O2NA: An Object-Oriented Non-Autoregressive Approach for Controllable Video Captioning (Short)

Fenglin Liu[1]*, Xuancheng Ren[2]†, Xian Wu[4], Bang Yang[1], Shen Ge[4], Yuexian Zou[1], Xu Sun[2,3]
[1]ADSPLAB, School of ECE, Peking University
[2]MOE Key Laboratory of Computational Linguistics, School of EECS, Peking University
[3]Center for Data Science, Peking University
[4]Tencent, Beijing, China
{fenglinliu98, renxc, yb.ece, zouyx, xusun}@pku.edu.cn
{kevinxwu, shenge}@tencent.com

Abstract

Video captioning combines video understanding and language generation. Different from image captioning that describes a static image with details of almost *every* object, video captioning usually considers a sequence of frames and biases towards *focused* objects, e.g., the objects that stay in focus regardless of the changing background. Therefore, detecting and properly accommodating focused objects is critical in video captioning. To enforce the description of focused objects and achieve controllable video captioning, we propose an Object-Oriented Non-Autoregressive approach (O2NA), which performs caption generation in three steps: 1) identify the focused objects and predict their locations in the target caption; 2) generate the related attribute words and relation words of these focused objects to form a draft caption; and 3) combine video information to refine the draft caption to a fluent final caption. Since the focused objects are generated and located ahead of other words, it is difficult to apply the word-by-word autoregressive generation process; instead, we adopt a non-autoregressive approach. The experiments on two benchmark datasets, i.e., MSR-VTT and MSVD, demonstrate the effectiveness of O2NA, which achieves results competitive with the state-of-the-arts but with both higher diversity and higher inference speed.

1 Introduction

The task of video captioning, which aims to generate a descriptive sentence based on the input video, has a wide range of applications. In recent years, deep neural models, particularly the models based on the encoder-decoder framework (Venugopalan et al., 2015; Pan et al., 2016; Xu et al., 2017; Aafaq et al., 2019; Liu et al., 2018, 2019b, 2020), have achieved great success in advancing the state-of-the-art (Pan et al., 2020; Zheng et al., 2020; Perez-Martin et al., 2021; Yang et al., 2019). These models usually entail the autoregressive property, i.e., conditioning each word on the previously generated words.

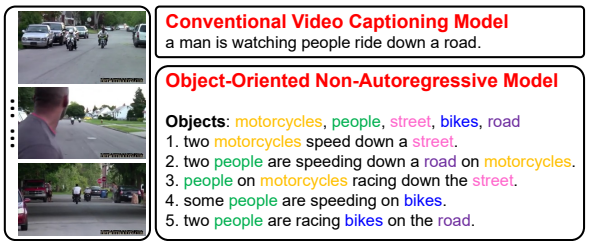

Figure 1: Examples of the captions generated by a state-of-the-art conventional video captioning model (Zheng et al., 2020) and our model. Compared to the conventional model, whose generation process is hardly controllable, our model can be guided to mention the desired objects (i.e., the colored objects) and generate diverse, object-oriented captions for a video.

In video captioning, one critical step is to detect and include focused objects. As exemplified in Figure 1, when a dangerous situation occurs, a captioning-based blind-aid system should focus on the dangerous objects on the road to alert the visually-impaired people, rather than over-describe the presence of pedestrians or shops nearby. It means that in the above example, *speeding vehicles* should be considered as focused objects and should be mentioned in the generated caption. While people could identify focused objects in video easily (Shinn-Cunningham, 2008; Corbetta and Shulman, 2002; Posner and Petersen, 1990), existing captioning systems can hardly be controlled to generate focused objects because of their word-to-word generation practice. Motivated by those observations, we introduce the problem of controllable video captioning in the sense of controlling contents.

As shown in Figure 2, to solve the controllable

*Equal Contributions.

video captioning problem, we propose the Object-Oriented Non-Autoregressive approach (O2NA). Different from conventional models that adopt a left-to-right or word-by-word decoding process, O2NA applies a non-autoregressive manner to control the caption generation. O2NA first detects all objects that appear in the video and then selects the focused objects for the final caption. For example, in the aforementioned blind-aid system, the system would select the dangerous objects *speeding vehicles* in case of an emergency. Next, the caption generation process consists of three main steps: 1) locate all focused objects in the proper locations of the target caption; 2) generate the related attribute words and relation words to form a draft caption; and 3) adopt the iterative refinement approach (Ghazvininejad et al., 2019; Lee et al., 2018) to proofread and improve the draft caption.

For each step, as there is no dependency among generated words, the words can be generated in parallel, indicating a fixed computing time regardless of caption length, while computing time of the conventional autoregressive approach is linear with the caption length. For long captions, conventional methods embody high inference latency, which limits their adoption in real-time applications, e.g., blind-aid system (Voykinska et al., 2016) and human-robot interaction (Das et al., 2017). According to our experiments on two benchmark datasets, i.e., MSR-VTT (Xu et al., 2016) and MSVD (a.k.a. Youtube2Text) (Guadarrama et al., 2013), our O2NA is able to produce a descriptive and fluent caption which outperforms several existing methods in terms of both accuracy and efficiency.

Overall, the main contributions of this paper are:

- We introduce the problem of controllable video captioning in the sense of controlled contents, which has more practical values than the existing studies on syntactic variations.
- Specifically, we propose the Object-Oriented Non-Autoregressive approach (O2NA) to tackle the controllable video captioning problem by injecting strong control signals conditioned on selected objects, with the benefits of fast and fixed inference time, which are critical for real-time applications.
- We evaluate our approach on two datasets. In particular, our O2NA achieves competitive results with the state-of-the-art methods with higher diversity and higher inference speed.

2 Approach

We first briefly introduce the backgrounds of our approach and then describe the approach in detail.

2.1 Backgrounds

The backgrounds are introduced from the used Video Representations and Basic Module.

Video Representations For video captioning, image and motion features have been widely used. Image features are good at illustrating the shapes, the colors and the relationships of the items in the image; Motion features are important for capturing the actions and temporal interactions. Following Pei et al. (2019), given a video, $N = 8$ key frames are uniformly sampled to extract image features I. Considering both the past and the future contexts, we take each key frame as the center to generate corresponding motion features M. Specifically, for the image features, we adopt the ResNet-101 (He et al., 2016) pre-trained on ImageNet (Deng et al., 2009) to extract the 2048-D image features $I \in \mathbb{R}^{N \times d_i}$ ($d_i = 2048$), which are the output of the last convolutional layer. The motion features are usually given by the 3D CNN (Tran et al., 2015), we adopt the ResNeXt-101 (Hara et al., 2018) pre-trained on the Kinetics dataset (Kay et al., 2017) to extract the 2048-D motion features $M \in \mathbb{R}^{N \times d_m}$ ($d_m = 2048$). In this paper, both features are projected to $d_h = 512$. Then, we use the concatenation of the two projected features as the video representations $V \in \mathbb{R}^{2N \times d_h}$ to our model.

Basic Module Our approach is adapted from the non-autoregressive decoding models (Lee et al., 2018; Ghazvininejad et al., 2019), which is based on the Transformer decoder (TFM) (Vaswani et al., 2017). Specifically, the TFM consists of a self-attention, a source-attention and a feed-forward network (FF). The multi-head attention (MHA) is the basic of self-attention and source-attention. Overall, the TFM is defined as follows:

$$\text{TFM}(Q, K, V) = \text{FF}(\text{MHA}(\text{MHA}(Q, Q, Q), K, V)). \quad (1)$$

Please refer to Vaswani et al. (2017) for the detailed introduction of the Transformer decoder (TFM).

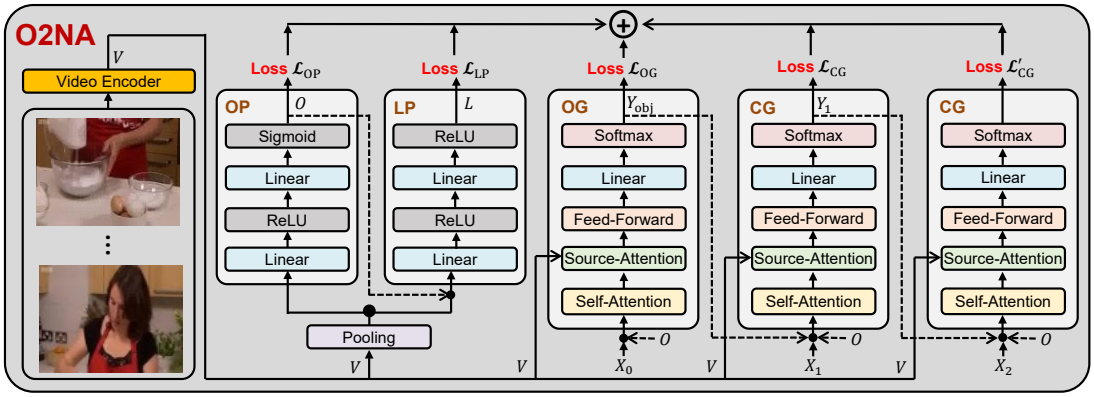

Figure 2: Illustration of our proposed O2NA, which consists of an object predictor (OP), a length predictor (LP), an object generator (OG) and a caption generator (CG). The object predictor and length predictor extract the objects appearing to the input video and estimate the length of target caption, respectively; The object generator locates all the focused objects we care about in the target caption; The caption generator generates the rest words to link focused objects to form a fluent caption. It is worth noting that the focused objects could be the objects predicted by the object predictor, the preferred objects given by the user or the pre-defined concerned objects, e.g., the dangerous objects in the captioning-based blind-aid system.

2.2 Object-Oriented Non-Autoregressive Approach (O2NA)

As stated above, we adopt the Transformer decoder (Vaswani et al., 2017) to implement our Object-Oriented Non-Autoregressive approach (O2NA). Specifically, as shown in Figure 2, O2NA consists of an object predictor, a length predictor and two Transformer decoders, where the first decoder focuses on generating all the objects we care about in parallel (i.e., object generator), and the second decoder pays attention to linking these objects to form a fluent caption (i.e., caption generator).

Object Predictor (OP) The OP is expected to predict the objects that appear in the given video. We first build an object vocabulary based on the training captions. Given this object vocabulary, we can associate each video with a set of objects according to its human-annotated captions. Specifically, we denote the ground truth objects as $O^* = \{o_1^*, o_2^*, \ldots, o_M^*\}$, where M represents the size of object vocabulary; $o_i^* = 1$ if the video is annotated with object i, and $o_i^* = 0$ otherwise. During the training phase, we directly use the ground truth objects O^*. At the inference stage, we adopt a two-layer non-linear layer to predict the objects $O \in \mathbb{R}^M$, defined as:

$$\begin{aligned} O &= \text{Object-Predictor}(V) \\ &= \sigma\left(\text{ReLU}\left(\text{MP}\left(V\right) W_{\text{O}_1}\right) W_{\text{O}_2}\right) \\ \text{where } \text{MP}\left(V\right) &= \frac{1}{2N}\sum\nolimits_{i=1}^{2N} v_i, \end{aligned} \tag{2}$$

where MP denotes the Mean Pooling, σ is the sigmoid function; $W_{\text{O}_1} \in \mathbb{R}^{d_h \times d_h}$ and $W_{\text{O}_2} \in \mathbb{R}^{d_h \times M}$ are the parameters to be learned. Next, following Wu et al. (2016), we minimize the element-wise logistic loss function $\mathcal{L}_{\text{OP}}$ to train our OP:

$$\mathcal{L}_{\text{OP}} = \sum\nolimits_{i=1}^{M} \log\left(1 + \exp\left(-o_i^* o_i\right)\right). \tag{3}$$

During the inference procedure, to select the final predicted objects, we set a threshold γ, which means that if the $o_i > \gamma$, we reset $o_i = 1$, and reset $o_i = 0$ otherwise. In particular, if we care about some specific objects, for example, the user preferred objects or the pre-defined dangerous objects in the captioning-based blind-aid system, we could just set the value of these concerned objects equal to 1, and set the value of other objects equal to 0.

Length Predictor (LP) In the generation process, the non-autoregressive decoding model needs to know the length of target captions (Ghazvininejad et al., 2019). To this end, at training time, we use the sequence length l^* of ground truth caption. At inference stage, given the video information $V \in \mathbb{R}^{2N \times d_h}$ and the focused objects $O \in \mathbb{R}^M$, we adopt a LP to predict the length l. In detail, we apply a two-layer network to achieve the effect:

$$\begin{aligned} l \sim p_l &= \text{Length-Predictor}(V, O) \\ &= \text{softmax}\left(\text{ReLU}\left(\left[\text{MP}(V) W_{\text{L}_V}; O W_{\text{L}_O}\right]\right) W_{\text{L}}\right), \end{aligned} \tag{4}$$

where $[\cdot;\cdot]$ represents the concatenation operation; $W_{\text{L}_V} \in \mathbb{R}^{d_h \times d_h}$, $W_{\text{L}_O} \in \mathbb{R}^{M \times d_h}$ and $W_{\text{L}} \in \mathbb{R}^{2d_h \times l_{max}}$ are learnable parameters; $l_{max} = 30$ denotes the pre-defined maximum sequence length. Thus, $p_l \in \mathbb{R}^{l_{max}}$ is a probability. We adopt the

cross entropy loss $\mathcal{L}_{\text{LP}}$ to train the LP, which can be defined as follows:

$$\mathcal{L}_{\text{LP}} = -\log(p_l(l^*|V, O^*)). \tag{5}$$

Object Generator (OG) The object generator is based on the non-autoregressive decoder and is dedicated to generating all the objects we care about at once. To achieve such effect, we adopt a single-layer Transformer decoder[1], followed by a linear layer and a softmax function. In implementation, the object generator takes the fully masked sequence $X_0 = (x_{\text{m}_1}, x_{\text{m}_2}, \ldots, x_{\text{m}_L}), x_{\text{m}_i} \in \mathbb{R}^{d_h}$ with predicted length l by length predictor as input. The $x_{\text{m}_i} = w_{\text{[MASK]}} + e_i$, where $w_{\text{[MASK]}}$ and e_i denotes the word embedding of [MASK] token and position embedding, respectively. Then the object information O is added to X_0, i.e., $x'_{\text{m}_i} = x_{\text{m}_i} + OW_O$, where $W_O \in \mathbb{R}^{M \times d_h}$. At last, the transformer decoder in the object generator takes the $X_0 \oplus OW_O$ as input ($\oplus$ denotes the matrix-vector addition), and generates all objects at the position in the final caption, i.e., an object-oriented coarse-grained caption, which can be defined as follows:

$$\begin{aligned} Y_{\text{obj}} \sim p_0 &= \text{Object-Generator}(X_0, V, O) \\ &= \text{softmax}(\text{TFM}(X_0 \oplus OW_O, V, V)W_{\text{OG}}), \end{aligned} \tag{6}$$

where $X_0 \in \mathbb{R}^{l \times d_h}$, $V \in \mathbb{R}^{2N \times d_h}$, $O \in \mathbb{R}^M$ represent the input sequence, the video representations and the predicted objects, respectively; $W_O \in \mathbb{R}^{M \times d_h}$ and $W_{\text{OG}} \in \mathbb{R}^{d_h \times |D|}$ are the matrices for linear transformation; $|D|$ is the size of vocabulary D. Each value of $p_0 \in \mathbb{R}^{l \times |D|}$ is a probability indicating how likely each word in D should be the current output word.

At training time, for each human-annotated caption, we mask all the non-object words based on the object vocabulary to acquire the ground truth object sequence $Y^*_{\text{obj}} = (\ldots, \text{[MASK]}, \ldots, \text{object}_i, \ldots)$. Our goal is to minimize the following standard cross entropy loss:

$$\mathcal{L}_{\text{OG}} = -\sum_{i=1}^{l^*} \log(p_0(y^*_{\text{obj}_i}|X_0, V, O^*)). \tag{7}$$

Caption Generator (CG) In implementation, the caption generator shares the same structure with object generator. The main differences between the two generators are the different generating objective and the input sequence. Specifically, the caption generator takes the object sequence X_1 as input, where X_1 equals to Y^*_{obj} and Y_{obj} at the training stage and inference stage, respectively, and generates the related attribute words and relation words to form a draft caption, which is defined as:

$$\begin{aligned} Y_1 \sim p_1 &= \text{Caption-Generator}(X_1, V, O) \\ &= \text{softmax}(\text{TFM}(X_1 \oplus OW'_O, V, V)W_{\text{CG}}), \end{aligned} \tag{8}$$

where $p_1 \in \mathbb{R}^{l \times |D|}$. Given the ground truth caption $Y^*_{\text{cap}} = (y^*_{\text{cap}_1}, y^*_{\text{cap}_2}, \ldots, y^*_{\text{cap}_l})$, we adopt standard cross entropy loss as the loss function to train the CG, which can be defined as follows:

$$\mathcal{L}_{\text{CG}} = -\sum_{i=1}^{l^*} \log(p_1(y^*_{\text{cap}_i}|X_1, V, O^*)). \tag{9}$$

Since the non-autoregressive approach removes the sequential dependency, we may have introduced the "multi-modality problem" (Gu et al., 2018) (i.e., a word could appear in multiple position to form different captions). So we further adopt the iterative refinement approach (Lee et al., 2018) to proofread Y_1. In implementation, to acquire the input sequence X_2, we randomly mask $n = \lfloor l * r \rfloor$ words in Y^*_{cap} and mask out top n words with the lowest confidence in Y_1 at the training time and inference time, respectively, where l and r represent the caption length and masking ratio, respectively, and the confidence is taken to be the output probability. To obtain the final caption, we employ the following equation, which is defined as:

$$Y_2 \sim p_2 = \text{Caption-Generator}(X_2, V, O). \tag{10}$$

Finally, the cross entropy loss is defined similar as Eq. (9):

$$\mathcal{L}'_{\text{CG}} = -\sum_{i=1}^{l^*} \log(p_2(y^*_{\text{cap}_i}|X_2, V, O^*)). \tag{11}$$

Overall, by combining the $\mathcal{L}_{\text{OP}}$ in Eq. (3), $\mathcal{L}_{\text{LP}}$ in Eq. (5), $\mathcal{L}_{\text{OG}}$ in Eq. (7), $\mathcal{L}_{\text{CG}}$ in Eq. (9) and $\mathcal{L}'_{\text{CG}}$ in Eq. (11), the full training objective is:

$$\mathcal{L}_{\text{full}} = \lambda_1\mathcal{L}_{\text{LP}} + \lambda_2\mathcal{L}_{\text{OP}} + \lambda_3\mathcal{L}_{\text{OG}} + \lambda_4\mathcal{L}_{\text{CG}} + \lambda_5\mathcal{L}'_{\text{CG}}, \tag{12}$$

where $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ and λ_5 are the hyperparameters that control the regularization. For simplicity, we set $\lambda_1 = \lambda_2 = \lambda_3 = \lambda_4 = \lambda_5 = 1$, since we find that our approach can achieve competitive results with the state-of-the-art models in major metrics under this setting, thus we do not attempt to explore other settings.

Overall, through Eq. (12), we are able to realize our Object-Oriented Non-Autoregressive approach (O2NA). The trained model is encouraged to describe the focused objects that a user cares about.

[1] Our experiments showed that using a single-layer Transformer decoder can achieve the best performance in major metrics with fastest inference speed.

Methods	Dataset: MSVD (Guadarrama et al., 2013)				Dataset: MSR-VTT (Xu et al., 2016)							
	BLEU-4	METEOR	ROUGE-L	CIDEr	BLEU-4	METEOR	ROUGE-L	CIDEr	Novel	Unique	Vocab	VPS
RecNet (Wang et al., 2018)	52.3	34.1	69.8	80.3	39.1	26.6	59.3	42.7	-	-	-	-
PickNet (Chen et al., 2018)	52.3	33.3	69.6	76.5	41.3	27.7	59.8	44.1	-	-	-	-
OA-BTG (Zhang and Peng, 2019)	56.9	36.2	-	90.6	41.4	28.2	-	46.9	-	-	-	-
MARN (Pei et al., 2019)	48.6	35.1	71.9	92.2	40.4	28.1	60.7	47.1	-	-	-	-
GRU-EVE (Aafaq et al., 2019)	47.9	35.0	71.5	78.1	38.3	28.4	60.7	48.1	-	-	-	-
POS-Control (Wang et al., 2019a)	52.5	34.1	71.3	88.7	42.0	28.2	61.6	48.7	-	-	-	-
STAT (Yan et al., 2020)	52.0	33.3	-	73.8	39.3	27.1	-	43.8	-	-	-	-
STGN-OAKD (Pan et al., 2020)	52.2	36.9	73.9	93.0	40.5	28.3	60.9	47.1	-	-	-	-
ORG-TRL (Zhang et al., 2020)	54.3	36.4	73.9	95.2	43.6	28.8	62.1	50.9	-	-	-	-
SAAT (Zheng et al., 2020)	46.5	33.5	69.4	81.0	39.9	27.7	61.2	51.0	26.8†	35.7†	3.9†	17.6†
SGN (Ryu et al., 2021)	52.8	35.5	72.9	94.3	40.8	28.3	60.8	49.5	-	-	-	-
SemSynAN (Perez-Martin et al., 2021)	64.4	41.9	79.5	111.5	46.4	30.4	64.7	51.9	-	-	-	-
O2NA (Ours)	55.4	37.4	74.5	96.4	41.6	28.5	62.4	51.1	37.2	46.7	4.6	70.8

Table 1: Performance of automatic evaluation on the test sets of MSVD and MSR-VTT. Higher is better in all columns. † denotes our own implementation. VPS stands for videos per second at the inference stage, which is measured on a single NVIDIA GeForce GTX 1080 Ti. In this paper, the Red- and the Blue- colored numbers denote the best and the second best results across all approaches, respectively. All existing video captioning systems follow the autoregressive approach to generate the captions and cannot control the video captioning process to ensure the inclusion of the focused objects. In comparison, O2NA can not only describe the focused objects, but also achieve competitive performances with the state-of-the-arts in major metrics with both higher diversity and faster inference.

3 Experiments[2]

We describe the main experimental results of our approach[3]. In comparable settings, twelve representative methods, including five most recently published state-of-the-art approaches, namely STAT (Yan et al., 2020), STGN-OAKD (Pan et al., 2020), ORG-TRL (Zhang et al., 2020), SAAT (Zheng et al., 2020), SGN (Ryu et al., 2021) and SemSynAN (Perez-Martin et al., 2021), are selected for comparison. Unless specifically stated, we directly report the results from the original papers. The results on the test of MSVD and MSR-VTT datasets are shown in Table 1. As we can see, our O2NA achieves the results competitive with the state-of-the-art models on the two datasets in major metrics. The competitive performances verify the validity of our O2NA for standard video captioning. More encouragingly, in terms of the metrics that evaluate the diversity of the generated captions, O2NA surpasses the previous state-of-the-art models with relatively 39%, 31% and 18% margins in terms of Novel, Unique and Vocab scores, which proves our arguments and corroborates the effectiveness of our approach. Moreover, since our O2NA generate the entire captions in three steps with a fixed generation time, we achieve the fastest inference speed (highest VPS in Table 1) among existing methods.

Overall, our O2NA achieves performances competitive with state-of-the-arts in major metrics but with higher diversity scores and faster inference speed. The experimental results show that our approach is able to generate fluent and diverse video captions with fast inference speed. More importantly, our O2NA allows an easy way to control the contents of video captions rather than merely syntactic variations in existing studies. These advantages of our approach could have the potential to promote the application of video captioning for real-time industrial applications.

4 Conclusions

In this work, we introduce the problem of controllable video captioning in the sense of controlled contents. In contrast to the existing studies considering syntactic variations, controlling contents is of more practical value. To tackle the problem, we propose the Object-Oriented Non-Autoregressive approach (O2NA), which encourages the model to describe the focused objects that a user cares about by generating captions conditioned on the focused objects non-autoregressively. The experiments verify the flexibility and prove the effectiveness of O2NA, which achieves competitive results with state-of-the-art models on two datasets in major metrics with higher diversity and faster inference.

Acknowledgments

We thank all the anonymous reviewers for their constructive suggestions. Xu Sun and Yuexian Zou are the corresponding authors of this paper (Short).

[2]For detailed introduction of the datasets, metrics and settings, please refer to the Appendix A.

[3]For detailed experiments and analyses of our approach, please refer to the Full version of our paper.

References

Nayyer Aafaq, Naveed Akhtar, Wei Liu, Syed Zulqarnain Gilani, and Ajmal Mian. 2019. Spatio-temporal dynamics and semantic attribute enriched visual encoding for video captioning. In *CVPR*.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *ACL*.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C. Lawrence Zitnick. 2015. Microsoft COCO captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*.

Yangyu Chen, Shuhui Wang, Weigang Zhang, and Qingming Huang. 2018. Less is more: Picking informative frames for video captioning. In *ECCV*.

Maurizio Corbetta and Gordon L Shulman. 2002. Control of goal-directed and stimulus-driven attention in the brain. *Nature reviews neuroscience*, 3(3):201–215.

Bo Dai, Sanja Fidler, and Dahua Lin. 2018. A neural compositional paradigm for image captioning. In *NeurIPS*.

Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José M. F. Moura, Devi Parikh, and Dhruv Batra. 2017. Visual dialog. In *CVPR*.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. 2009. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255. IEEE Computer Society.

Marjan Ghazvininejad, Omer Levy, Yinhan Liu, and Luke Zettlemoyer. 2019. Constant-time machine translation with conditional masked language models. *arXiv preprint arXiv:1904.09324*.

Jiatao Gu, James Bradbury, Caiming Xiong, Victor O. K. Li, and Richard Socher. 2018. Non-autoregressive neural machine translation. In *ICLR*.

Sergio Guadarrama, Niveda Krishnamoorthy, Girish Malkarnenkar, Subhashini Venugopalan, Raymond J. Mooney, Trevor Darrell, and Kate Saenko. 2013. Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In *ICCV*.

Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. 2018. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *CVPR*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*.

Will Kay, João Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. 2017. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*.

Yoon Kim and Alexander M. Rush. 2016. Sequence-level knowledge distillation. In *EMNLP*.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. In *ICLR*.

Jason Lee, Elman Mansimov, and Kyunghyun Cho. 2018. Deterministic non-autoregressive neural sequence modeling by iterative refinement. In *EMNLP*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *ACL*.

Chin-Yew Lin and Eduard H. Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *NAACL-HLT*.

Fenglin Liu, Yuanxin Liu, Xuancheng Ren, Xiaodong He, Kai Lei, and Xu Sun. 2019a. Aligning visual regions and textual concepts for semantic-grounded image representations. In *NeurIPS*.

Fenglin Liu, Xuancheng Ren, Yuanxin Liu, Kai Lei, and Xu Sun. 2019b. Exploring and distilling cross-modal information for image captioning. In *IJCAI*.

Fenglin Liu, Xuancheng Ren, Yuanxin Liu, Houfeng Wang, and Xu Sun. 2018. simnet: Stepwise image-topic merging network for generating detailed and comprehensive image captions. In *EMNLP*.

Fenglin Liu, Xian Wu, Shen Ge, Xiaoyu Zhang, Wei Fan, and Yuexian Zou. 2020. Bridging the gap between vision and language domains for improved image captioning. In *ACM Multimedia*.

Boxiao Pan, Haoye Cai, De-An Huang, Kuan-Hui Lee, Adrien Gaidon, Ehsan Adeli, and Juan Carlos Niebles. 2020. Spatio-temporal graph for video captioning with knowledge distillation. In *CVPR*.

Yingwei Pan, Tao Mei, Ting Yao, Houqiang Li, and Yong Rui. 2016. Jointly modeling embedding and translation to bridge video and language. In *CVPR*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for automatic evaluation of machine translation. In *ACL*.

Wenjie Pei, Jiyuan Zhang, Xiangrong Wang, Lei Ke, Xiaoyong Shen, and Yu-Wing Tai. 2019. Memory-attended recurrent network for video captioning. In *CVPR*.

Jesus Perez-Martin, Benjamin Bustos, and Jorge Perez. 2021. Improving video captioning with temporal composition of a visual-syntactic embedding. In *WACV*.

Michael I Posner and Steven E Petersen. 1990. The attention system of the human brain. *Annual review of neuroscience*, 13(1):25–42.

Hobin Ryu, Sunghun Kang, Haeyong Kang, and Chang D. Yoo. 2021. Semantic grouping network for video captioning. In *AAAI*.

Barbara G Shinn-Cunningham. 2008. Object-based auditory and visual attention. *Trends in cognitive sciences*, 12(5):182–186.

Du Tran, Lubomir D. Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. 2015. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*.

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *CVPR*.

Subhashini Venugopalan, Huijuan Xu, Jeff Donahue, Marcus Rohrbach, Raymond J. Mooney, and Kate Saenko. 2015. Translating videos to natural language using deep recurrent neural networks. In *NAACL-HLT*.

Violeta Voykinska, Shiri Azenkot, Shaomei Wu, and Gilly Leshed. 2016. How blind people interact with visual content on social networking services. In *CSCW*.

Bairui Wang, Lin Ma, Wei Zhang, Wenhao Jiang, Jingwen Wang, and Wei Liu. 2019a. Controllable video captioning with POS sequence guidance based on gated fusion network. In *ICCV*.

Bairui Wang, Lin Ma, Wei Zhang, and Wei Liu. 2018. Reconstruction network for video captioning. In *CVPR*.

Yiren Wang, Fei Tian, Di He, Tao Qin, ChengXiang Zhai, and Tie-Yan Liu. 2019b. Non-autoregressive machine translation with auxiliary regularization. In *AAAI*.

Qi Wu, Chunhua Shen, Lingqiao Liu, Anthony R. Dick, and Anton van den Hengel. 2016. What value do explicit high level concepts have in vision to language problems? In *CVPR*.

Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. MSR-VTT: A large video description dataset for bridging video and language. In *CVPR*.

Jun Xu, Ting Yao, Yongdong Zhang, and Tao Mei. 2017. Learning multimodal attention LSTM networks for video captioning. In *ACM MM*.

Chenggang Yan, Yunbin Tu, Xingzheng Wang, Yongbing Zhang, Xinhong Hao, Yongdong Zhang, and Qionghai Dai. 2020. STAT: spatial-temporal attention mechanism for video captioning. *IEEE Trans. Multim.*

Bang Yang, Fenglin Liu, and Yuexian Zou. 2019. Non-autoregressive video captioning with iterative refinement. *arXiv preprint arXiv:1911.12018*.

Junchao Zhang and Yuxin Peng. 2019. Object-aware aggregation with bidirectional temporal graph for video captioning. In *CVPR*.

Ziqi Zhang, Yaya Shi, Chunfeng Yuan, Bing Li, Peijin Wang, Weiming Hu, and Zheng-Jun Zha. 2020. Object relational graph with teacher-recommended learning for video captioning. In *CVPR*.

Qi Zheng, Chaoyue Wang, and Dacheng Tao. 2020. Syntax-aware action targeting for video captioning. In *CVPR*.

A Datasets, Metrics and Settings

A.1 Datasets

Our results are evaluated on the benchmark Microsoft Video Description (MSR-VTT) (Xu et al., 2016) and Microsoft Video Description (MSVD) (Guadarrama et al., 2013) datasets. For MSR-VTT, the dataset contains 10,000 video clips, and each video is paired with 20 annotated sentences. Following common practice (Pei et al., 2019; Yang et al., 2019; Pan et al., 2020), we use the official splits to report our results. Thus, there are 6513, 497 and 2990 video clips in the training set, validation set and test set, respectively. For MSVD, it contains 1,970 video clips and roughly 80,000 English sentences. We follow the split settings in Pei et al. (2019), resulting in 1,200, 100 and 670 videos for the training set, validation set and test set, respectively. Following previous works, we replace caption words that occur less than 3 times in the training set with the [UNK] token, plus with a [MASK] token, resulting in a vocabulary of 10,546 words for MSR-VTT and 9,467 words for MSVD.

A.2 Metrics

We test the model performance with a standard captioning evaluation toolkit (Chen et al., 2015). It reports the widely-used automatic evaluation metrics CIDEr (Vedantam et al., 2015), ROUGE-L (Lin, 2004), METEOR (Lin and Hovy, 2003; Banerjee and Lavie, 2005) and BLEU (Papineni et al., 2002). Among them, CIDEr, which incorporates the consensus of a reference set for an example, is based on n-gram matching, is specifically designed for

evaluating captioning systems. BLEU and METEOR are originally designed for machine translation evaluation, while ROUGE-L is proposed for automatic evaluation of the extracted text summarization. Besides, we further adopt the evaluation metrics Novel, Unique and Vocab Usage, provided by Dai et al. (2018), to evaluate the diversity of the generated captions. Novel is calculated by the percentage of generated captions that have not been seen in the training data; Unique is calculated by the percentage of generated unique words among the other all generated captions; Vocab Usage denotes the percentage of words that are used to generate captions in the vocabulary.

A.3 Settings

As stated in Section 2.1, we set $N = 8$, $d_i = d_m = 2048$ and $d_h = 512$ for the video representations. All category tags (Xu et al., 2016) included in MSR-VTT. For the object predictor, to compare with existing methods, we set the threshold $\gamma = 0.8$ and directly select all the predicted objects to generate captions. For the length predictor, the maximum sequence length l_{max} is set to 30. For the object generator and caption generator, following the original setting as in Transformer (Vaswani et al., 2017), the model size $d_h = 512$. The number of heads in multi-head attention is set to 8 and the feed-forward network dimension is set to 2048. The masking ratio $r = 0.5$. To build the object vocabulary, we use the spaCy library[4] for noun tagging from the training dataset, resulting in 5,647 and 4,681 noun words for MSR-VTT and MSVD, respectively. The tagged noun words are taken as the object words, building up the object vocabulary with sizes of 5,647 and 4,681 for MSR-VTT and MSVD, respectively. Therefore, we do not use external data to build the object vocabulary. Specifically, the object predictor labels will match the words used to name objects in the captions. We use Adam optimizer (Kingma and Ba, 2014) with a batch size of 64 and a learning rate of 5e-4 within maximum 50 epochs for parameter optimization.

As each video is annotated with multiple sentences, i.e., Video – $\{\text{Caption}_i\}$, where each sentence Caption_i includes a set of objects $\{\text{Object}_i\}$, we use all objects appearing in these sentences as the ground truth objects for each video to train the object predictor. However, we treat the different sentences as independent training samples, i.e., Video – Caption_i – $\{\text{Object}_i\}$, to train length predictor, object generator and caption generator. In this manner, we can ensure that the focused objects $\{\text{Object}_i\}$ appears in the target sentence Caption_i during training and inference, which allows an easy way to control the contents of video captions.

Following the non-autoregressive decoding models of neural machine translation, we incorporate the knowledge distillation (Kim and Rush, 2016; Gu et al., 2018) and de-duplication (Wang et al., 2019b) techniques to improve the performance of our non-autoregressive model on MSR-VTT. Furthermore, following Gu et al. (2018); Wang et al. (2019b); Yang et al. (2019), to generate the captions, we also adopt the teacher re-scoring technique and noisy parallel decoding (Gu et al., 2018; Yang et al., 2019) techniques, which could generate a set of candidate sentences in parallel, then, we select the candidate sentence with the highest output probability as the final generated caption. For the detailed introduction of these techniques, please refer to original papers (Kim and Rush, 2016; Gu et al., 2018; Wang et al., 2019b; Yang et al., 2019).

[4] https://spacy.io/