Open-book Video Captioning with Retrieve-Copy-Generate Network

Ziqi Zhang\textsuperscript{1,2,3}, Zhongang Qi\textsuperscript{2}, Chunfeng Yuan\textsuperscript{1,*}, Ying Shan\textsuperscript{2}, Bing Li\textsuperscript{1}, Ying Deng\textsuperscript{5}, Weiming Hu\textsuperscript{1,3,4}

\textsuperscript{1}NLPR, Institute of Automation, Chinese Academy of Sciences
\textsuperscript{2}Applied Research Center (ARC), Tencent PCG
\textsuperscript{3}School of Artificial Intelligence, University of Chinese Academy of Sciences
\textsuperscript{4}CAS Center for Excellence in Brain Science and Intelligence Technology
\textsuperscript{5}School of Aeronautical Manufacturing Engineering, Nanchang Hangkong University

\{ziqi.zhang,cfyuan,bli,wmhu\}@nlpr.ia.ac.cn, \{zhongangqi,yingsshan\}@tencent.com

Abstract

In this paper, we convert traditional video captioning task into a new paradigm, i.e., Open-book Video Captioning, which generates natural language under the prompts of video-content-relevant sentences, not limited to the video itself. To address the open-book video captioning problem, we propose a novel Retrieve-Copy-Generate network, where a pluggable video-to-text retriever is constructed to retrieve sentences as hints from the training corpus effectively, and a copy-mechanism generator is introduced to extract expressions from multi-retrieved sentences dynamically. The two modules can be trained end-to-end or separately, which is flexible and extensible. Our framework coordinates the conventional retrieval-based methods with orthodox encoder-decoder methods, which can not only draw on the diverse expressions in the retrieved sentences but also generate natural and accurate content of the video. Extensive experiments on several benchmark datasets show that our proposed approach surpasses the state-of-the-art performance, indicating the effectiveness and promising of the proposed paradigm in the task of video captioning.

1. Introduction

Video captioning is one of the most important vision-language tasks, and it seeks to automatically describe what has happened in the video according to the visual content. Recently, many promising methods \cite{36, 22, 24, 34, 2} have been proposed to address this task. These methods mainly focus on learning the spatial-temporal representations of videos to fully tap visual information and devising novel decoders to achieve visual-textual alignment or controllable decoding. In general, there exist some drawbacks for most of the existing work: first, since the video content is the only source of input, the generation process lacks appropriate guidance, resulting in the generations of more generic sentences; second, the memory or knowledge domain of the model is fixed after training and cannot be expanded or revisited unless retraining.

To address these issues, we propose an Open-book Video Captioning paradigm. We first compare the two cross-modal tasks for better illustration: Video-Text Retrieval (VTR) and Video Captioning (VC). VTR is a discriminative task that can access all the information of visual and textual modalities all the time; VC as a generative task can only produce words based on current generated words and visual information, which is more challenging than VTR. Instead of performing the VC task directly, we propose to convert it into two-stages: we first perform VTR to search for sentences relevant to the given video from the text corpus; then, we leverage the retrieval sentences as extra hints or guidance for caption generation. During the inference, the generator can generate words based on the video content or directly copy expressions from retrieved sentences.

*Corresponding author.
The flexible VTR and the changeable corpus provide the possibility for the model’s extension or revision.

The inspiration for the proposed paradigm comes from the Open-domain Question Answering task [18, 10, 15], which requires a system to answer any questions utilizing large-scale documents. Open means not providing the system with documents containing the correct answers directly but requires the system to retrieve documents related to the question from massive corpus and then generate the correct answer based on them. We claim that the open-domain mechanism is also effective in cross-modal interaction area and can thus be applied to the video captioning task.

This mechanism essentially extends the knowledge domain of the model learned only from the labeled data. It is known that producing large-scale, high-quality labeled data is extremely laborious and time-consuming. Instead, a model learns to collect related references, distinguish useful hints, abstract and summarize information from external weakly labeled or unlabeled documents breaks through the limitations of labeled data. This is valuable especially for the industry-scale video platforms where hundreds of millions weakly labeled or unlabeled data are generated every day. Unlike traditional semi-supervised learning that utilizes fixed weakly labeled or unlabeled examples directly for training, the proposed paradigm makes the model learn to extract useful information from changeable weakly labeled or unlabeled corpus directly for inference.

To realize the aforementioned open-book video captioning, we introduce a novel Retrieve-Copy-Generate (RCG) network. We introduce a Video-to-Text Retriever to search for video-content-relevant sentences from the corpus containing the whole sentences of the training set. Our retriever follows the Bi-encoders [14] structure and utilizes both motion and appearance features to search desired sentences efficiently and effectively. For the example in Fig.1, the top retrieved sentences contain expressions “on a mat”, “does somersaults”, and “someone watches”, which describe the given video accurately. Then, the retrieved sentences and the visual features are passed to the generator. A novel Copy-Mechanism Generator is introduced, which dynamically decides whether to copy the expressions directly from multi-retrieved sentences or generate new words from the video contents. The model combines the information from the video content and the words copied from retrievals to generate the final caption “A kid doing a somersault on a mat while a boy watches”, which is much better than the generic caption “A little girl practices in a room”.

The contributions of this work are listed as follows:

(1) We propose to solve the video captioning task with an open-book paradigm, which generates captions under the guidance of video-content-retrieval sentences, not limited to the video itself.

(2) We introduce a novel Retrieve-Copy-Generate network to tackle this task, where an improved cross-modal retriever is utilized to provide hints for generator, and a copy-mechanism generator is proposed for dynamical copying and better generation.

(3) The extensive experimental results highlight the benefits of combining cross-modal retrieval with copy-mechanism generation for the video caption task. The proposed approach achieves state-of-the-art results on VATEX and superior performance on MSR-VTT.

2. Related Works

Video Captioning. Reviewing the early retrieval-based visual description methods can be roughly divided into two categories: [21, 9, 16] cast the problem as retrieval from a visual space problem and transfer the retrieved descriptions to a novel image/video; [5, 25, 11] cast the problem as retrieval from a multimodal space problem. Although retrieval-based methods can find human-like sentences with similar semantics to the video, it is challenging to generate an entirely correct description due to limited retrieval samples. With the advent of the encoder-decoder framework, most of the current work is studying how to better use visual features [35, 23, 1, 36, 22, 24] and design elaborate models [34, 2, 28, 13] to generate sentences directly. However, the diversity and controllability of sentences generated in this way are not satisfactory. We coordinate the classical retrieval-based method with modern encoder-decoder method to generate descriptions with diverse expressions and accurate video contents.

Video-Text Retrieval. Video-Text Retrieval is a fundamental discriminative vision-language task that helps to learn the semantic alignment of different modalities. In general, there are two kinds of model architectures, i.e., Bi-encoders [4, 6, 19, 20] and Cross-encoders [26, 30, 33, 3]. Bi-encoders map the query and candidates into a common feature space with two separate encoders. Since there is no interaction between their features, Bi-encoders are lower-accuracy but very efficient during evaluation. Compared with it, Cross-encoders yield rich interactions between query and candidates by integrating features at an early stage. This helps to gain a higher-accuracy but steep computational cost. We make efficient and effective improvements to the Bi-encoders based model, which is more applicable to our paradigm.

Retrieval Augmented Generation Tasks. A series of NLP works utilize the retrieved knowledge for better generation, such as open-domain question answering (Open-QA), neural machine translation(NMT), and dialog generation. REALM [10] and ORQA [17] show promising results on the Open-QA task by combining masked language models with a differentiable retriever. DPR [15] implements a dense embedding-based retriever to replace traditional sparse retrievers and achieves significant performance. RAG [18] combines this learnable retriever in
3.3. Effective Video-to-Text Retriever

The major function of our retriever is to find the top-k most similar sentences \( z \) given video \( x \) in a massive retrieval corpus \( \mathcal{Z} \). Note that \(|\mathcal{Z}|\) can be large enough to cover all the video contents ideally (the corpus contain nearly 0.3 million pieces of sentences in our implement), and \( k \) is usually small, such as \( 1 \sim 30 \).

The video-to-text retriever applies Bi-encoders architecture: the textual encoder \( E_{txt}(\cdot) \) maps all sentences in the corpus \( \mathcal{Z} \) into \( d \)-dimensional vectors and constructs a candidate dataset; the visual encoder \( E_{vis}(\cdot) \) maps the video \( x \)
into a $d$-dimensional vector as a query. The whole retrieval model is trained by metric learning, which embeds visual and textual modalities into a joint high-dimensional semantic space. Subsequently, the similarity between the video and text is defined as the dot-product of their embedding vectors:

$$\text{sim}(x, z) = E_{\text{txt}}(z) \cdot E_{\text{vis}}(x).$$ (2)

In this way, the sentences whose embeddings are top-$k$ closest to the video embedding are sorted by the retriever. We use a simple but effective way to build these two encoders.

**Textual Encoder.** Given a sentence $z = \{s_1, \ldots, s_L\}$, each word is first input to a bi-LSTM to generate a sequence of $d$-dimensional context-aware word embeddings $w = \{w_1, \ldots, w_L\}$:

$$\bar{w}_t = \text{LSTM}(W_s, s_t, w_{t-1}; \eta_s),$$
$$\hat{w}_t = \text{LSTM}(W_s, s_t, w_{t+1}; \eta_s),$$
$$w_t = (\bar{w}_t + \hat{w}_t)/2,$$

where $W_s$ is a learnable word embedding matrix and $\eta_s$ denotes the parameters of LSTM.

Subsequently, all the embeddings are aggregated to a single vector as the overall representation. For simplicity, we denote the aggregation function as $\text{Agg}(\cdot; v)$ which utilizes the multiplicative attention mechanism, where the parameter $v \in \mathbb{R}^d$ can be viewed as a learnable core that gives higher weights to more discriminative features:

$$\text{Agg}(w; v) = \sum_{t=1}^{L} \alpha_t w_t,$$ (4)
$$\alpha_t = \text{softmax}(v^T w_t).$$ (5)

Thus, the word embeddings are aggregated to a single vector via $\bar{e}_w = \text{Agg}(w; v_a)$, where $v_a$ is the parameters of the word aggregation function.

**Visual Encoder.** It is assumed that appearance features $x^a = \{x^a_1, \ldots, x^a_K\}$ and motion features $x^m = \{x^m_1, \ldots, x^m_K\}$ together constitute the representation of the video $x$. Each feature is the original feature after linear transformation to $d$-dimensional. The video is usually divided into $K$ key frames and segments.

For visual encoder, we directly aggregate features at different moments according to their importance, since the sequential information has been involved in motion features, and appearance features only provide discrimination for the main objects in the video. We aggregate motion embeddings via $\bar{e}_m = \text{Agg}(x^m; v_m)$ and appearance embeddings via $\bar{e}_a = \text{Agg}(x^a; v_a)$ by reusing the aggregation function defined by Eq 4, where $v_m, v_a \in \mathbb{R}^d$ are the parameters of two modalities’ aggregation functions.

We take the average of appearance and motion similarities as the final video-text similarity:

$$\text{sim}(x, z) = \bar{e}_w^T (\bar{e}_m + \bar{e}_a)/2,$$ (6)

where $\bar{e} = \frac{e}{||e||_2}$ is the operation of $L_2$ normalization.

**Pre-training and Retrieval.** The training of our cross-modal retriever follows the contrastive learning, where each positive pair $(x^+, z^+)$ should be closer than any other negative pairs $(x^+, z^-)$ and $(x^-, z^+)$ in a mini-batch. The max-margin ranking loss function pushes the hardest negative pair $\Delta$ distance away from the positive pair, as follows:

$$L_{rel} = [\Delta + \text{sim}(x^+, z^-) - \text{sim}(x^+, z^+)]_+ + [\Delta + \text{sim}(x^-, z^+) - \text{sim}(x^+, z^+)]_+,$$ (7)

where $[x]_+ = \max(x, 0)$ and $\Delta$ is the slack coefficient.

The sentences belonging to other videos in the mini-batch are all negative samples of this video and vice versa.

Due to the independent structure of Bi-encoders, the textual embeddings can be calculated offline in advance for efficient evaluation. Given video $x$ as the query, top-$k$ most relevant retrieved sentences can be found following Eq 6, and the probability of retrieved sentence $z_t$ is estimated as:

$$p_{\text{voc}}(z_t|x) = \text{softmax}(\text{sim}(x, z_t)), \quad z_t \in \{z_1, \ldots, z_{\text{topk}}\}.$$ (8)

### 3.2. Copy-mechanism Caption Generator

To generate captions based on the video content and the retrieved sentences, we design a novel copy-mechanism caption generator, which consists of the Hierarchical Caption Decoder and the Dynamic Multi-pointers Module. We describe the proposed generator in detail as follows:

**Hierarchical Caption Decoder.** It consists of attention-LSTM and language-LSTM. Formally, the attention-LSTM tries to focus on different visual features in the feature dimension. The current hidden state of the attention-LSTM $h^a_t$ depends on the previous hidden state $h^a_{t-1}$ and generated word $y_{t-1}$:

$$h^a_t = \text{LSTM}(W_e y_{t-1}, h^a_{t-1}; \theta_a),$$ (9)
$$c_t^v = \text{Att}(h^a_t, x, x; \theta_v),$$ (10)

where, $\text{Att}(\text{query}, \text{key}, \text{value}; \theta)$ denotes standard additive attention module with parameters $\theta$ for simplify; $W_e$ is the word embedding matrix.

Then, the language-LSTM aggregates the current state $h^a_t$ and the visual context $c_t^v$ to generate the probability distribution of the fixed vocabulary $p_{\text{voc}}$ at each time step:

$$h^l_t = \text{LSTM}(h^a_t, c_t^v, h^l_{t-1}; \theta_l),$$ (11)
$$p_{\text{voc}} = \text{softmax}(W_{\text{voc}} h^l_t + b_{\text{voc}}),$$ (12)

where $p_{\text{voc}} \in \mathbb{R}^{1 \times V}$, $\theta_v, \theta_a, \theta_l$ are parameters of the visual attention module and hierarchical-LSTMs; $W_{\text{voc}}$ and $b_{\text{voc}}$ are all learnable parameters.
Dynamic Multi-pointers Module. In Sec.3.1, we get the top-k most similar retrieved sentences to the given video. We also leverage the bi-LSTM$(\cdot; \theta_z)$ to encode these retrieved sentences to $z = \{z_1, \cdots, z_{\text{topk}}\}$. Each retrieved sentence $z_i$ consists of a set of words with their embeddings $z_i = \{z_{i1}, \cdots, z_{iL}\}$.

To draw on the expressions in multiple retrieved sentences, we propose the multi-pointers module improved on Pointer-Networks. At each decoding step $t$, the multi-pointers module acts on each retrieved sentence $z_i$ separately, uses the hidden state $h^t_i$ as the query to attend $L$ words, and produces the word probability distribution $p_{\text{ret},i} = \{p_{\text{ret},i}^1, \cdots, p_{\text{ret},i}^L\}$ of the corresponding sentence:

$$p_{\text{ret},i}^t = \text{Att}(h^t_i, z_i, z_i; \theta_r),$$

where $\text{Att}(\cdot)$ is the additive attention module with parameters $\theta_r$; $c^t_{i,t}$ denotes the context of the retrieved sentence which is the weighted summation of $z_i$ by $p_{\text{ret},i}$. Since not all the words in the retrieved sentence are valid, the model needs to decide whether to copy or generate dynamically. The probabilities of copying words from each retrieved sentence $z_i$ are determined by the semantic context of retrieved sentence $c^t_{i,t}$ and the decoder’s state $h^t_i$:

$$p_{\text{copy},i}^t = \sigma(W_r c^t_{i,t} + W_l h^t_i),$$

Finally, we get the generation probability distribution $p_\theta = \{p_{\text{gen}}, \cdots, p_{\text{gen}}\} \in \mathbb{R}^{L \times V}$, which is summarized by accumulating the probability distribution of the fixed vocabulary $p_{\text{voc}} = \{p_{\text{voc}}^1, \cdots, p_{\text{voc}}^L\}$ and the dynamic words in retrieved sentences $p_{\text{ret}} \in \mathbb{R}^{\text{topk} \times V}$ conditioned on copy probability $p_{\text{copy}} \in \mathbb{R}^{\text{topk} \times 1}$ via broadcasting. Note that $p_{\text{ret}}$ has been extended to the same size of fixed vocabulary.

$$p_\theta = (1 - p_{\text{copy}}) p_{\text{voc}} + p_{\text{copy}} p_{\text{ret}}.$$  

3.3. Training

Review the definition of the open-book video captioning, the final probability of target word is jointly predicted by the similarities $p_\eta$ of the retrieved sentences and the generation probabilities $p_\theta$ with copy-mechanism by substituting Eq.8 and Eq.15 into Eq.1. Our goal is to minimize the negative log-likelihood of each target word $y_i$:

$$\mathcal{L}_{\text{gen}} = - \sum_{t=1}^{T} \log \sum_{i=1}^{\text{topk}} p_\eta(z_i)p_\theta,i(y_i).$$

These two components can be trained separately. Assuming that given an off-the-shelf retriever, our model can directly use the retrieval results for generation. In this case, we keep the retriever fixed, only fine-tuning the generator. This provides convenience for replacing better retrievers or adapting to different datasets.

Additionally, the retriever and generator can be jointly trained end-to-end in an iterative manner for better performance. However, updating the retriever directly during training may decrease its performance drastically as the generator has not been well trained to begin with. For a stable training, we add the ranking loss mentioned in Sec3.2 to the generation loss as a constraint $i.e.$, $\mathcal{L} = \mathcal{L}_{\text{ret}} + \mathcal{L}_{\text{gen}}$. Moreover, we periodically (per epoch in our work) perform the retrieval process because it is costly and frequently changing the retrieval results will confuse the generator.

4. Experiments

4.1. Experimental Settings

Datasets. We carry out all the experiments on MSR-VTT and recent VATEX, which are two large-scale video-caption datasets. The MSR-VTT [31] contains 10,000 open-domain video clips from YouTube website. For each clip, there are 20 human descriptions and one of 20 categories (music, sports, etc.). We follow the standard splits with 6,573 videos for training, 497 videos for validation, and 2,990 videos for testing. The most recent VATEX [29] dataset reuses the videos from Kinetics-600 that contains 41,269 video clips with 10 English text sentences. According to the official splits, the dataset is divided into 25,991 training, 3,000 validation, and 6,000 public testing.

Due to the strict quality control, VATEX has enricher, longer (average 16 tokens v.s. 9 tokens in MSR-VTT) and more accurate annotations. Therefore, we conduct and report most of the experiments on this dataset. Other experiments can be seen in the supplementary materials.

Evaluation Metrics. We use standard captioning metrics, i.e., BLEU-4, Meteor, Rouge-L, and CIDEr, to evaluate the performance of video captioning. We pay more attention to CIDEr during experiments, since only CIDEr weights the n-grams that relevant to the video content, which can better reflect the capability on producing novel expressions. Moreover, we introduce metrics in information retrieval, including Recall at K (R@K), Median Rank (MedR), and Mean Rank (MnR), to measure the performance of the video-text retrieval. R@K measures the proportion of correct targets retrieved from K samples. MedR and MnR represent the median and average rank of correct targets in the retrieved ranking list separately.

Implementation Details. For the extraction of visual features, we use C3D pre-trained on Kinetics-400 and InceptionResNetV2 pre-trained on ImageNet to extract motion and appearance features, respectively. We also conduct extra experiments for VATEX with other features, i.e., the I3D motion features pre-trained on Kinetics-600 and ResNet152 appearance features pre-trained on ImageNet. All the features above are extracted from 28 key-frames/segments of video sampling at equal intervals. For
the textual embedding, the sentences longer than 40 words are truncated and initialized the word embedding with GloVe by spaCy toolkit.

For the setting of retriever, the joint embedding size of Bi-encoders is 1024. We set the margin \( \Delta \) as 0.2, the batch size as 128 and the learning rate as \( 2e^{-4} \). The retriever converges in around 10 epochs, and the best model is selected from the best results on the validation. Note that during the training and testing phase of RCG, the sentences are retrieved only from the corpus of training set. Furthermore, the sentences belonging to their own video should be excluded. Otherwise, the answer will be leaked, and the training will be destroyed.

For the setting of generator, we keep the features in sync with the retriever. The hidden size of the hierarchical-LSTMs is 1024, and the state size of all the attention modules is 512. The model is optimized by Adam. The learning rate is initialized with \( 2e^{-4} \) and decayed 0.5 times every 3 epochs. The batch size is set to 64, and the training of the model can be converged with no more than 20 epochs. During the validation, the beam-search (beam size is 3) is used for a better generation.

### 4.2. Quantitative Analysis

The core of proposed open-book video captioning is to assist the caption generation by introducing the cross-modal retrieval. We elaborate the following Q&As to better illustrate the impact open-book captioning and prove the effectiveness of our model. Unless otherwise specified, all the experiments are carried out on VATEX.

#### Does the performance of the retriever affect the results?

We leverage different features to train the retrievers and simulate retrievers with different capabilities. We report the performance of the video-text retrieval and corresponding generation using different retrievers in Table 1. In lines 3 to 5, we see that the retriever with better performance can significantly improve the generation. An intuitive explanation is that a good retriever can find sentences closer to the video content and provide better expressions. Moreover, we find the results are similar between the model without retriever in line 1 and the model with a randomly initialized retriever as the worst retriever in line 2. In the worst case, the generator will not rely on the retrieved sentences reflecting the robustness of our model.

#### Does the number of retrieved sentences affect the results?

We analyze the effect of using different numbers of retrieved sentences in training and testing phases. In training phase, we explore 1 to 10 sentences for training, and 10 sentences are used for testing. As illustrated in Table 2, we find that a moderate number of retrieved sentences (3 for VATEX) are helpful for generation during training. This is because the retrieved sentence does not exactly correspond to the video, and the noise is also introduced with useful information. In testing phase, we select a well-trained model with fixed retriever trained on 3 sentences and test with various 1 to 30 sentences retrieved from training set as hints. Compared with the results in Table 3, lines 4 to 6, it demonstrates that more sentences may bring more hints for generation until saturation. In summary, too many retrieved sentences with noise are not conducive to model training, and a trained model can adaptively select useful cues from multiple sentences.

#### Does the quality of the retrieval corpus affect the results?

We conduct this experiment by randomly selecting different proportions of sentences in training set to simulate retrieval corpora of different quality. In Table 4, lines 1 to 4, it illustrates large scale retrieval corpus is conducive to producing better generation, which may be because the

---

**Table 1. Performance of video-text retrieval and video captioning tasks using fixed retrievers trained by different features.**

<table>
<thead>
<tr>
<th># Retrievals Training</th>
<th>Video-Text Retrieval</th>
<th>Video Captioning</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R@1</td>
<td>R@5</td>
</tr>
<tr>
<td>1 w/o Retriever</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>2 Random</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td>3 ResNet</td>
<td>17.2</td>
<td>37.7</td>
</tr>
<tr>
<td>4 I3D</td>
<td>24.9</td>
<td>51.0</td>
</tr>
<tr>
<td>5 ResNet+I3D</td>
<td>29.4</td>
<td>56.9</td>
</tr>
</tbody>
</table>

**Table 2. Performance of training the model with different numbers of retrieved sentences. The model is tested via top-10 sentences.**

<table>
<thead>
<tr>
<th># Retrievals Training</th>
<th>Fixed</th>
<th>CIDEr</th>
<th>BLEU-4</th>
<th>Rouge-L</th>
<th>Meteor</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>✓</td>
<td>55.3</td>
<td>33.4</td>
<td>50.0</td>
<td>23.5</td>
</tr>
<tr>
<td>3</td>
<td>×</td>
<td>56.3</td>
<td>33.4</td>
<td>50.0</td>
<td>23.6</td>
</tr>
<tr>
<td>5</td>
<td>✓</td>
<td>57.5</td>
<td>33.9</td>
<td>50.2</td>
<td>23.7</td>
</tr>
<tr>
<td>7</td>
<td>×</td>
<td>56.8</td>
<td>33.9</td>
<td>50.2</td>
<td>23.6</td>
</tr>
<tr>
<td>10</td>
<td>✓</td>
<td>56.4</td>
<td>33.6</td>
<td>50.0</td>
<td>23.5</td>
</tr>
<tr>
<td>10</td>
<td>×</td>
<td>57.1</td>
<td>33.7</td>
<td>50.1</td>
<td>23.6</td>
</tr>
</tbody>
</table>

**Table 3. Performance of testing the model with different numbers of retrieved sentences. The model is trained via top-3 retrieved sentences.**

<table>
<thead>
<tr>
<th># Retrievals Testing</th>
<th>CIDEr</th>
<th>BLEU-4</th>
<th>Rouge-L</th>
<th>Meteor</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>50.5</td>
<td>30.9</td>
<td>48.5</td>
<td>22.5</td>
</tr>
<tr>
<td>2</td>
<td>54.1</td>
<td>32.6</td>
<td>49.6</td>
<td>23.3</td>
</tr>
<tr>
<td>3</td>
<td>55.8</td>
<td>33.0</td>
<td>50.0</td>
<td>23.5</td>
</tr>
<tr>
<td>4</td>
<td>56.8</td>
<td>33.4</td>
<td>50.1</td>
<td>23.6</td>
</tr>
<tr>
<td>5</td>
<td>57.4</td>
<td>33.5</td>
<td>50.2</td>
<td>23.7</td>
</tr>
<tr>
<td>6</td>
<td>57.5</td>
<td>33.5</td>
<td>50.1</td>
<td>23.7</td>
</tr>
</tbody>
</table>

**Table 4. Performance of using different qualities of the corpus for testing. Different qualities are simulated through the corpus size.**

<table>
<thead>
<tr>
<th>Retrieval Corpus</th>
<th>Corpus Size</th>
<th>CIDEr</th>
<th>BLEU-4</th>
<th>Rouge-L</th>
<th>Meteor</th>
</tr>
</thead>
<tbody>
<tr>
<td>TrainSet</td>
<td>+0.1%</td>
<td>39.9</td>
<td>29.4</td>
<td>47.7</td>
<td>21.9</td>
</tr>
<tr>
<td>TrainSet</td>
<td>+1%</td>
<td>48.9</td>
<td>31.4</td>
<td>49.0</td>
<td>22.9</td>
</tr>
<tr>
<td>TrainSet</td>
<td>+10%</td>
<td>56.0</td>
<td>33.2</td>
<td>49.9</td>
<td>23.5</td>
</tr>
<tr>
<td>TrainSet</td>
<td>+100%</td>
<td>56.8</td>
<td>33.4</td>
<td>50.1</td>
<td>23.6</td>
</tr>
<tr>
<td>TestSet +Oracle</td>
<td>58.9</td>
<td>34.3</td>
<td>50.5</td>
<td>23.9</td>
<td></td>
</tr>
</tbody>
</table>
higher quality corpus contains more semantically similar sentences, and it provides more hints related to the video content for generation. Furthermore, assuming that our retrieval corpus is good enough to contain sentences that correctly describe the video. It can be seen in Tab. 4 line 5, a significant improvement than ever before if we combine training set and test set as the Oracle corpus for testing. Note that the Oracle corpus is only intended to show that our model can retrieve better sentences for generation and is not involved in the training process.

**Which is better, fixed or jointly trained retriever?** We choose CIDEr as the metric of caption performance since it reflects the generation associated with video content. We plot the real-time test scores of fixed retriever with different sentences, jointly trained retriever and without retriever during training. As shown in Fig. 4, the accuracy is significantly improved, and the model converges faster after introducing our retriever. The jointly trained retriever is also better than fixed in the term of accuracy. The same conclusion can be drawn from Tab. 2. This may be because joint trained retriever can increase the probability of sentences that are more helpful for generation.

For efficiency, we profile the model speed on a server with CPU E5-2650 v4 @ 2.20GHz, 128 GB memory and GTX1080Ti. All tests are performed using a single GPU. The retrieval speed is about 254 videos/sec, which is the same for both retrievers. The fixed retriever’s training speed is 137.6 videos/sec, which is slightly faster than the co-trained retriever of 127.2 videos/sec by 8.2%.

**How is the generalization of the model for cross-dataset videos?** In practical applications, the input video distribution is not necessarily the same as that of training data. This puts forward requirements for the generalization of the model. For this experiment, we pre-train models on VATEX and measure the performances on MSRVTT that is unseen in training. As shown in Tab. 5, the performance of our RCG in line 3 is better than the baseline generation model in line 1. The comparison to line 3, 5 shows that higher quality of the retrieval corpus leads to better performance. Furthermore, we select a retriever trained on MSRVTT, and the comparison to line 6 shows a better retriever can further improve performance. The above experiments also show that our RCG can be extended by changing different retriever and retrieval corpus.

### 4.3. Comparison to State-of-the-Arts
After exploring of our proposed model, we compare it with models published on the most recent conferences. For a fair comparison, we use the standard motion and appearance feature extractions used by most models. Reinforce-
more consistent with our motivation and practical industrial.

4.4. Qualitative Analysis
We visualize the heatmap of the words copied from the retrieved sentences and their probabilities during the generation process, as illustrated in Fig.3. According to the heatmap, whether the words come from the retrieved sentences and each sentence’s contribution can be seen intuitively. In Fig.5, we also visualize the attention weights of retrieved sentences during the process of video-text retrieval. Most of the weights are focused on keywords, e.g., “scuba diver”, “ocean water” and “masks”, which proves the effectiveness of our retriever. Comparing the baseline model, the proposed RCG can acquire more diverse expressions, e.g., “breathing out of their masks” benefit from the coping mechanism, and correct the expressions from “a game of frisbee” to “playing catch” under the guidance of retrieved sentences.

5. Conclusion
In this paper, we have presented the RCG for open-book video captioning. RCG efficiently retrieves video-content-relevant sentences from text corpus through a cross-modal retriever, jointly copies cues from multi-retrieved sentences and generates through a copy-mechanism caption generation. The improved performance proves the effectiveness of our method. Moreover, it outperforms ORG-TRL model even without fine-grained object features and external knowledge, which obtains 3.9% and 15.7% relative gains on CIDEr metric for MSR-VTT and VATEX. We have a comparable performance for the other metrics of MSR-VTT and have achieved the best results in VATEX. In addition, compared to MSRVTT, the improvement in VATEX is particularly obvious. It may be because the videos in VATEX are collected by categories and have a large number of videos with the same semantics. This is

Table 6. Performance comparison with state-of-the-art methods on MSR-VTT and VATEX testing set. +RL denotes reinforcement learning, +Audio means introducing audio features.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Method</th>
<th>Ref.</th>
<th>CIDEr</th>
<th>BLEU-4</th>
<th>Rouge-L</th>
<th>Meteor</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSR-VTT</td>
<td>POS-CGI[29]</td>
<td>ICCV19</td>
<td>48.7</td>
<td>42.0</td>
<td>61.6</td>
<td>28.2</td>
</tr>
<tr>
<td></td>
<td>POS-VCT[33]</td>
<td>ICCV19</td>
<td>49.1</td>
<td>42.3</td>
<td>62.8</td>
<td>29.7</td>
</tr>
<tr>
<td></td>
<td>SAAIT[37]</td>
<td>CVPR20</td>
<td>49.1</td>
<td>40.5</td>
<td>60.9</td>
<td>28.2</td>
</tr>
<tr>
<td></td>
<td>+RL</td>
<td></td>
<td>51.0</td>
<td>39.9</td>
<td>61.2</td>
<td>27.7</td>
</tr>
<tr>
<td></td>
<td>STG-KD[22]</td>
<td>CVPR20</td>
<td>47.1</td>
<td>40.5</td>
<td>60.9</td>
<td>28.3</td>
</tr>
<tr>
<td></td>
<td>+Audio</td>
<td></td>
<td>50.6</td>
<td>43.9</td>
<td>-</td>
<td>29.5</td>
</tr>
<tr>
<td></td>
<td>ORG-TRL[36]</td>
<td>CVPR20</td>
<td>50.9</td>
<td>43.6</td>
<td>62.1</td>
<td>28.8</td>
</tr>
<tr>
<td>Baseline</td>
<td>Ours</td>
<td></td>
<td>49.8</td>
<td>42.2</td>
<td>61.2</td>
<td>28.2</td>
</tr>
<tr>
<td>+FixRet</td>
<td>Ours</td>
<td></td>
<td>52.3</td>
<td>43.1</td>
<td>61.9</td>
<td>29.0</td>
</tr>
<tr>
<td>+TrainRet</td>
<td>Ours</td>
<td></td>
<td>52.9</td>
<td>42.8</td>
<td>61.7</td>
<td>29.3</td>
</tr>
</tbody>
</table>

| VATEX  | VATEX[29]   | ICCV19| 45.6  | 28.7   | 47.2    | 21.9   |
|        | ORG-TRL[36] | CVPR20| 49.7  | 32.1   | 48.9    | 22.2   |
| Baseline| Ours       |       | 49.2  | 31.3   | 48.5    | 21.9   |
| +FixRet | Ours       |       | 56.8  | 33.4   | 50.1    | 23.6   |
| +TrainRet| Ours     |       | 57.5  | 33.9   | 50.2    | 23.7   |

Acknowledgment This work is supported by the National Key R&D Plan (No.2018AAA0102802, 2018AAA0102803, 2018AAA0102800, 2018YFC0823003, 2017YFB1002801, 2020AAA106800), Natural Science Foundation of China (No.62036011, 61721004, 61772225, 61972397), National Natural Science Foundation of China (No.62036011, 61721004, 61772225, 61972397), Natural Science Foundation of Guangdong (No.2018B030315046), Beijing Natural Science Foundation (No.L1820058).
References


