S2TD: A Tree-Structured Decoder for Image Paragraph Captioning

Yihui Shi
Beijing University of Posts and Telecommunications, 100876, China
yhshi@bupt.edu.cn

Yun Liu
Beijing University of Posts and Telecommunications, 100876, China
yunliu@bupt.edu.cn

Fangxiang Feng
Beijing University of Posts and Telecommunications, 100876, China
fxfeng@bupt.edu.cn

Ruifan Li*
Beijing University of Posts and Telecommunications, 100876, China
rfli@bupt.edu.cn

Zhanyu Ma
Beijing University of Posts and Telecommunications, 100876, China
mazhanyu@bupt.edu.cn

Xiaojie Wang
Beijing University of Posts and Telecommunications, 100876, China
xjwang@bupt.edu.cn

ABSTRACT

Image paragraph captioning, a task to generate the paragraph description for a given image, usually requires mining and organizing linguistic counterparts from abundant visual clues. Limited by sequential decoding perspective, previous methods have difficulty in organizing the visual clues holistically or capturing the structural nature of linguistic descriptions. In this paper, we propose a novel tree-structured visual paragraph decoder network, called *Splitting to Tree Decoder* (S2TD) to address this problem. The key idea is to model the paragraph decoding process as a top-down binary tree expansion. S2TD consists of three modules: a split module, a score module, and a word-level RNN. The split module iteratively splits ancestral visual representations into two parts through a gating mechanism. To determine the tree topology, the score module uses cosine similarity to evaluate the nodes splitting. A novel tree structure loss is proposed to enable end-to-end learning. After the tree expansion, the word-level RNN decodes leaf nodes into sentences forming a coherent paragraph. Extensive experiments are conducted on the Stanford benchmark dataset. The experimental results show promising performance of our proposed S2TD.

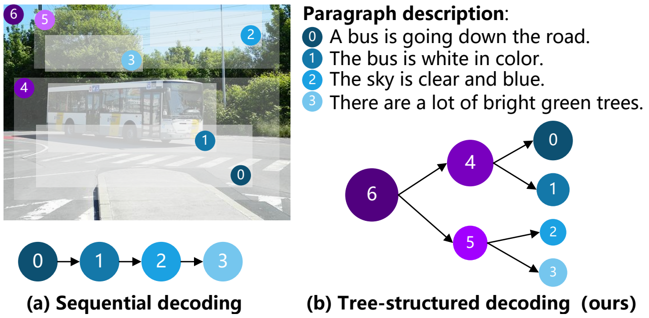
CCS CONCEPTS

• **Computing methodologies → Computer vision tasks**.

KEYWORDS

image captioning, paragraph generation, tree-structured decoder, vision and language

ACM Reference Format:
Yihui Shi, Yun Liu, Fangxiang Feng, Ruifan Li, Zhanyu Ma, and Xiaojie Wang. 2021. S2TD: A Tree-Structured Decoder for Image Paragraph Captioning. In *ACM Multimedia Asia (MMAsia '21), December 1–3, 2021, Gold Coast, Australia.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3469877.3490585



Paragraph description:
0 A bus is going down the road.
1 The bus is white in color.
2 The sky is clear and blue.
3 There are a lot of bright green trees.

(a) Sequential decoding **(b) Tree-structured decoding (ours)**

Figure 1: Two perspectives on visual paragraph generation.

1 INTRODUCTION

Image paragraph captioning aims to generate a semantically rich and coherent paragraph description for an image. This fundamental cross-modal task requires both visual understanding and linguistic processing. Although significant results have been achieved in single sentence captioning, such as [1, 5, 7, 9, 15, 21, 22, 33, 35, 36], generating a paragraph remains a challenge. Unlike a coarse sentence summary, the paragraph description of an image requires incorporating more visual details and considering the structural nature among multiple sentences. To this aim, the idea that the linguistic structure of a paragraph having a close relationship with visual planning should be emphasized.

Recently, various approaches have been proposed to improve the performance on image paragraph captioning. Generally, these approaches can be grouped into two categories. In the pioneering work, Krause et al. [11] propose a hierarchical decoder composed of a sentence-level and a word-level Recurrent Neural Network (RNN) to explicitly consider topic transition across sentences. With this intuitive modeling, variants of hierarchical decoders are then proposed [2, 3, 13, 14, 26, 29, 30, 32]. Besides, non-hierarchical decoders, which consider the paragraph as a sequence of words, also achieve comparable performance [16, 18, 27, 31].

However, the aforementioned strategy of sequential decoding has its disadvantage in nature. This strategy could over-simplify

[*] Corresponding Author.

MMAsia '21, December 1–3, 2021, Gold Coast, Australia
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-8607-4/21/12…$15.00
https://doi.org/10.1145/3469877.3490585

the visual observing procedure and ignore the latent sentence-level structure. As illustrated in Figure 1(a), the sequential decoding could start from any salient object and then decide where to describe next. The subsequent sentences largely depend on the previously generated ones but without a holistic control. Furthermore, this type of decoders could be stuck in or return to the same location. This results in the redundancy of visual information and the decoders probably choose a less consistent order to describe. Consequently, the overall coherence of paragraph could be impaired. Under the worst condition, the generated paragraph collapses into monotonously repetitive sentences.

To address the shortcomings of the sequential decoding methods, motivated by the human cognitive process from whole to part, we propose to model the paragraph decoding process as a top-down binary tree expansion. As illustrated in Figure 1(b), tree-structured decoding starts from the entire image and then progressively expands each intermediate observation into two subparts. After the expansion, all leaf nodes are collected to generate a coherent paragraph. Note that tree-structured decoding is superior to sequential decoding in two aspects. i) The top-down expansion encourages a more holistic control over visual clues. ii) Tree structures characterize relations within the paragraph more effectively by grouping topic-related sentences into the same sub-trees.

In this paper, we present a novel tree-structured visual paragraph decoder called *Splitting to Tree Decoder* (S2TD). To generate a paragraph description, S2TD first constructs a tree explicitly based on region-level visual representations and then decodes its leaf nodes from the left most to the right most into a paragraph. Specifically, S2TD composes three modules: a split module, a score module and a word-level RNN. The split module expands each parental visual representation into two parts. The score module controls the tree topology by providing a decision score for the current expansion generated by the split module. S2TD is trained end-to-end by a combination of cross-entropy loss and a tree structure loss. The latter loss learns from the pre-parsed structures which are bottom-up clustered out of ground truth paragraphs. Note that every node of the tree can be decoded by the word-level RNN which could lead to a better interpretability of paragraph generation.

To emphasize, our contributions are itemized as follows.

- We introduce a novel modeling perspective (i.e., tree structured decoding) for visual paragraph generation. Tree structures provide an inductive bias to model structural relations among generated sentences and to bridge visual observations and linguistic counterparts.
- We propose a novel tree structured decoder (i.e., S2TD) for image paragraph captioning task. We devise a topology prediction technique to perform the top-down binary tree expansion, consisting of a node splitting mechanism and a novel tree structure loss.
- We conduct extensive experiments on Stanford image paragraph benchmark dataset. We provide in-depth discussion on our proposed method, showcasing the benefits of tree-structured decoding for generating diverse and coherent paragraphs. Our source code is made public for knowledge sharing and further study.[1]

2 RELATED WORK

2.1 Image Paragraph Captioning

Methods of image paragraph captioning can be divided into two categories: the hierarchical approach [2, 3, 11, 13, 13, 14, 26, 29, 30, 32] and the non-hierarchical approach [16, 18, 27, 31]. For the hierarchical approach, paragraphs are modeled as a sequence of sentences. Topic representations of each sentence and their transition relationship play central roles. Krause et al. [11] present the pioneering work with a hierarchical solution. Topics are sequentially obtained from sentence-level RNN and then decoded into sentences by word-level RNN. Then, various improvements have been made by learning coherent topic transitions [2, 14], advanced attention mechanism [30], encoding richer visual information into topics [3, 26, 32], and imposing densely supervision over the hierarchy [29]. Nevertheless, most methods are limited by the sequential assumption thus mainly consider relationships between nearby sentences which are not effective enough.

For the non-hierarchical approach, paragraphs are modeled as a sequence of words. One line of works focus on tackling the problem of long-distance dependency. Specifically, Melas-Kyiazi et al. [18] adopt a trigram repetition penalty with self-critical strategy [21] to reduce redundancy. Luo et al. [16] combine intrinsic curiosity and extrinsic metric-based rewards to encourage better exploration on less frequently occurred phrases. Other works aim for better use of visual information by exploring depth estimation [27] and learning relationships between objects [31]. However, both the hierarchical and non-hierarchical approaches are largely based on sequential decoding for visual representations but the linguistic structures within paragraphs are ignored.

2.2 Tree-structured Decoders for Image

Very recently, incorporating tree structures into visual decoders has attracted attention. For example, Ma et al. [17] propose an image-to-tree framework to generate the sentence-level description. This method converts the dependency tree into a fully ternary tree, in which each node of the ternary tree corresponds to a word or a special token. To generate tree-structured markups from images involving math and chemical formula, Zhang et al. [34] design an RNN-based decoder that produces a sequence of sub-tree structures and can recurrently build the final tree. However, these two methods require ground-truth labels for each node which are usually not available. In contrast, our S2TD only requires leaf nodes labelling. Besides, Wang et al. [25] propose to infer tree structures of recipes from food images, and then obtain tree embeddings with graph attention networks to generate better recipes. In comparison, our S2TD tries to align the observation of images with trees during decoding.

3 APPROACH

Image paragraph captioning aims to describe an image $\mathcal{I}$ with a paragraph $\mathcal{P}$, which consists of M sentences, i.e., $\mathcal{P} = \{\mathcal{S}_i\}_{i=1}^{M}$. Each sentence $\mathcal{S}_i$ consists of N_i words, i.e., $\mathcal{S}_i = \{w_{i,j}\}_{j=1}^{N_i}$. First, a set of K region-level visual semantic representations $\mathcal{V}$ are extracted from the image $\mathcal{I}$, where $\mathcal{V} = \{\boldsymbol{v}_k^{bbox}\}_{k=1}^{K}$ and the bounding box $\boldsymbol{v}_k^{bbox} \in \mathbb{R}^{D_v}$. Furthermore, as an informative summary of overall

[1] The source code is available at https://github.com/bupt-mmai/S2TD.

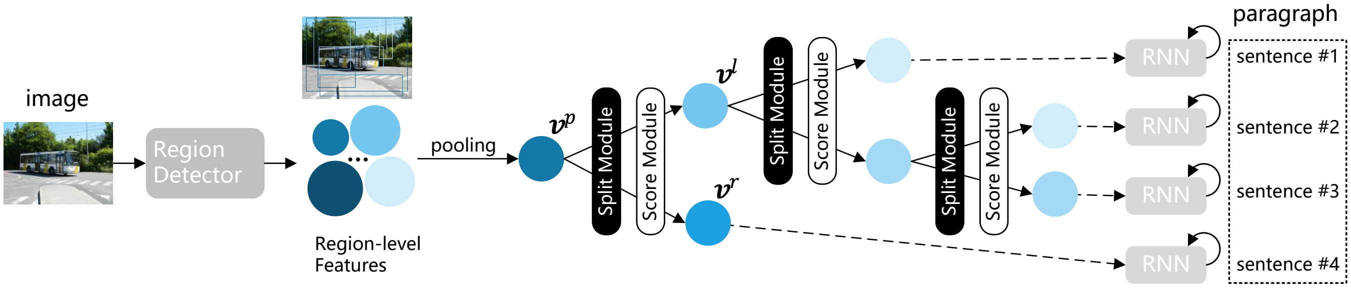


Figure 2: The framework of proposed tree-structured visual paragraph decoder, i.e., S2TD. Three key components, including a split module, a score module and a word-level RNN work collaboratively to describe an image.

visual clues, the global representation of the image is obtained by using element-wise maximum, i.e., $\boldsymbol{v}^g = \max(\{\boldsymbol{v}_k^{bbox}\}_{k=1}^K)$.

The framework of our proposed S2TD tree-structured decoder is depicted in Figure 2. Given the global representation $\boldsymbol{v}^g$ as the first node input, the split module learns to expand it into two nodes. Controlled by the score module, the expansion of each node is repeated until certain condition meets. Once the tree has been constructed, all leaf nodes are input into the word-level RNN to generate several sentences, entirely forming a coherent paragraph.

3.1 Our Proposed S2TD

S2TD is a hierarchical decoder composed of three components: a split module, a score module, and a word-level RNN.

Split Module. A basic operation of top-down binary tree expansion is to split a node into two. Such operation should be able to meet two requirements. i) Preserve parent node information without incorporating noise during the tree expansion. ii) Model explicit difference between left and right child nodes. Thus, we design a simple but effective split module to calculate two child nodes' representations with respect to the parent node.

Specifically, given a parent node representation $\boldsymbol{v}^p$, a gated unit with LayerNorm [12] is computed as follows,

$$\boldsymbol{g}^s = \sigma\left(W_s \cdot \text{LayerNorm}(\boldsymbol{v}^p) + b_s\right), \tag{1}$$

where $W_s \in \mathbb{R}^{D_v \times D_v}$ and $b_s \in \mathbb{R}^{D_v}$ are learnable weights and biases, respectively. The symbol σ denotes the Sigmoid function. The gate $\boldsymbol{g}^s$ controls the parent node information being passed to the left child node. Symmetrically, $\mathbf{1} - \boldsymbol{g}^s$ does for the right child node. Then, we obtain the representations of the left and the right child nodes using two equations,

$$\begin{aligned} \boldsymbol{v}^l &= \boldsymbol{v}^p \odot \boldsymbol{g}^s, \\ \boldsymbol{v}^r &= \boldsymbol{v}^p \odot \left(\mathbf{1} - \boldsymbol{g}^s\right), \end{aligned} \tag{2}$$

where $\odot$ denotes the element-wise product. Note that the design of $\boldsymbol{g}^s$ and $\mathbf{1} - \boldsymbol{g}^s$ encourages an explicit difference on representations between the left and right nodes.

Score Module. One central problem with the tree-structured decoding is to predict the topology of the tree. As we model the constructing procedure as a top-down binary tree expansion, we address this problem by progressively deciding whether to split the current leaf node. To be specific, given a set of proposal $\{\boldsymbol{v}^p, \boldsymbol{v}^l, \boldsymbol{v}^r\}$ produced by above split module, the score module then calculates a decision score s^p by cosine similarity as

Algorithm 1 Binary tree expansion during inference

```
Input: Global visual features $v^g$, maximum number of sentences $M$, decision score threshold $\alpha$
Initialize: An empty queue $Q$, an empty tree $\mathcal{T}$
Output: Expanded tree $\mathcal{T}$
Push $v^g$ into $Q$ and register $v^g$ to $\mathcal{T}$.
while $Q$ is not empty and numbers of leaf nodes in $\mathcal{T}$ is smaller than $M$ do
    Pop $v^p$ out of $Q$.
    Expand $v^p$ into $v^l$ and $v^r$ by Split Module.
    Calculate score $s^p$ by Score Module.
    if $s^p \leq \alpha$ then
        Push $v^l$ then $v^r$ into $Q$.
        Register $v^l$ and $v^r$ to $\mathcal{T}$.
    end if
end while
return $\mathcal{T}$
```

$$s^p = \cos\left(\boldsymbol{v}^l, \boldsymbol{v}^r\right) = \frac{(\boldsymbol{v}^l)^{\mathrm{T}} \boldsymbol{v}^r}{\|\boldsymbol{v}^l\| \|\boldsymbol{v}^r\|}. \tag{3}$$

We could keep or discard the proposal based on the decision score s^p. If the score $s^p \leq \alpha$ then we keep the proposal. The hyperparameter $\alpha \in [0, 1]$ denotes a constant threshold. The intuitive behind is that overly similar node representations would result in redundant descriptions. The binary tree expansion during inference is provided in Algorithm 1.

Word-level RNN. After the tree expansion, the word-level RNN is responsible for generating the words of a sentence conditioned on a node representation. To obtain the i-th sentence of the paragraph, a single layer Highway Network [23] is first used to transform the corresponding leaf node representation $\boldsymbol{v}_i^{leaf}$ into a topic vector $\boldsymbol{t}_i \in \mathbb{R}^{D_e}$ by

$$\boldsymbol{t}_i = \text{Highway}_0(\boldsymbol{v}_i^{leaf}). \tag{4}$$

Then, we adopt LSTM [8] to perform sentence decoding by recurrently obtaining the hidden state $\boldsymbol{h}_{i,j} \in \mathbb{R}^{D_e}$ by

$$\boldsymbol{h}_{i,j} = \text{LSTM}(W_e \boldsymbol{w}_{i,j-1}, \boldsymbol{h}_{i,j-1}), \tag{5}$$

where $\boldsymbol{w}_{i,j-1} \in \mathbb{R}^{D_w}$ denotes the one-hot coding of the corresponding input word. The weight $W_e \in \mathbb{R}^{D_e \times D_w}$ is a learnable word embedding matrix with a vocabulary size D_w. Empirically, the hidden state is initiated by $\boldsymbol{h}_{i,0} = \mathrm{Highway}_1(\boldsymbol{t}_i)$ and the memory cell is initiated as zeros. By concatenating hidden states at each step with the topic vector, the conditional distribution over output words is calculated as follows,

$$\boldsymbol{p}_{i,j} = \mathrm{Softmax}\left(W_p\left[\boldsymbol{h}_{i,j};\boldsymbol{t}_i\right] + b_p\right), \tag{6}$$

where $W_p \in \mathbb{R}^{D_w \times 2D_e}$ and $b_p \in \mathbb{R}^{D_w}$ are learnable weights and biases, respectively.

3.2 Learning Tree Topology

Pre-Parsed Paragraph Tree. We propose to use a hierarchical clustering method to mine tree structures from ground-truth paragraphs. We first encode each sentence into a dense vector by a pre-trained language model. Then, the classical Ward's minimum variance method [28] is used. We impose a constraint that only nearby clusters can be merged. Finally, we obtain a tree in which leaf nodes from left to right correspond to the sentences of the given paragraph. Note that pre-parsed trees are not required during inference.

Tree Structure Loss. During training, the choices of node splitting and topology learning targets are determined by the corresponding pre-parsed tree. For a target tree $\mathcal{T}$, we design a learning criteria for each node's decision score s^p as follows,

$$\mathcal{L}_T(s^p) = \begin{cases} \max\left(0, \cos(\boldsymbol{v}^l, \boldsymbol{v}^r)\right) & p \notin \mathbb{L} \\ \max\left(0, \alpha - \cos(\boldsymbol{v}^l, \boldsymbol{v}^r)\right) & p \in \mathbb{L}, \end{cases} \tag{7}$$

where $\mathbb{L} \subset \mathcal{T}$ denotes the set of leaf nodes. For nodes to be split (i.e., non-leaf), our loss encourages their child nodes to become less similar. In contrast, for nodes to be stopped (i.e., leaf), our loss pushes the decision score over the threshold α. Thus, we can smooth the learning of score transition between non-leaf and leaf nodes.

3.3 Loss and Training

Supervise Learning. For a training sample $(\mathcal{I}, \mathcal{P}^{gt})$ where $\mathcal{P}^{gt}$ denotes the ground truth paragraph description for the image $\mathcal{I}$, we first acquire the pre-parsed tree-structure $\mathcal{T}$ over $\mathcal{P}^{gt}$. Then, S2TD is trained in an end-to-end fashion. The loss function is defined as a combination of two terms: a tree structure loss $\mathcal{L}_T$ on decision score and a cross-entropy loss $\mathcal{L}_W$ on word-level distribution. Specifically, The total loss $\mathcal{L}$ is defined as follows,

$$\mathcal{L} = \sum_{p \in \mathcal{T}} \mathcal{L}_T\left(s^p\right) + \sum_{i=1}^{M}\sum_{j=1}^{N_j} \mathcal{L}_W\left(\boldsymbol{p}_{i,j}, \hat{w}_{i,j}\right). \tag{8}$$

Reinforcement Learning. Using self-critical sequence training (SCST) [18, 21], S2TD can be further optimized by non-differentiable metrics. Specifically, the expected gradient of a metric is calculated by SCST as

$$\nabla_\theta \mathrm{L}(\theta) = -\mathrm{E}_{w^s \sim p_\theta}[(r(w^s) - r(w^g))\nabla_\theta \log p_\theta(w^s)], \tag{9}$$

where w^s and w^g denote the sampled and greedily decoded paragraph, respectively. r denotes a reward from metrics and p_θ represents the captioning model. For simplicity, S2TD does not perform tree topology learning during SCST.

4 EXPERIMENTS

4.1 Dataset and Metrics

We conduct the experiments on Stanford image paragraph benchmark dataset collected in [11]. The dataset contains 19551 images. The training set contains 14575 image-paragraph pairs, the validation set has 2487 pairs, and the testing set has 2489 pairs. On average, each paragraph caption contains 67.5 words and 5.7 sentences, each of which consists of an average of 11.9 words. We use the public COCO captioning evaluation tool [4]. Six standard metrics: BLEU-{1,2,3,4} [19], METEOR [6] and CIDEr [24] are adopted.

4.2 Implementation Details

We apply widely used region detector Faster-RCNN [1] to detect regions of interest and to extract region-level features. Top 36 regions are detected for each image and the dimensionality of extracted features is 2048. Then, we use a learnable single layer Highway Network to reduce its dimensionality, i.e., $D_v = 1024$. During the training and inference, the parameters of the region detector are kept unchanged.

For each paragraph, at most six sentences are kept, and each sentence contains at most 31 words. We replace words that appear less than two times with a special token <unk>. Symbols <bos> and <eos> are included which denote the starting and ending of a sentence, respectively. The final vocabulary size D_w euqals 6730. To obtain pre-parsed paragraph trees, we use the sentence transformer, i.e., Sentence-BERT [20] to encode sentences into dense representations.

For our S2TD, we use a double-layer LSTM in the word-level RNN. Both the embedding size and the hidden size of each layer are set to $D_e = 512$. The threshold α is set to be 0.3. The word-level RNN adopts greedy search during inference. For supervised learning, S2TD is trained by Adam [10] optimizer with a learning rate 5×10^{-4} and a batch size 16. For every epoch, the average score of BLEU-4 and METEOR is calculated on the validation set. We reduce the learning rate by 0.8 every five epochs and perform early stopping after 20 epochs stagnation. Our reinforcement learning adopts the same procedure as in [18], except that we adopt both BLEU-4 and METEOR as the reward function. The learning rate is fixed at 3×10^{-5} and the model is tuned for at most 50 epochs. All model checkpoints and settings are chosen on the validation performance.

4.3 Performance Evaluation

We compare our proposed S2TD with state-of-the-art sequential decoding methods. These sequential decoding methods are categorized into two groups: i) Hierarchical methods, including **RH** [11], **RTT-GAN** [14], **CAPG-VAE** [2], **Dual-CNN** [13], **CAE-LSTM**[26], **DHPV** [29] and **IMAP** [30]; and ii) Non-Hierarchical methods, including **DAM-ATT** [27], **SCST** [18], **CRL** [16] and **OR-ATT** [31]. All experimental results are shown in Table 1.

In Table 1, we have the following two findings. i) Under the non-RL setting, hierarchical methods consistently outperform non-hierarchical methods. By explicitly modeling sentence representations, hierarchical models can better capture topic transition within paragraphs. Based on that, our S2TD further introduces

Table 1: Performance comparison of state-of-the-arts on Stanford image paragraph benchmark dataset. Hierarchical and non-hierarchical methods are grouped in the middle. Symbols B{1, 2, 3, 4}, M, C are short for BLEU-{1, 2, 3, 4}, METEOR, CIDEr.

Model	Non-RL Performance						RL Performance					
	B1	B2	B3	B4	M	C	B1	B2	B3	B4	M	C
RH [11]	41.90	24.11	14.23	8.69	15.95	13.52	-	-	-	-	-	-
RTT-GAN [14]	41.99	24.86	14.89	9.03	17.12	16.87	-	-	-	-	-	-
CAPG-VAE [2]	42.38	25.52	15.15	9.43	**18.62**	20.93	-	-	-	-	-	-
Dual-CNN [13]	41.60	24.40	14.30	8.60	15.60	17.40	-	-	-	-	-	-
CAE-LSTM [26]	-	-	-	-	-	-	-	-	-	9.67	**18.82**	25.15
DHPV [29]	40.35	24.45	15.41	**10.03**	16.10	20.70	43.35	26.73	16.92	**10.99**	17.02	22.47
IMAP [30]	42.38	25.87	**15.51**	9.42	16.56	20.76	44.45	27.93	17.14	10.29	17.36	24.07
DAM-ATT [27]	35.02	20.24	11.68	6.57	13.91	17.32	-	-	-	-	-	-
SCST [18]	32.78	19.00	11.40	6.89	13.66	12.89	29.67	16.45	9.74	5.88	13.63	13.77
SCST+RP [18]	35.68	22.40	14.04	8.70	15.17	22.68	43.54	27.44	17.33	10.58	17.86	30.63
CRL [16]	-	-	-	-	-	-	43.12	27.03	16.72	9.95	17.42	31.47
OR-ATT [31]	34.97	20.17	12.21	7.46	13.58	16.27	32.84	18.30	10.67	6.21	13.44	14.88
OR-ATT+RP [31]	37.50	23.34	14.63	9.00	15.43	**22.85**	43.76	**28.08**	**17.88**	10.95	17.82	**33.38**
Our S2TD	44.32	25.86	14.80	8.33	16.89	21.41	43.70	26.67	16.30	9.79	17.32	22.84
Our S2TD+RP	**44.59**	**26.06**	14.93	8.35	17.00	21.92	**44.47**	27.38	16.87	10.17	17.64	24.33

Table 2: Ablation study on S2TD under diverse variants.

Model	B1	B2	B3	B4	M	C
Split-H	42.83	24.46	13.74	7.58	16.38	19.00
Split-G	42.87	24.99	14.61	8.46	16.37	19.50
Score-C	37.37	21.21	11.82	6.44	14.74	16.27
BFS-R	44.30	25.91	14.80	8.28	16.83	20.51
DFS-L	44.02	25.57	14.60	8.19	16.71	20.82
DFS-R	43.34	25.04	14.42	7.99	16.45	19.53
Full S2TD	44.32	25.86	14.80	8.33	16.89	21.41

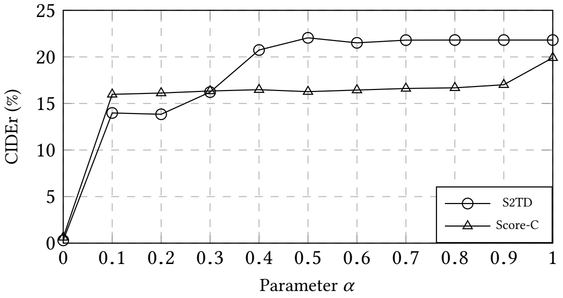


Figure 3: S2TD compared with Score-C on α of the tree loss.

structural assumptions. With only a simple tree architecture, our S2TD achieves competitive performance, especially in terms of BLEU-1 and CIDEr. This demonstrates the effectiveness of tree-structured decoding.

ii) Under the RL setting, richer visual encoding (e.g. CAE-LSTM and OR-ATT) and advanced reinforcement learning methods (e.g. DHPV and CRL) are the keys to high performance. The trial and error property of RL mitigates the structural shortage of non-hierarchical methods by sampling numerous paragraphs. However, without repetition penalty (i.e., RP), the performance of non-hierarchical methods like SCST and OR-ATT will decrease dramatically due to redundant phrases. Hierarchical methods like our S2TD are more robust with less drop. In general, our S2TD achieves competitive performance compared with state-of-the-art methods.

4.4 Ablation Study

In this section, we conduct ablation studies to better understand our proposed S2TD. Specifically, we evaluate variants on module design and diverse inference strategies.

On module design. We compare S2TD with three variants. i) **Split-H** replaces the gate unit in the split module with two single-layer Highway networks. ii) **Split-G** utilizes two independent gates instead of one. iii) **Score-C** adopts a binary classification variant of the score module. Given the parent node, the decision score is provided by a MLP with a sigmoid activation. The tree structure loss is replaced with a binary cross-entropy. During inference, the node splitting is performed if the predicted score is less than 0.5. The results are reported in Table 2.

We observe that S2TD benefits more from the gating mechanism compared to direct node generation. The gating design in Eq. (2) further improves the performance. These showcase the effectiveness of our split module. Nevertheless, the large gap between Score-C and S2TD indicates that our proposed method is more reasonable when performing tree expansion. Note that the learning targets of Score-C change dramatically during the transition from non-leaf to leaf nodes. In contrast, our proposed tree structure loss is much smoother as we optimize relative similarity instead of specific targets.

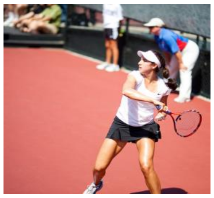
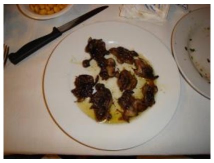
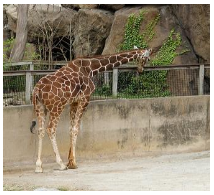
	SCST + RP	S2TD (Ours)	Sentence Tree by S2TD
	A woman is standing on a tennis court. She is wearing a white visor and a white skirt. The woman is holding a tennis racket. The woman is holding a racket in her hands. The court is green and white. The court is made of green. The tennis court is white. There are people sitting on the court. The man is wearing a black shirt.	A woman is standing on a tennis court playing tennis. She is wearing a white shirt and white shorts. She is wearing a white visor a white skirt and a white skirt. She is holding a tennis racket in her hand. A chain link fence is behind the woman. There are many people sitting in the stands watching the people.	**it is sunny outside and the tennis court is sunny** ├── a woman is playing tennis on a tennis court │ ├── a woman is standing on a tennis court playing tennis │ │ ├── a woman is standing on a tennis court playing tennis │ │ └── she is wearing a white shirt and white shorts │ └── she is holding a tennis racket in her hand │ ├── she is wearing a white visor a white skirt and a white skirt │ └── she is holding a tennis racket in her hand └── behind the fence there are many people watching ├── a chain link fence is behind the woman └── there are many people sitting in the stands watching the people
	A plate of food is on a plate. The plate is white. The plate is white. There is a plate on the plate. There is a fork on the plate. The plate has a white plate on it. There are two pieces of food on the table. There are a fork on the plate. There is a white bowl on the table. The table is white and white .	There is a table with a white tablecloth on it. There is a round plate on the table. On the plate is a plate of food that has a white sauce carved into it. The knife has been cut into a bun. There are pieces of food on top of the table. There are peas and carrots on the table in front of the plate.	**a plate of food is on a table** ├── a plate of food is on a table │ ├── this is a plate of food │ │ ├── there is a table with a white tablecloth on it │ │ └── there is a round plate on the table │ └── on the plate is a plate of food that has a white sauce carved into it └── there are also two glasses of food on top of the table ├── there is also a knife and fork on the table │ ├── the knife has been cut into a bun │ └── there are pieces of food on top of the table └── there are peas and carrots on the table in front of the plate
	A giraffe is standing in a zoo. The giraffe is standing on the ground. The giraffe is brown. The giraffe has a long neck. The giraffes are standing on a rock. The wall is made of wood. The rocks are brown. There is a rock on the wall. The tree is made of dirt .	A giraffe is standing in front of a large rock. It is standing on a dirt ground and there is a large rock to the right of the giraffe. The giraffe is leaning against a wall. There is a fence behind the giraffe and a stack of rocks and plants. The sun is shining on the enclosure and the enclosure.	**a giraffe with brown spots is standing near the fence** ├── a giraffe is standing on the wall │ ├── a giraffe is standing in front of a large rock │ │ ├── a giraffe is standing in front of a large rock │ │ └── it is standing on a dirt ground and there is a large rock to the right of the giraffe │ └── the giraffe is leaning against a wall └── the enclosure is surrounded by a large rock wall ├── there is a fence behind the giraffe and a stack of rocks and plants └── the sun is shining on the enclosure and the enclosure

Figure 4: Qualitative comparison on generated captions for images with SCST+RP and our S2TD.

We further compare our S2TD with Score-C to show the motivation of our tree loss. Specifically, given a trained model, we vary its threshold to observe its effect on text generation. The result is shown in Figure 3. For a fair comparison, we retrain another S2TD with $\alpha = 0.5$. Compared to Score-C, S2TD shows smoother change of the CIDEr scores, when α increasing from 0.1 to 0.5. It shows that S2TD can better evaluate different splittings by more continuous scores which verify the plausibility of our tree structure loss.

On inference strategy. Our inference strategy has been shown in Algorithm 1. We can replace the queue with a stack or reverse the pushing order of the child nodes. Thus, if we denote Algorithm 1 as **BFS-L** (i.e., Breadth-First Searching with left node first), we could have the other three: **BFS-R**, **DFS-L**, and **DFS-R**. The results are reported in Table 2. We observe that BFS strategies outperform DFS. This result is in line with expectations since BFS aims to capture broader topics while DFS tries to detail specific topics. Also, we observe that expanding left nodes first leads to better metrics. We suppose that it is because the left sub-tree of the root mainly describes the salient objects while the right focuses on the background. This will be illustrated in Section 4.5.

4.5 Qualitative Analysis

In this section, we demonstrate the model capability by comparing the captioning results with those of SCST+RP model.

On Paragraph. In Figure 4, we present description examples by SCST+RP and our S2TD. We induce two pros of S2TD. i) Paragraphs are less redundant. For the first image, the SCST+RP repetitively describes 'tennis racket' and 'tennis court', while our S2TD provides a compact description. The sentence tree provides additional evidence. The sentences of the same sub-tree are topic-related and with different aspects. ii) Paragraphs are more coherent. For the second image, compared to SCST+RP, our S2TD changes the viewpoint from 'plate' to 'table' more smoothly by better utilizing the relation between the aforementioned objects. Moreover, S2TD provides more details like 'knife', 'peas' and 'carrots'.

On Sentence Tree. In Figure 4, we have two observations from the sentence trees. i) Each node representation can be decoded into a fluent and image-related sentence. As expected, the sentence of root node which corresponds to the global visual representation provides overall impression. ii) The split of parent node is semantically image-related. For the third image, S2TD first divides the image into the salient object 'giraffe' and the background 'enclosure'. Then, the background details such as 'rock wall' and 'plants' are obtained. This type of decoding process is close to human cognitive process. This also demonstrates the motivation of our designed node-splitting mechanism.

5 CONCLUSION

In this work, we propose a tree-structured decoder S2TD for image paragraph captioning, which models the paragraph decoding process as a top-down binary tree expansion. We show the promising performance of tree-structured decoding for generating diverse and coherent paragraphs. In the future, we will extend our method with richer visual encoding, and devise a corresponding RL method for tree structures.

ACKNOWLEDGMENTS

This work was supported in part by the under Grant 2019YFF0303300 and Subject II under Grant 2019YFF0303302, in part by the National Natural Science Foundation of China under Grants 61802026 and 61906018, in part by the 111 Project under Grant B08004, and in part by the Fundamental Research Funds for the Central Universities under Grant 2021RC36.

REFERENCES

[1] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 6077–6086.

[2] Moitreya Chatterjee and Alexander G Schwing. 2018. Diverse and coherent paragraph generation from images. In *European Conference Conputer Vision (ECCV)*. 729–744.

[3] Wenbin Che, Xiaopeng Fan, Ruiqin Xiong, and Debin Zhao. 2018. Paragraph generation network with visual relationship detection. In *ACM Conference on Multimedia (MM)*. 1435–1443.

[4] 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. (2015). http://de.arxiv.org/pdf/1504.00325

[5] Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. 2020. Meshed-memory transformer for image captioning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 10578–10587.

[6] Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Workshop on Statistical Machine Translation*. 376–380.

[7] Simao Herdade, Armin Kappeler, Kofi Boakye, and Joao Soares. 2019. Image captioning: Transforming objects into words. In *Advances in Neural Information Processing Systems (NeuIPS)*. 11137–11147.

[8] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation* 9, 8 (1997), 1735–1780.

[9] Lun Huang, Wenmin Wang, Jie Chen, and Xiao-Yong Wei. 2019. Attention on attention for image captioning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 4634–4643.

[10] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations (ICLR)*. https://arxiv.org/abs/1412.6980v8

[11] Jonathan Krause, Justin Johnson, Ranjay Krishna, and Li Fei-Fei. 2017. A hierarchical approach for generating descriptive image paragraphs. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 317–325.

[12] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. In *Deep Learning Symposium, Advances in Neural Information Processing Systems (NIPS)*. https://openreview.net/forum?id=BJLa_ZC9

[13] Ruifan Li, Haoyu Liang, Yihui Shi, Fangxiang Feng, and Xiaojie Wang. 2020. Dual-CNN: A Convolutional language decoder for paragraph image captioning. *Neurocomputing* (2020).

[14] Xiaodan Liang, Zhiting Hu, Hao Zhang, Chuang Gan, and Eric P Xing. 2017. Recurrent topic-transition gan for visual paragraph generation. In *IEEE International Conference on Computer Vision (ICCV)*. 3362–3371.

[15] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2018. Neural baby talk. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 7219–7228.

[16] Yadan Luo, Zi Huang, Zheng Zhang, Ziwei Wang, Jingjing Li, and Yang Yang. 2019. Curiosity-driven reinforcement learning for diverse visual paragraph generation. In *ACM Conference on Multimedia (MM)*. 2341–2350.

[17] Zhiming Ma, Chun Yuan, Yangyang Cheng, and Xinrui Zhu. 2019. Image-to-Tree: A Tree-Structured Decoder for Image Captioning. In *IEEE International Conference on Multimedia and Expo (ICME)*. 1294–1299.

[18] Luke Melas-Kyriazi, Alexander M Rush, and George Han. 2018. Training for diversity in image paragraph captioning. In *The 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 757–761.

[19] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Annual Meeting of the Association for Computational Linguistics (ACL)*. 311–318.

[20] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 3982–3992.

[21] Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 7008–7024.

[22] Zhan Shi, Xu Zhou, Xipeng Qiu, and Xiaodan Zhu. 2020. Improving Image Captioning with Better Use of Caption. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*. 7454–7464.

[23] Rupesh K Srivastava, Klaus Greff, and Jürgen Schmidhuber. 2015. Training very deep networks. *Advances in Neural Information Processing Systems (NIPS)* 28 (2015), 2377–2385.

[24] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 4566–4575.

[25] Hao Wang, Guosheng Lin, Steven CH Hoi, and Chunyan Miao. 2020. Structure-Aware Generation Network for Recipe Generation from Images. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 359–374.

[26] Jing Wang, Yingwei Pan, Ting Yao, Jinhui Tang, and Tao Mei. 2019. Convolutional auto-encoding of sentence topics for image paragraph generation. In *International Joint Conference on Artificial Intelligence (IJCAI)*. 940–946.

[27] Ziwei Wang, Yadan Luo, Yang Li, Zi Huang, and Hongzhi Yin. 2018. Look deeper see richer: Depth-aware image paragraph captioning. In *ACM Conference on Multimedia (MM)*. 672–680.

[28] Joe H Ward Jr. 1963. Hierarchical grouping to optimize an objective function. *J. Amer. Statist. Assoc.* 58, 301 (1963), 236–244.

[29] Siying Wu, Zheng-Jun Zha, Zilei Wang, Houqiang Li, and Feng Wu. 2019. Densely Supervised Hierarchical Policy-Value Network for Image Paragraph Generation.. In *International Joint Conference on Artificial Intelligence (IJCAI)*. 975–981.

[30] Chunpu Xu, Yu Li, Chengming Li, Xiang Ao, Min Yang, and Jinwen Tian. 2020. Interactive Key-Value Memory-augmented Attention for Image Paragraph Captioning. In *Proceedings of the 28th International Conference on Computational Linguistics (COLING)*. 3132–3142.

[31] Li-Chuan Yang, Chih-Yuan Yang, and Jane Yung-jen Hsu. 2021. Object Relation Attention for Image Paragraph Captioning. In *AAAI Conference on Artificial Intelligence (AAAI)*, Vol. 35. 3136–3144.

[32] Xu Yang, Chongyang Gao, Hanwang Zhang, and Jianfei Cai. 2020. Hierarchical Scene Graph Encoder-Decoder for Image Paragraph Captioning. In *ACM Conference on Multimedia (MM)*. 4181–4189.

[33] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. 2019. Hierarchy parsing for image captioning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2621–2629.

[34] Jianshu Zhang, Jun Du, Yongxin Yang, Yi-Zhe Song, Si Wei, and Lirong Dai. 2020. A Tree-Structured Decoder for Image-to-Markup Generation. In *International Conference on Machine Learning (ICML)*. 11076–11085.

[35] Xuying Zhang, Xiaoshuai Sun, Yunpeng Luo, Jiayi Ji, Yiyi Zhou, Yongjian Wu, Feiyue Huang, and Rongrong Ji. 2021. RSTNet: Captioning With Adaptive Attention on Visual and Non-Visual Words. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 15465–15474.

[36] Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason Corso, and Jianfeng Gao. 2020. Unified vision-language pre-training for image captioning and vqa. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, Vol. 34. 13041–13049.