Adaptive Curriculum Learning for Video Captioning

SHANHAO LI, BANG YANG, AND YUEXIAN ZOU (Senior Member, IEEE)
AISPLAB, School of Electronic and Computer Engineering, Peking University, Shenzhen 518055, China
Corresponding author: Yuexian Zou (zouyx@pku.edu.cn)

This work was supported by the National Natural Science Foundation of China (NSFC) under Grant 62176008.

ABSTRACT A portion of the data in video captioning datasets are noisy and unsuitable for models to learn at early stages, e.g., there could be a generic 4-word-long caption lacking distinctive details of video content and a 19-word-long description with rare words and complex structure given the average caption length of around 9. The conventional training method, i.e., learning by random sampling indiscriminately from the whole training set, may cause data bias problems and undermine the model performance. In this work, we present a novel learning strategy, Adaptive Curriculum Learning (ACL), to alleviate the adverse effect of such problems. The main idea of our approach is to allow a model to learn from the data within its competence. Specifically, a difficulty measurement is first defined to evaluate the learning difficulty of video-caption pairs, and training data can be ranked accordingly. Then, based on the learning difficulties and model competence, an adaptive sampling approach is developed to provide suitable training subsets for video captioning models in different training stages. Notably, our proposed ACL is applicable to most existing video captioning works as it requires no modifications of the model architecture. Extensive experiments are conducted on mainstream benchmarks, i.e., MSVD and MSR-VTT datasets. The results show that both RNN-based and Transformer-based models achieve consistent performance improvements with our ACL strategy.

INDEX TERMS Video captioning, curriculum learning, RNN, transformer.

I. INTRODUCTION

Video captioning research aims at automatically generating a descriptive sentence from a given video and has many potential applications in video retrieval, video surveillance, and assistance by describing to visually impaired people. It requires an algorithm that would first comprehend the main event of video content and then translate it into a fluent and descriptive sentence. Most video captioning models adopt the classic encoder-decoder architecture [1], which relies on convolution neural networks (CNN) to extract video features and dedicated recurrent neural networks (RNN) or recently fashionable Transformer [2] to decode features into texts.

Recent advances in video captioning mainly focus on designing more sophisticated networks for fine-grained feature extraction [3]–[5], better multimodal fusion [6]–[9], more controllable caption generation [9]–[12], etc. Despite the improvements in network design, existing methods pay little attention to the quality of video captioning data and train the models by random sampling indiscriminately from the whole training set. As shown in Fig. 1, a portion of captions from even the widely used MSR-VTT dataset [13] are annotated in two undesirable ways: 1) Some are generic and partial, lacking rich details of the video content; 2) Others are lengthy and complicated for models to learn at early training stages. Indiscriminately training on such data can cause severe data bias problems for video captioning models and result in underperformance regardless of the model design. On the contrary, a dynamic training strategy that enables models to learn from suitable data at different stages could potentially promote model performance.

Research on curriculum learning, a training strategy that is inspired by the human learning process and can be traced back to [14], believes that deep neural models can achieve better performance if the training examples are presented in a meaningful order, i.e., from easy ones to gradually complex ones. While extensive studies on curriculum learning in various tasks of computer vision [15]–[18] and natural
language processing [19]–[22] have proven to be effective, it is strenuous to directly apply curriculum learning in video captioning because of the following two reasons: (1) Appropriate difficulty measurements for both videos and captions are needed, and (2) the widely used easy-to-hard learning strategy may not be the best choice given the aforementioned noise of those biased captions.

Motivated by the above discussion, we attempt to address the data bias problem in video captioning by proposing Adaptive Curriculum Learning (ACL), a training strategy that utilizes training data by selecting the most suitable ones for models to learn at different stages. The main idea behind this is to first train the models with simple yet informative video-caption data and then adaptively add more and even noisy data, e.g., superficial and prolonged ones, as the training proceeds. Specifically, we design an IDF-like measurement to define the learning difficulty of video-caption pairs, which is based on the amount and diversity of words within captions and objects within videos. Furthermore, since we want to exclude the noisy data and learn from the most suitable video-caption pairs at first, we propose a method of setting the difficulty of the initial training subset, and this enables us to train and expand from arbitrary levels of difficulty to suit our need. While conventional curriculum learning usually uses a fixed learning schedule to match training data with the gradually increased model competence [22], we instead propose an adaptive schedule for measuring the model ability and training subset selection. Combined with the aforementioned methods, we conduct experiments on the two public datasets, i.e., MSVD [23] and MSR-VTT [13] datasets, to evaluate the effectiveness, and the results show that our proposed ACL consistently boosts model performance without introducing any model parameters.

Our contributions can be concluded as follows:

- To the best of our knowledge, this work is the first attempt to apply curriculum learning, an alternative to conventional random sampling, to alleviate the data bias problem within the video captioning datasets.
- We design a novel training approach Adaptive Curriculum Learning (ACL) which assesses the difficulty of video-caption pairs and maintains a training subset of varied initial difficulties and adaptive sizes based on the model ability.
- Experiments are conducted on two mainstream video captioning datasets with both RNN-based and Transformer-based models. The results demonstrate the effectiveness of our proposed ACL, which can constantly improve model performance.

II. RELATED WORKS
A. VIDEO CAPTIONING

In the deep learning era, video captioning research has significantly benefited from the rapid development of computer vision and natural language processing.
Venugopalan et al. [1] introduce the Encoder-Decoder architecture into the field of video captioning. Convolutional neural network (CNN) as the encoder is used on frames of videos to attain representative features, and then these features are fed into LSTM to decode into words. Researchers [6], [7], [24] have found that CNN models pre-trained on different tasks as encoders can extract valuable video information for the downstream captioning tasks, such as Inception [25] and ResNet [26] from image classification, 3D [27] from video classification, and even VGGish [28] from audio classification. Moreover, instead of the mean pooling used [1] that ignores the temporal information within videos, CNN-detected features can be better utilized with attention-based methods [6], [29]–[33] that adjust the spatial and temporal focus. Among these works, Anderson et al. [29] propose bottom-up and top-down attention mechanisms that help model focus on the object and salient regions. Gao et al. [34] argue that attention on visual information is of no use when generating non-visual words and should instead rely on language context information. They propose an adaptive attention mechanism that switches between visual and language information for decoding. More recently, Yang et al. [35] present a non-autoregressive and coarse-to-fine decoding procedure using Transformers [2] as the decoder and can achieve faster inference speed.

B. DATA BIAS IN VISION LANGUAGE TASK

Data bias in this work refers to the problem of training on and learning little from those easy and hard examples during the early stages of training. In a different context, the definition of the problem might vary. For example, Chen et al. find spelling mistakes and the use of special characters in the MSR-VTT dataset and propose an annotation cleaning method [36] that focuses on improving the quality of the captions, which requires a manual and sophisticated design to resolve the problems. There is also an existing work [37] in image captioning that measures the quality of captions and requires modification to the loss function when training. Hou et al. in [38] investigate the data imbalance problem at the vocabulary level and utilize part-of-speech (POS) tags as visual cues to alleviate such bias when translating video to languages. This work requires an extra POS tag generator and model adoption for tags. By comparison, our work is less invasive to training stage or model architecture and thus more flexible than those previous works on data bias.

C. CURRICULUM LEARNING

Bengio et al. first propose curriculum learning (CL) in [14], which is inspired from the observation that humans learn much better when the examples are presented in a meaningful order, and gradually introducing more complex ones. And since then, researchers have exploiting CL in various fields of study, such as computer vision [15]–[18] and natural language processing tasks [19]–[22]. Specifically, Zhang et al. [39] research on several measures for linguistic difficulties, such as sentence length and word frequencies, and training schemes such as easy-to-hard and even hard-to-easy ones. Instead of ranking data by difficulty, Kumar et al. [21] design the curriculum by the noise of data and select bins of data through reinforcement learning. Platanios et al. [22] simplify the data selection process as model competence that can be calculated by a monotonically increasing function, such as linear or root functions, and show that CL can decrease training time significantly while improving accuracy for natural machine translation tasks. Liu et al. [40] apply CL to medical report generation by attaining different sorted batches with different metrics and switching between batches to utilize the limited medical data.

III. APPROACH

In this section, the proposed Adaptive Curriculum Learning (ACL) is explained in detail. ACL consists of two parts: difficulty measurements for video-captions pairs (in Section III-A) and adaptive data selection (in Section III-B). The former calculates the learning difficulties of the data from the training set, and the latter selects suitable subsets of data for models at different training stages. Furthermore, ACL explores different initial difficulties for the subset selection to alleviate data bias within the video captioning dataset.

A. DIFFICULTY MEASUREMENTS

In the following, we first introduce how to measure the difficulty of a video and its corresponding captions and then the overall difficulty of a video-caption pair.

a: VIDEO DIFFICULTY

Although video difficulty can be measured by various aspects, we here only focus on the object-oriented one and suggest that the amount and diversity of objects within video content matter. Formally, given a video \( V = \{v_m\}_{m=1}^M \) of length \( M \), video difficulty is defined as:

\[
\text{d}_{\text{video}}(V) = \ln \sum_{m=1}^M (v_m^{N_o} + v_m^{N_t} + 1).
\]

(1)

where \( \ln(\cdot) \) is used to reshape the distribution, \( v_m^{N_o} \) is the total number of objects detected at \( m \)-th frame whereas \( v_m^{N_t} \) the number of object types. The extra \( 1 \) is to avoid calculating...
\[ \ln(0). \] In practice, we directly utilize pre-trained object detectors to obtain both \( v^*_m \) and \( v^* \).

**b: CAPTION DIFFICULTY**

Sentence length is an intuitive measurement for caption difficulty. However, a sentence with multiple rare words can have the same length as those with many commonly used words, but the former is harder to learn and should have a higher caption difficulty. Therefore, we follow the work [22] from neural machine translation and apply the word rarity metric to estimate the caption difficulty. Given a corpus of \( K \) captions \( \{C_k\}_{k=1}^K \), where each caption \( C_k \) has \( T_k \) words, i.e., \( C_k = \{w_{k,t}\}_{t=1}^{T_k} \), we first calculate the frequency of each word in the vocabulary \( W \) of the corpus:

\[ \forall w \in W, \quad \hat{p}(w) = \frac{1}{N_{\text{total}}} \sum_{k=1}^K \sum_{t=1}^{T_k} 1(w_{k,t} = w). \]  
(2)

where \( 1(\cdot) \) is the indicator function and \( N_{\text{total}} \) is the total length of all captions in the corpus, i.e., \( N_{\text{total}} = \sum_k |C_k| \). Next, the difficulty for each caption \( C_k \) can be calculated as:

\[ d_{\text{caption}}(C_k) = -\sum_{t=1}^{T_k} \log \hat{p}(w_{k,t}). \]  
(3)

**c: VIDEO-CAPTION DIFFICULTY**

The difficulty for a video-caption pair \((V, C)\) can be formulated as the weighted sum of difficulties from both video and caption aspects:

\[ d_{\text{pair}}(V, C) = \alpha \cdot d_{\text{video}}(V) + (1 - \alpha) \cdot d_{\text{caption}}(C). \]  
(4)

where \( \alpha \) is a hyperparameter for difficulty balance. In practice, both \( d_{\text{video}}(V) \) and \( d_{\text{caption}}(C) \) are divided by their max value before summation so that the \( \alpha \) can achieve better balance regardless of their different orders of magnitude.

**B. ADAPTIVE SUBSET SELECTION**

We rank data from easy to hard according to the proposed difficulty measures, and then different subsets of video-caption pairs are selected for different training stages. To alleviate data bias, we introduce a hyperparameter \( \gamma \) that sets the initial subset difficulty, so that in the vital early stages, models are free from the noisy data that are either partial or complicated and focus on the most suitable data.

1) **SUBSET SIZE**

In order to determine the subset size, we consider two methods from predefined and adaptive points of view.
TABLE 1. Effect of $\alpha$ (the hyperparameter to balance video difficulty and caption difficulty). The ARB models trained on the MSVD benchmark with and without our ACL are used for the case study here. ARB without ACL is trained by conventional random sampling from the whole training set and can be seen as the baseline. With our proposed ACL, the model achieves the best scores on most metrics when $\alpha$ is set to 0.2.

<table>
<thead>
<tr>
<th>Model</th>
<th>$\alpha$</th>
<th>BLEU@4</th>
<th>METEOR</th>
<th>ROUGE-L</th>
<th>CIDEr</th>
</tr>
</thead>
<tbody>
<tr>
<td>w/o ACL</td>
<td>0.0</td>
<td>47.5</td>
<td>34.8</td>
<td>71.1</td>
<td>88.8</td>
</tr>
<tr>
<td>ARB</td>
<td>0.2</td>
<td>48.3</td>
<td>35.4</td>
<td>72.0</td>
<td>90.5</td>
</tr>
<tr>
<td>w/ ACL</td>
<td>0.4</td>
<td>47.5</td>
<td>34.4</td>
<td>71.1</td>
<td>86.2</td>
</tr>
<tr>
<td></td>
<td>0.6</td>
<td>48.1</td>
<td>35.1</td>
<td>71.3</td>
<td>89.2</td>
</tr>
<tr>
<td></td>
<td>0.8</td>
<td>48.1</td>
<td>34.9</td>
<td>71.7</td>
<td>87.4</td>
</tr>
<tr>
<td></td>
<td>1.0</td>
<td>48.6</td>
<td>34.5</td>
<td>71.1</td>
<td>84.7</td>
</tr>
</tbody>
</table>

$a$: PREDEFINED SIZE

Models tend to improve the capability rapidly in the early training stages, and then converge slowly as the training continues. Therefore, the subset size should increase rapidly at first and smoothly later on. Based on this observation, the predefined subset size (first proposed in [22]) is calculated by a square root function:

$$s_p(t) = \min(1, \sqrt{\frac{1 - s_0^2}{T} + \frac{t^2}{T^2}}).$$

(5)

where $t$ is the training time, $s_0$ is the initial percentage, and $T$ is the time when the curriculum learning stage completes, and the model should learn from the whole training set afterward. In the following experiments, curriculum learning with the predefined size is referred to as PCL.

$b$: ADAPTIVE SIZE

We also introduce an adaptive subset size selecting method. Instead of using the square root function to simulate the change of the model ability, we directly use the metric result from model performance on the validation set, and allow the model to pace the subset size by itself:

$$s_a(t) = \min(1, s_a(t - 1) + \max(\frac{\Delta M_t}{M}, \Delta c));$$

$$s_a(0) = s_0.$$ 

(6)

where $M$ is the metric result for a fully trained baseline model ($M$ is attained beforehand), $\Delta M_t$ is the improvement of the model at time $t$ compared with the previous training stage (i.e., at time $t - 1$), and $\Delta c$ guarantees the minimum increment in subset size. The curriculum learning equipped with the adaptive size is referred to as ACL.

2) INITIAL SUBSET DIFFICULTY

Instead of directly applying curriculum learning, which exposes the easy and noisy video-caption pairs to the model at the early stages, we propose a strategy that can select subsets at arbitrary initial difficulty and avoid such undesirable data. It is implemented by setting the left and right boundary with a hyperparameter $\gamma$. Note that the left and right boundaries refer to the percentile of certain video-captioning pairs. For example, suppose 20% of the training data are less difficult than a $(V, C)$, then this video-caption pair $(V, C)$ is at the 20th percentile. And a subset with 25% as the left boundary and 75% as the right boundary contains video-captionation pairs between the 25th percentile and the 75th percentile and has a size that is 50% of the whole training set. The left and right boundaries are defined as follows:

$$b_L(t) = \max(0, \gamma(1 - s(t)));$$

$$b_U(t) = \min(1, \gamma(1 - s(t)) + s(t)).$$

(7)

where $\gamma$ is the hyperparameter that determines the initial difficulty. The conventional easy-to-hard fashion in most curriculum learning works is a special case of our proposed strategy if $\gamma$ is set to 0, and we even support the hard-to-easy strategy simply by setting $\gamma$ to 1. We can start training on the subset of specified difficulties by setting $\gamma$ to any number in between.

C. TRAINING PROCEDURE OF ACL

The proposed curriculum learning takes effect on how to select the training data, as illustrated in Fig. 2. First, we calculate the difficulty of every video-caption pair from the whole training set and sort by their difficulties. The method for calculating difficulty is explained in Section III-A. Then, during each training stage, we select the most suitable subset, according to Section III-B, for models to train on. After each stage, the model is evaluated on the validation set and provides feedback for subset selection in the next stage. After the curriculum learning stage completes, we would train the model on the whole training set. It is clear that our curriculum learning strategy requires little change to the current model trainer and is therefore applicable to most existing works.

IV. EXPERIMENTS

A. DATASETS

In our experiments, we utilize two standard datasets for video captioning. The first one is MSVD [23]. It consists of 1970 short video clips, each of which is paired with roughly 41 English captions annotated via crowdsourcing. We follow the existing works [30], [34], [35] and split the dataset into 1200, 100, and 670 videos clips for training, validation, and testing, respectively. The second one is MSR-VTT [13], a relatively large-scale video captioning dataset covering 20 categories, which contains 10,000 video clips and 200,000 video-captioning pairs. We follow the official splits and take 6513 clips for training, 497 for validating, and 2990 for testing.

B. BASELINES

To evaluate the proposed learning strategies, we select three captioning models as baselines, including TopDown [29], ParAbhLSTMMat [34], and ARB [35]. We would like to apply the proposed strategy to various architectures, that is why the former two RNN-based models and the latter transformer-based one are adopted in our experiments.
TABLE 2. Experiments on the proposed difficulty measurements. The ARB models trained with 25% and 50% of the training set from MSR-VTT datasets are used here. The video-caption pairs are sorted by their difficulties, and then a continuous range of data (i.e., Data Range* in percentile) is selected. During training, the models are trained solely on the particular range of data.

<table>
<thead>
<tr>
<th>Model</th>
<th>Data Size</th>
<th>Data Range*</th>
<th>BLEU@4</th>
<th>METEOR</th>
<th>ROUGE-L</th>
<th>CIDEr</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARB</td>
<td>25%</td>
<td>0th - 25th</td>
<td>39.0</td>
<td>25.9</td>
<td>58.5</td>
<td>42.4</td>
</tr>
<tr>
<td></td>
<td>25th - 50th</td>
<td>36.0</td>
<td>27.3</td>
<td>57.6</td>
<td>41.3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>50th - 75th</td>
<td>31.0</td>
<td>27.0</td>
<td>54.1</td>
<td>33.4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>75th - 100th</td>
<td>23.8</td>
<td>25.0</td>
<td>51.1</td>
<td>23.6</td>
<td></td>
</tr>
<tr>
<td></td>
<td>50%</td>
<td>0th - 50th</td>
<td>40.2</td>
<td>27.3</td>
<td>59.6</td>
<td>45.8</td>
</tr>
<tr>
<td></td>
<td>50th - 100th</td>
<td>32.7</td>
<td>27.5</td>
<td>55.3</td>
<td>35.8</td>
<td></td>
</tr>
</tbody>
</table>

TABLE 3. Effect of our proposed strategies on three baselines on MSVD and MSR-VTT datasets. By incorporating the PCL and ACL, the performances of three different baseline models on two datasets can be substantially boosted across major evaluation metrics on the benchmark. And models trained with ACL are noticeably better than baselines, which is trained with random sampling from the whole training set.

<table>
<thead>
<tr>
<th>Model</th>
<th>MSVD</th>
<th>MSR-VTT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>BLEU@4</td>
<td>METEOR</td>
</tr>
<tr>
<td>TopDown [29]</td>
<td>51.8</td>
<td>34.1</td>
</tr>
<tr>
<td>w/ PCL</td>
<td>51.9</td>
<td>34.5</td>
</tr>
<tr>
<td>w/ ACL</td>
<td>53.6</td>
<td>35.2</td>
</tr>
<tr>
<td>ParAHL-LSTM [34]</td>
<td>48.9</td>
<td>34.4</td>
</tr>
<tr>
<td>w/ PCL</td>
<td>50.0</td>
<td>34.4</td>
</tr>
<tr>
<td>w/ ACL</td>
<td>51.0</td>
<td>34.7</td>
</tr>
</tbody>
</table>

C. METRICS
Baselines trained with the conventional random sampling and with our proposed strategies are evaluated with four common metrics. These benchmarks include BLEU [41], METEOR [42], ROUGE-L [43] and CIDEr [44]. For all metrics, a higher score indicates better performance. Specifically, BLEU@n is a modified n-gram precision score with a penalty for short sentences. METEOR measures by precision and recall between generated and reference sentences. ROUGE-L considers the longest common subsequence between sentences and is often used in measuring the performance of automatic summarization. Out of four metrics, CIDEr is the only one designed for vision to language tasks such as video captioning and image captioning.

D. EXPERIMENTAL SETTINGS
To calculate video difficulty, we firstly use Faster R-CNN detector [45] with the ResNet-50 backbone [26] to detect objects from uniformly sampled 20 frames per video, and then filter out the detection results with lower confidence than a threshold of 0.8. For feature extraction, we adopt the same settings as in the recent work [35], where ResNet-101 [26] and 3D version of ResNeXt-101 [46] are used, and all category tags in MSR-VTT are also used in the models.

During the experiments, the maximum length of captions is set to 30, and beam search [47] with the beam size of 6 is used for all models. The time T in Predefined Size is set to 30, roughly the same time as the Adaptive Size expands to the whole training set, and α is set to 0.2 for difficulty balance. Δc is set as 2%, and σ0 is 1%. Initial difficulty γ is set around 0.25 in the strategy comparison as it shows the best potential. The learning rate is set to 0.0005 for RNN-based models and 0.0003 for the Transformer-based model. The best model parameters are selected based on the model performance of the validation set after 50 training epochs.

E. RESULTS AND ANALYSIS
We first investigate the impact of different difficulty balance (controlled by α) and initial difficulty (controlled by γ) on the model performance, while validating the difficulty measurements. And then, we report three model results trained with different learning strategies on MSVD and MSR-VTT datasets, with analysis from both qualitative and quantitative points of view. It should be noted that all experiments with and without our proposed ACL can be seen as ablation studies, i.e., Table 1 and 3.

1) EFFECT OF DIFFICULTY BALANCE
We conduct the experiments on the benchmark MSVD dataset to analyze the effect of the hyperparameter α, which balances the video difficulty and caption difficulty. The results of the ARB model [35] with and without the proposed ACL are shown in Table 1. We can find that almost all our approaches with different values of α can outperform the baseline model, which further proves the effectiveness of our approach. Besides, we can notice that when the α is set to 0.2, our approach can achieve the best results across all
2) EFFECT OF DIFFICULTY MEASUREMENTS
We experiment with the difficulty measurements proposed in Section III-A by training the ARB model solely on different ranges of sorted continuous data, and the results are shown in Table 2. For example, the data range “0th - 25th” means that the video-caption pairs used in model training are the data with the least difficulties and consist of 25% of the total training data from the dataset. The hyperparameter of difficulty balance ($\alpha$) is set to 0.2 when sorting the data pairs, and all models are trained with random sampling from the designated data range. We can observe that as the subset of data ranking higher in percentile, the model attains fewer scores, which proves that our difficulty measurements distinguish the easy from the hard, and indicates that the model struggles to learn from complex video-caption pairs. Furthermore, compared with a subset size of 25%, training on the subset of more data shows greater model performance, and hence we would continue training the model on the whole training set after the curriculum learning stage completes.

3) EFFECT OF INITIAL DIFFICULTY
Hyperparameter $\gamma$ controls the difficulty of the initial training subset. As shown in Fig. 4, $\gamma$ is tested out from 0 to 3/4 with an interval of 1/8. It can be found that the model performance peaks when the model is trained with $\gamma = 1/4$. And starting by training models on easier data (e.g., $\gamma = 0$) or harder ones (e.g., $\gamma = 3/4$) shows less potential. Though training solely on the easier data can outperform harder ones, starting with the easiest video-caption pairs may not be the best choice. We consider that the former could mislead the model into delivering less informative captions, whereas learning the latter in the early stage is beyond the model ability. Therefore, models perform best if trained on detailed video-caption pairs within the models’ ability.

4) EFFECT OF PCL AND ACL
The three models in Table 3, i.e., TopDown, ParAhLSTM, and ARB, are trained by three different learning strategies. For each model, using the conventional random sampling from the whole training set can be seen as baselines. And below baselines are the two options for increasing subset size (i.e., Predefined Size and Adaptive Size). While both are substantially better than the baselines, the ACL strategy constantly outperforms the PCL strategy. We believe that the ACL, by using the performance on the validation set, has a more accurate subset size for models at different training stages. Moreover, the universal improvement over CIDEr up to 1.7 score in MSVD dataset and 2.5 score in MSR-VTT dataset, indicates that our ACL with tuned initial difficulty $\gamma$ are superior to the conventional strategy and can help the existing models to achieve better performance.

5) QUALITATIVE ANALYSIS
Fig. 5 shows six examples of ground-truth captions (GT) and model-generated captions. Specifically, these captions are generated by the ARB model trained on the MSR-VTT
dataset. We highlight the caption improvements of the model trained with our ACL over the baseline trained with random sampling (RS). Considering videos (a), (c), and (f) in Fig. 5, the model trained with ACL successfully describes the place where the videos are recorded, i.e., mat, kitchen, and runway, and these details are missing in the captions by RS. In videos (b) and (c), the vague descriptions by RS, something and food, are replaced with more detailed terms, computer program and dish. Moreover, for videos (e) and (f), the model with ACL also notices blue shirt and black dress that the subjects are wearing and infers that there is a camera pointed to the subject in video (e). In videos (d) and (f), the baseline model does not give an accurate or even valid description. In contrast, the model trained with ACL correctly gives the subjects as well as the actions. Overall, the model with our ACL yields better captioning results in terms of accuracy and detail.

V. FUTURE RESEARCH DIRECTIONS
A limitation of our approach is that α is a hyperparameter, which indicates that we have to perform extensive experiments to select the optimal value. A more desirable solution would be to adaptively determine the value of α, maybe in terms of the output distribution during the training stage, to realize more efficient optimization. We will further seek to address this limitation in future work. Besides, it can be interesting to apply ACL to other broad multimodal tasks, such as image captioning and visual question answering.

VI. CONCLUSION
In this paper, we proposed a novel strategy, Adaptive Curriculum Learning (ACL), for training video captioning models to alleviate data bias in the datasets. ACL samples a suitable subset of data from the training set for different training stages. By exploring the initial difficulty of training subsets, we found that the model learns best if starting with descriptive video-caption pairs that fit into the model ability. We also proposed an adaptive approach of setting subset size to match with the model ability and further enhance the curriculum learning effect. Experimental results demonstrate the effectiveness of our proposed ACL, which consistently boosts the performance of RNN-based and Transformer-based models and promotes accurate and detailed captioning.

REFERENCES