy loss is more efficient. Moreover, the number of clusters that the vocabulary is assigned must be small. A detailed analysis of the experiments, shows that the generated cluster vectors

978-1-7281-3401-7/18/$31.00 ©2019 IEEE

TABLE I
PERFORMANCE EVALUATION RESULTS USING INCEPTION-V4 AS FEATURE EXTRACTOR ON THE MSR-VTT DATASET. DIFFERENT CONFIGURATIONS OF THE NUMBER OF CLUSTERS AND THE PARAMETER λ ARE PROVIDED

Metric	Baseline	Proposed λ=1			Proposed λ=0.7			Proposed λ=0.3		
		10 Clusters	*20 Clusters*	*100 Clusters*	*10 Clusters*	*20 Clusters*	*100 Clusters*	*10 Clusters*	*20 Clusters*	*100 Clusters*
Blue@1	68.89	68.67	70.01	68.93	67.98	68.94	68.44	**76.45**	69.47	69.60
Blue@2	49.59	49.18	50.52	49.79	48.64	49.77	49.07	**60.30**	50.16	50.52
Blue@3	34.46	34.24	35.42	34.98	33.71	35.86	34.20	**45.83**	35.18	35.35
Blue@4	23.37	23.32	24.19	23.86	22.71	23.79	23.12	**33.76**	24.04	24.04
METEOR	23.56	23.21	23.66	23.49	23.53	23.66	23.51	**25.87**	23.57	23.57
ROUNGE-L	50.78	50.83	51.57	51.19	50.65	51.12	50.61	**56.74**	51.17	51.44
CIDEr-D	28.37	28.13	29.32	28.70	28.66	28.99	28.70	**36.97**	28.64	29.57

TABLE II
PERFORMANCE EVALUATION RESULTS USING RESNET-152 AS FEATURE EXTRACTOR ON THE MSR-VTT DATASET. DIFFERENT CONFIGURATIONS OF THE NUMBER OF CLUSTERS AND THE PARAMETER λ ARE PROVIDED

Metric	Baseline	Proposed λ=1			Proposed λ=0.7			Proposed λ=0.3		
		10 Clusters	*20 Clusters*	*100 Clusters*	*10 Clusters*	*20 Clusters*	*100 Clusters*	*10 Clusters*	*20 Clusters*	*100 Clusters*
Blue@1	69.89	70.19	69.80	69.71	69.60	70.08	70.33	75.01	**75.91**	71.22
Blue@2	49.97	51.50	50.71	50.41	50.87	50.91	51.45	57.81	**59.11**	52.48
Blue@3	34.84	36.74	35.83	35.35	35.77	35.89	36.38	43.00	**44.42**	37.30
Blue@4	23.57	25.42	24.65	23.98	24.23	24.64	24.97	31.09	**32.30**	25.73
METEOR	23.28	23.91	23.87	23.61	23.81	23.83	24.11	25.15	**25.88**	24.13
ROUNGE-L	50.95	51.79	51.46	51.48	51.88	51.72	52.11	55.50	**56.10**	52.49
CIDEr-D	28.50	30.46	29.01	28.84	30.24	29.80	30.25	36.10	**37.05**	30.95

should be no greater than the max length sequence of captions, which in our experiments is equal to 28. This happens because a large number of clusters generates sparse cluster vectors that subsequently increase the introduced global loss and make the minimization problem more difficult. Consequently, this leads the model to generate more abstract sentences. Furthermore, in Fig.2 an indicative result of the proposed method compared to the baseline approach is presented. The most relevant caption of the 20 ground truth captions is presented. Specifically, on the first row, both the baseline and the proposed approach perform a satisfying caption prediction. Moreover, the rows two and three represent promising and not satisfying results, respectively.

TABLE III
PERFORMANCE EVALUATION RESULTS USING INCEPTION-V4 AS FEATURE EXTRACTOR ON THE MSVD DATASET. FACTOR λ IS EQUAL TO 0.3 AND THE NUMBER OF CLUSTERS IS EQUAL TO 10, 20 AND 100 AS CAN BE OBSERVED ON VARIABLE C

Metric	Baseline	Proposed C=10	Proposed C=20	Proposed C=100
Blue@1	62.73	65.47	**66.23**	64.23
Blue@2	47.55	51.06	**51.89**	48.62
Blue@3	37.87	42.23	**42.70**	39.17
Blue@4	29.13	33.66	**34.10**	30.29
METEOR	24.00	24.17	**24.91**	24.72
ROUNGE-L	57.14	59.70	**59.81**	57.89
CIDEr-D	57.54	61.81	**65.39**	59.12

In order to verify the robustness of the proposed method, experiments have been carried out on an additional dataset. The best configuration settings using the MSR-VTT dataset have been selected. The factor λ is set equal to 0.3 and the number of clusters equal to 10, 20 and 100. In Table III the performance of the proposed method using the MSVD dataset is depicted. As it can be observed, the proposed method outperforms the baseline approach in all cases. The proposed approach improves the performance significantly when the number of clusters is 20. More specifically, the proposed method increases the performance by 17%, 4%, 5% and 14% in terms of Blue@4, METEOR, ROUNGE-L and CIDEr-D, respectively.

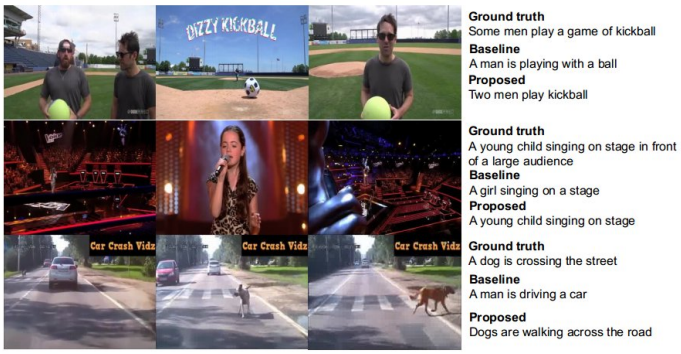

Fig. 2. Indicative results using MSR-VTT dataset are presented. The most relevant -from the 20 sentences- to the baseline model ground truth caption has been selected and referred as ground truth. Additionally, the sentences predicted by the baseline and the proposed model are also depicted.

V. CONCLUSION

In this work, a novel loss function is proposed in order to improve the performance of video captioning techniques using the textual information. In particular, a supervision mechanism for guiding the video captioning learning process, by taking into account the video captions' similarity in correspondence

with the visual content was proposed. More specifically, the proposed method makes use of the textual information in the form of cluster vectors so as to perform a kind of global sentence similarity. It is proved that the proposed approach is agnostic of the feature extractor that may be used. Furthermore, the introduced loss function not only penalizes the captions that have been miss-classified on predefined clusters, but also rewards the captions that are predicted correctly. The experimental results also demonstrate that the optimal number of clusters depends on the length of the dataset's vocabulary. Future work will include investigation of end-to-end architectures that could generate clusters while the video captioning problem is resolved and vice-versa.

ACKNOWLEDGMENT

The work presented in this paper was supported by the European Commission under contract H2020-787061 ANITA.

REFERENCES

[1] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 160–167.

[2] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko, "Sequence to sequence-video to text," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4534–4542.

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

[4] H. Yu, J. Wang, Z. Huang, Y. Yang, and W. Xu, "Video paragraph captioning using hierarchical recurrent neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4584–4593.

[5] Á. Peris, M. Bolanos, P. Radeva, and F. Casacuberta, "Video description using bidirectional recurrent neural networks," in *International Conference on Artificial Neural Networks*. Springer, 2016, pp. 3–11.

[6] L. Gao, Z. Guo, H. Zhang, X. Xu, and H. T. Shen, "Video captioning with attention-based lstm and semantic consistency," *IEEE Transactions on Multimedia*, vol. 19, no. 9, pp. 2045–2055, 2017.

[7] M. Zanfir, E. Marinoiu, and C. Sminchisescu, "Spatio-temporal attention models for grounded video captioning," in *Asian Conference on Computer Vision*. Springer, 2016, pp. 104–119.

[8] Y. Pu, M. R. Min, Z. Gan, and L. Carin, "Adaptive feature abstraction for translating video to text," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[9] X. Wang, W. Chen, J. Wu, Y.-F. Wang, and W. Yang Wang, "Video captioning via hierarchical reinforcement learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4213–4222.

[10] L. Zhou, Y. Zhou, J. J. Corso, R. Socher, and C. Xiong, "End-to-end dense video captioning with masked transformer," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8739–8748.

[11] 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 IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5288–5296.

[12] D. L. Chen and W. B. Dolan, "Collecting highly parallel data for paraphrase evaluation," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 2011, pp. 190–200.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[14] Y. Bin, Y. Yang, F. Shen, X. Xu, and H. T. Shen, "Bidirectional long-short term memory for video description," in *Proceedings of the 2016 ACM on Multimedia Conference*. ACM, 2016, pp. 436–440.

[15] J. Wang, W. Jiang, L. Ma, W. Liu, and Y. Xu, "Bidirectional attentive fusion with context gating for dense video captioning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7190–7198.

[16] L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville, "Describing videos by exploiting temporal structure," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4507–4515.

[17] R. Shetty and J. Laaksonen, "Frame-and segment-level features and candidate pool evaluation for video caption generation," in *Proceedings of the 2016 ACM on Multimedia Conference*. ACM, 2016, pp. 1073–1076.

[18] C.-Y. Ma, A. Kadav, I. Melvin, Z. Kira, G. AlRegib, and H. P. Graf, "Grounded objects and interactions for video captioning," *arXiv preprint arXiv:1711.06354*, 2017.

[19] P. Pan, Z. Xu, Y. Yang, F. Wu, and Y. Zhuang, "Hierarchical recurrent neural encoder for video representation with application to captioning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1029–1038.

[20] J. Song, Z. Guo, L. Gao, W. Liu, D. Zhang, and H. T. Shen, "Hierarchical lstm with adjusted temporal attention for video captioning," *arXiv preprint arXiv:1706.01231*, 2017.

[21] L. Baraldi, C. Grana, and R. Cucchiara, "Hierarchical boundary-aware neural encoder for video captioning," in *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*. IEEE, 2017, pp. 3185–3194.

[22] S. Phan, G. E. Henter, Y. Miyao, and S. Satoh, "Consensus-based sequence training for video captioning," *arXiv preprint arXiv:1712.09532*, 2017.

[23] Y. Chen, S. Wang, W. Zhang, and Q. Huang, "Less is more: Picking informative frames for video captioning," *arXiv preprint arXiv:1803.01457*, 2018.

[24] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[25] A. Islam and D. Inkpen, "Semantic text similarity using corpus-based word similarity and string similarity," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 2, no. 2, p. 10, 2008.

[26] J. A. Hartigan and M. A. Wong, "Algorithm as 136: A k-means clustering algorithm," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, no. 1, pp. 100–108, 1979.

[27] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko, "Translating videos to natural language using deep recurrent neural networks," *arXiv preprint arXiv:1412.4729*, 2014.

[28] D. Kinga and J. B. Adam, "A method for stochastic optimization," in *International Conference on Learning Representations (ICLR)*, vol. 5, 2015.

[29] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," in *NIPS-W*, 2017.

[30] M. Denkowski and A. Lavie, "Meteor universal: Language specific translation evaluation for any target language," in *Proceedings of the ninth workshop on statistical machine translation*, 2014, pp. 376–380.

[31] K. 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*. Association for Computational Linguistics, 2002, pp. 311–318.

[32] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," *Text Summarization Branches Out*, 2004.

[33] R. Vedantam, C. Lawrence Zitnick, and D. Parikh, "Cider: Consensus-based image description evaluation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 4566–4575.

[34] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning." in *AAAI*, vol. 4, 2017, p. 12.

[35] 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*, 2016, pp. 770–778.