applied sciences

MDPI

Article

Boosted Transformer for Image Captioning

Jiangyun Li [1,2,‡], Peng Yao [1,2,†,‡], Longteng Guo [3] and Weicun Zhang [1,2,*]

[1] School of Automation & Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China
[2] Key Laboratory of Knowledge Automation for Industrial Processes, Ministry of Education, Beijing 100083, China
[3] National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
* Correspondence: weicunzhang@ustb.edu.cn
† Current address: University Of Science and Technology Beijing, Beijing 100083, China.
‡ These authors contributed equally to this work.

Received: 17 July 2019; Accepted: 5 August 2019; Published: 9 August 2019

Abstract: Image captioning attempts to generate a description given an image, usually taking Convolutional Neural Network as the encoder to extract the visual features and a sequence model, among which the self-attention mechanism has achieved advanced progress recently, as the decoder to generate descriptions. However, this predominant encoder-decoder architecture has some problems to be solved. On the encoder side, without the semantic concepts, the extracted visual features do not make full use of the image information. On the decoder side, the sequence self-attention only relies on word representations, lacking the guidance of visual information and easily influenced by the language prior. In this paper, we propose a novel boosted transformer model with two attention modules for the above-mentioned problems, i.e., "Concept-Guided Attention" (CGA) and "Vision-Guided Attention" (VGA). Our model utilizes CGA in the encoder, to obtain the boosted visual features by integrating the instance-level concepts into the visual features. In the decoder, we stack VGA, which uses the visual information as a bridge to model internal relationships among the sequences and can be an auxiliary module of sequence self-attention. Quantitative and qualitative results on the Microsoft COCO dataset demonstrate the better performance of our model than the state-of-the-art approaches.

Keywords: image captioning; self-attention; deep learning; transformer

1. Introduction

Image captioning, a challenging task to generate a descriptive sentence within an image automatically, has gained increasing attention as a prominent interdisciplinary research problem in both the Computer Vision (CV) and Natural Language Processing (NLP) areas recently. This task has many important practical applications, such as assisting visually-impaired people to understand image content or improving image retrieval quality by discovering salient content. For humans, this task is easy to achieve, while it is incredibly difficult for machines because they not only need to recognize the specific objects and the relationship between them in the image, but also need to integrate the aforementioned elements into a properly-formed sentence.

Even though many works [1–5] have been devoted to improving the quality of generated descriptions, almost all the proposed methods are based on the CNN + LSTM framework, where the Convolutional Neural Network (CNN) is adopted as the image encoder to extract visual features and Long Short Term Memory (LSTM) is applied as the caption decoder to generate sentences. With the development of Neural Machine Translation (NMT), a novel architecture appeared, transformer [6].

It is based on the self-attention mechanism and has advanced the state-of-the-art methods on many NLP tasks. Without recurrence, transformer accelerates the training process and uses the self-attention to draw global dependencies between different inputs. We apply the transformer architecture to the image captioning task in this paper. The common methods regard the CNN and the transformer encoder as the whole image encoder and the transformer decoder as the caption decoder. However, such an architecture for the image captioning task has some limitations as follows.

For the image encoder, semantic concepts, as shown in Figure 1, are full of rich semantic cues and can be regarded as complementary knowledge to visual features. However, the commonly-used image information is only from the visual features. The lack of semantic concepts greatly increases the training difficulty since the unsolved semantic alignment problem has to be settled when training. Therefore, we devise a concept-guided attention module to boost these visual features with the aid of instance-level semantic concepts. Concept-guided attention is composed of two parallel self-attention modules and an integration module, where the former is used to contextualize the visual features and semantic concepts individually, and the latter is applied, complementing each other for boosted visual features.

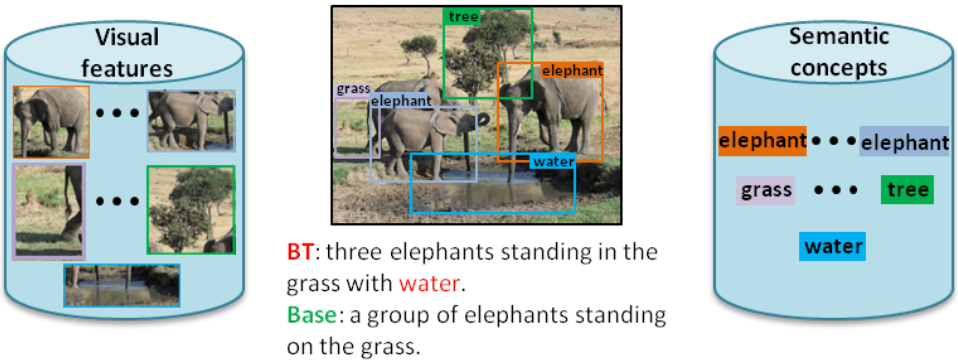

Figure 1. Illustrations of the completed image representations, which are composed of the visual features and the corresponding semantic concepts. Compared with the baseline transformer, our model, Boosted Transformer (BT), can generate more image-associated descriptions. The red words represent the novel tokens discovered by our model.

For the caption decoder, transformer utilizes the self-attention mechanism to contextualize the sequence information. This mechanism only uses the word representations of tokens as inputs, and we name it as sequence self-attention (seq-to-self). However, without the guidance of visual information, sequence self-attention has some bias in modeling the correlation between different word representations. This further generates an inaccurate sequence representation because the correlation between word representations is changeable with different visual scenarios. As a result, the obtained sequence representation is possible to focus on a mistaken region in sequence-to-image (seq-to-image) attention, which is used to attend to specific image regions. For example, the word "elephant" is close to the word "grass" and far from "water" since the descriptions containing both "elephant" and "grass" appear more frequently in the training set. Under the influence of this inductive bias [7], the trained model is apt at producing a result for which if an image contains the objects "elephant" and "grass", the fixed collocation in its description, "elephant standing on the grass", appears with a high probability, as shown in Figure 1. Therefore, we propose vision-guided attention to mitigate this problem, which draws in the visual information to improve the original sequence self-attention.

The contributions of this paper are as follows:

- We propose a Boosted Transformer (BT) model for image captioning, which replaces the recurrent architecture with the self-attention mechanism. The proposed model makes full use of the semantic concepts and the visual features to improve the generated description.

- We design Concept-Guided Attention (CGA) to utilize the instance-level semantic concepts as auxiliary information for the boosted visual features.
- Vision-Guided Attention (VGA) is devised to compensate for the lack of visual information in sequence self-attention so that the model can produce more reasonable attention results for accurate and image-associated descriptions.
- Extensive experiments on the Microsoft COCO dataset validate the effectiveness of our proposed modules, which can further promote the performance of the transformer model. The proposed BT model significantly outperforms previous state-of-the-art methods.

The rest of the paper is organized as follows. In Section 2, some related works about image captioning and transformer are briefly introduced. In Section 3, we describe the overall architecture of the BT model and our proposed CGA and VGA modules in detail. In Section 4, we introduce the experiments validated on COCO datasets and some visualizations of experimental results for clear explanations. In Section 5, we make a concise conclusion for our method.

2. Related Work

A large number of methods has been proposed for image captioning since the emergence of deep learning [8]. The work in [9] proposed an encoder-decoder framework to generate a sentence given an image, where CNN was used as the image encoder to extract the image features and LSTM was used as the caption decoder to generate image descriptions word-by-word. Nevertheless, this method only fed a constant vector of representing image features into the beginning step of LSTM, and the subsequent steps did not use the image information. To solve this problem, the work in [10] proposed an attention-based caption model, which attended to different salient regions of the image dynamically at each step when generating the corresponding words, instead of feeding all image features to the caption decoder at the initial step. The work in [11] proposed adaptive attention, determining when to attend to an image or visual sentinel.

In addition to image features, semantic concepts are also utilized widely as another form of image representation. Some recent works [12–14] used semantic concepts as supplementary information to enhance semantic representations of visual features. The work in [15] proved the effectiveness of object concepts in image captioning and discussed five different architectures of the caption model. The work in [7,16] made full use of the broad semantic concepts, i.e., object, attribute, and relation, to construct a scene graph and then obtained a richer image representations by employing Graph Convolutional Network (GCN). Most of them regarded the visual features and semantic concepts as an ensemble body and then fed them into the LSTM decoder. Self-attention is beneficial to increase the global information for each region feature, so we propose a CGA module to process visual features and semantic concepts with two independent self-attention mechanisms and finally integrating them.

The recurrence-based caption models have been firmly established as state-of-the-art approaches and are popular in the sequence generation field. However, recurrent models have some limitations on parallel computation and gradient vanishing/exploding [17]. Recently, the work in [1] applied the transformer [6] decoder architecture to the task of image captioning and added multi-level supervision to improve the model performance. Transformer is a standard encoder-decoder framework proposed for neural machine translation and has advanced the state-of-the-art on various natural language processing tasks. This architecture uses the self-attention mechanism to compute hidden representations of two arbitrary positional inputs, avoiding the vanishing and exploding gradients. The transformer decoder utilizes the sequence self-attention with three identical word representations as queries, keys, and values to compute the attention map. However, this process ignores the affect of visual information and is easily influenced by inductive bias. In this paper, we stack a vision-guided attention mechanism to mitigate this problem.

3. Methodology

In this section, we first introduce some background about transformer in Section 3.1, which is the baseline model of our work. Then, we elaborate the overall architecture of our proposed model in Section 3.2. Finally, the proposed CGA and VGA modules are illustrated in Sections 3.3 and 3.4.

3.1. Transformer Background

Vanilla transformer [6] adopts the encoder-decoder architecture with each part stacking N identical layers, composed of the multi-head attention module and the feed-forward network module. Each module is coupled with the layer normalization and residual connection, which can simplify and steady the optimization process. Given that the transformer contains no recurrence or convolution, the sequential order is not leveraged and can notably affect the quality of generated descriptions. The work in [6] added an extra positional encoding after learning the word embedding to avoid this situation. The positional encoding is defined as:

$$PE_{(pos,2i)} = \sin(pos/10,000^{2i/d_{model}}) \tag{1}$$

$$PE_{(pos,2i+1)} = \cos(pos/10,000^{2i/d_{model}}) \tag{2}$$

where pos is the absolute position of the token in the caption and i is the dimension. Transformer uses scaled dot-product attention to deduce the correlation between queries Q and keys K and then obtains the weighted sum of the values V. Scaled dot-product attention is the basis of multi-head attention, and this computational process is shown as follows:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{3}$$

where the attention inputs are composed of queries Q, keys K, and values V with dimensions d_k, d_k, and d_v. $\frac{1}{\sqrt{d_k}}$ is used to counteract the effect of the large value of d_k pushing the softmax function into the regions of small gradients. Dot-product attention is faster and more space-efficient in practice because it can be implemented by parallel optimization [6].

Compared with single attention, multi-head attention can learn different representation subspaces at different positions. It contains h identical attention heads, and each head is a scaled dot-product attention, performing the attention function on queries, keys, and values independently. Finally, the h attention outputs are concatenated and projected back to the original dimension, producing the final values.

$$MultiHead(Q, K, V) = Concat(head_1, \ldots, head_h)W^O \tag{4}$$

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{5}$$

where $W^O \in \mathbb{R}^{hd_v \times d_{model}}$, $W_i^Q \in \mathbb{R}^{d_{model} \times d_k}$, $W_i^K \in \mathbb{R}^{d_{model} \times d_k}$, $W_i^V \in \mathbb{R}^{d_{model} \times d_v}$ are trainable projection matrices. To reduce the total computational cost, the work in [6] projected the original dimension of d_{model}=512 onto $d_k = d_v = d_{model}/h$ = 64 and h = 8.

The feed-forward network is another basic component. It is a two-layered fully-connected network with a ReLU activation function, which is applied to each position separately and can improve the nonlinear ability of the network.

$$FFN(x) = \max(0, xW_1 + b_1)W_2 + b_2 \tag{6}$$

where $W_1 \in \mathbb{R}^{d_{model} \times d_{ff}}$, $W_2 \in \mathbb{R}^{d_{ff} \times d_{model}}$ are trainable projection matrices and b_1, b_2 are the biases. In this work, d_{ff} = 2048 and d_{model} = 512.

3.2. Boosted Transformer

In this section, we introduce a boosted transformer to solve the problem of learning to generate a natural description given an image. We regard the combination of CNN, concept-guided attention, and the feed-forward network as the BT encoder to obtain the boosted visual features. As for the BT decoder, it is similar to transformer with the exception of adding a vision-guided attention module. The overall architecture of BT is shown in Figure 2.

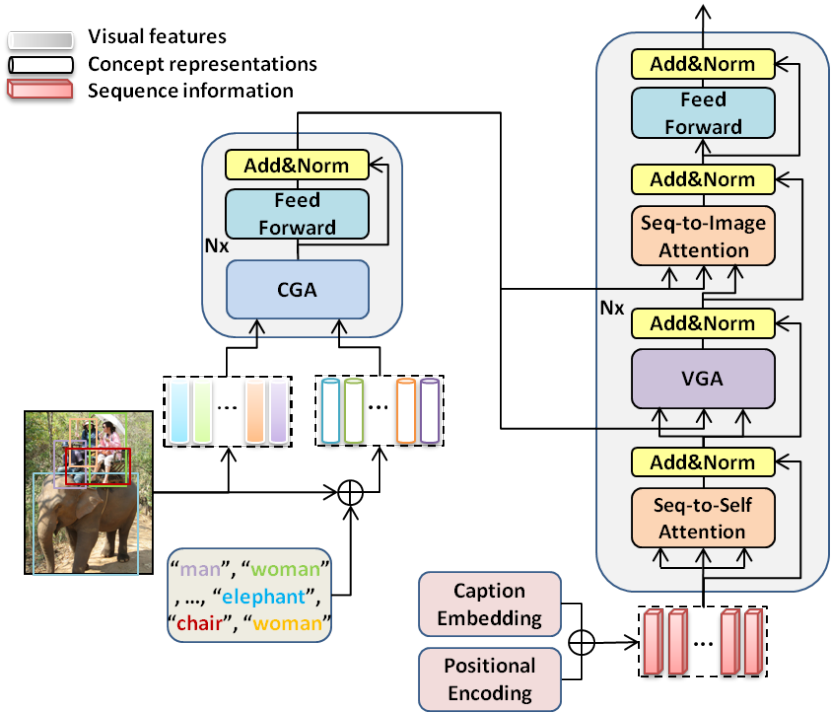

Figure 2. An illustration of boosted transformer for image captioning. The overall architecture of the model is a transformer-based encoder-decoder framework. Faster R-CNN is first leveraged to detect a set of salient image regions and then to obtain the aligned visual features and semantic concepts. Next, a concept-guided attention module composed of two self-attention mechanisms and an integration module is used to get the boosted visual features. After that, The boosted visual features and the sequence representations are exploited by the caption decoder, which contains a sequence self-attention, vision-guided attention, sequence-to-image attention, and a feed-forward network to generate a caption. seq, sequence.

Image encoder: Traditional image captioning uses a CNN model pre-trained on the image classification task (e.g., Resnet101 [18]) as the feature extractor, but recent research showed that an object-based feature extractor can obtain more accurate information about the specific image regions. In this paper, we acquire the visual features of the image through the latter approach as the way adopted by faster RCNN [19]. For simplicity, given an image I, we can get image features V and semantic concepts Y, where V = $\{v_1, v_2, ..., v_n\} \in \mathbb{R}^{n \times 2048}$, Y = $\{y_1, y_2, ..., y_n\} \in \mathbb{R}^n$ indicate the n regional features and the corresponding semantic concepts extracted by the detection model. Especially, we adopt the linear-projected visual features as the visual features for the coherence with the model dimension in the following sections, where $V \in \mathbb{R}^{n \times d_{model}}$. After that, CGA is used to contextualize the visual features and the semantic concepts for the boosted visual features as elaborated in Section 3.3.

Caption decoder: Given a caption, word embedding [20] is first used to project each token into the word vector, forming a sequence matrix $E^{cap}(Y) = \{w_1, w_2, ...w_L\} \in \mathbb{R}^{L \times d_{model}}$, where L is the length of the caption and word embedding $E^{cap} \in \mathbb{R}^{vocab \times d_{model}}$ is a trainable matrix. Then, we append the position information to sequence matrix $E^{cap}(Y)$ by the positional encoding as Equations (1) and (2),

obtaining the final sequence representations $S \in \mathbb{R}^{L \times d_{model}}$. The BT decoder contains a feed-forward network and three kinds of attention modules, i.e., (1) sequence self-attention, (2) VGA, and (3) sequence-to-image attention. Sequence self-attention has three identical inputs, in which queries, keys, and values are all sequence representations. This attention mechanism not only allows each positional vector to reflect itself, but also attends to all the other positional vectors in the sequence, promoting the contextual information only with the sequence representations. Sequence-to-image attention imitates the distinctive attention mechanism [21] in which the queries come from the sequence information of the decoder, while the keys and values are from visual features. This allows every positional word representation to attend to all the regional features, so that different regional features can be treated discriminately for the final prediction. As for the VGA module, we elaborate it in Section 3.4.

3.3. Concept-Guided Attention for the Encoder

The proposed concept-guided attention is an integrated module containing two kind of inputs as shown in Figure 3a, used to develop the mutual effectiveness between visual features and semantic concepts for image captioning. It is the detail representation of the boosted transformer encoder layer in Figure 2. To obtain a concept representation associated with image content, we concatenate visual features V^i and embedding vectors of semantic concepts $E^{se}(Y)$ in the channel dimension and then project the results to hidden dimension d_{model} with a nonlinear function ϕ_o. Residual connection from the visual features to the output of projection helps to keep the visual peculiarity. After that, the concept representations C and visual features V^i are contextualized by two independent self-attention modules and then integrated into boosted visual features.

$$C = \phi_o([V^i; E^{se}(Y)]) + V^i \tag{7}$$

$$V^{i+1} = FFN(CGA(V^i, C)) \tag{8}$$

where $E^{se} \in \mathbb{R}^{vocab \times d_o}$ is another word embedding for projecting the semantic concepts into the vector, independent of caption word embedding E^{cap}. The concept vocabulary is 472 and $d_o = 128$ for this smaller vocabulary, compared with the caption vocabulary of 9488 words. ϕ_o is a nonlinear projection layer and implemented as the fully-connected network with a ReLU activation function and dropout. V^i is the visual features inputs of the i^{th} layer of the image encoder. V^0 represents the initial features extracted from the image, and V^{i+1} is the boosted visual features integrated by V^i and C. The outputs of the encoder layer V^{i+1} are regarded as the next encoder layer's inputs. FFN is the feed-forward network as shown in Equation (6).

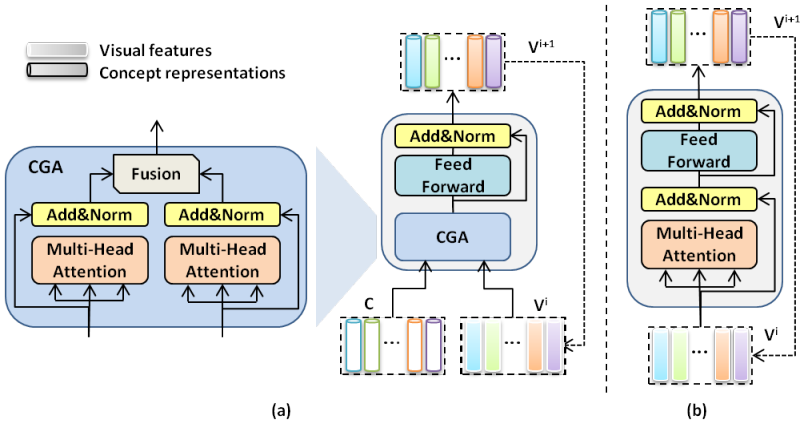

Figure 3. The overview of the BT encoder. Our proposed image encoder adopts a flexible architecture, which can decide whether to use the concept representations. (**a**) is an encoder layer with the visual features and the concept representations as inputs. (**b**) only uses the visual features as the inputs. The outputs of either (**a**) or (**b**) serve as the next layer's visual feature inputs.

Here, we devise two fusion strategies to integrate the different representations, e.g., gated sum and channel connect. For the gated sum strategy, we set a learnable parameter as the weight value, as shown in Equation (9).

$$CGA(V^i, C) = \lambda \times MultiHead(V^i, V^i, V^i) + (1 - \lambda) \times MultiHead(C, C, C) \tag{9}$$

For the channel connect strategy, we concatenate the visual features and concept representations in the channel dimension and then reduce it to the model hidden dimension with a linear matrix, as shown in Equation (10).

$$CGA(V^i, C) = W\left(\left[MultiHead(V^i, V^i, V^i); MultiHead(C, C, C)\right]\right) \tag{10}$$

where $W \in \mathbb{R}^{(d_{model}+d_o)\times d_{model}}$, and λ is learnable with the training process. Concept representations C are obtained by Equation (7).

3.4. Vision-Guided Attention for the Decoder

Given that sequence self-attention can be easily influenced by the language prior, we propose an intuitive attention module, VGA, which considers the impact of visual information and can help to generate more image-related descriptions.

We use an example in Figure 4 to show the intuition behind the VGA module. When we obtain a partial sequence "woman standing on" and generate the next word "beach", the relevant regions in the image will be activated as shown in the green box in Figure 4b. Those activated areas not only cover the beach, but also the partial skirt, with the hand in a way. Therefore, it should be supposed that the words such as "woman" and "on" are more important to this word generation. The obtained contextual vector by this mechanism can have more reliable cues to point at the generated word. However, VGA cannot replace the sequence self-attention because VGA has strengths in generating visual words, whereas the linguistic words [11], such as "of" and "the" are more dependent on the latter. Therefore, we stack a VGA module following the sequence self-attention so that each token in the caption can obtain a more reasonable attention distribution, relying on both word representations' similarity and the visual relationship.

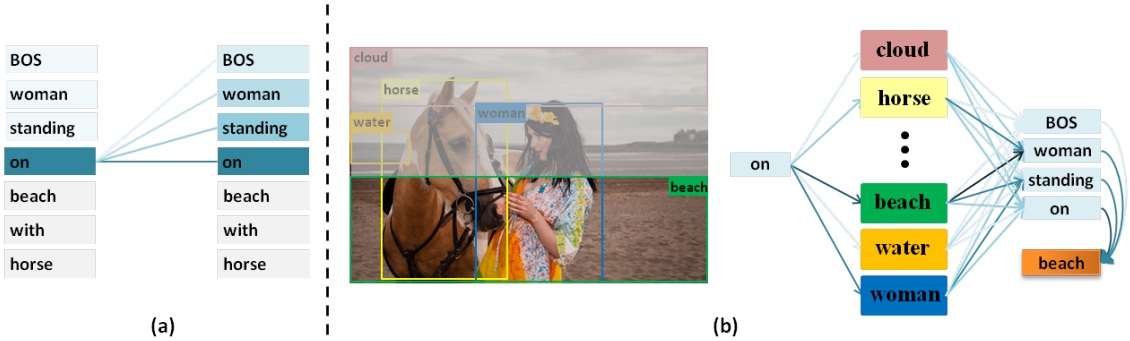

Figure 4. The comparison between (**a**) sequence self-attention and (**b**) vision-guided attention. Sequence self-attention computes the correlation between different tokens directly using the word representations. Vision-guided attention computes the correlation between target tokens, and other tokens are indirectly bridged through visual information. "BOS" represents Begin token Of Sentence. The same colorful blocks in (**b**) represent the corresponding visual features.

VGA is substantially a double-attention process, as shown in Figure 5a, which decomposes the seq-to-self attention map into the seq-to-image attention map and the image-to-seq attention map. On the one hand, given a set of sequence information $S^i = \{s_1^i, s_2^i, ..., s_k^i\}$ and visual information $V = \{v_1, v_2, ..., v_n\}$, the seq-to-image attention map $M_{s2i} \in \mathbb{R}^{k\times n}$ is first built upon the scaled

dot-product, representing the correlation between S and V, which can be interpreted as the relative importance about different visual regions when generating word i. On the other hand, we use the image-to-seq attention map $M_{i2s} \in \mathbb{R}^{n \times k}$ to represent the correlation between visual information and sequence information. M_{i2s} can be obtained by the same process as M_{s2i} or transposing M_{s2i}; here, we adopt $M_{i2s} = M_{s2i}^T$ to reduce the computation. Given that only the previously-generated (before time t) sequence information can be utilized, we overlay the time mask on M_{i2s} to obtain a time-dependent attention map $\widetilde{M}_{i2s} \in \mathbb{R}^{k \times n \times k}$, as shown in Figure 5b. It has a similar motivation to vanilla transformer [6], implementing this by masking out (setting to $-\infty$) the corresponding values in the input of the softmax function, as shown in Figure 5c, representing illegal connections. Finally, we reshape $M_{s2i} \in \mathbb{R}^{k \times n}$ into $\widetilde{M}_{s2i} \in \mathbb{R}^{k \times 1 \times n}$ and then apply this sequence-image attention map $\widetilde{M}_{s2i}$ to the time-dependent attention map $\widetilde{M}_{i2s}$, obtaining the final boosted seq-to-self attention map $M_{s2s} \in \mathbb{R}^{n \times n}$.

$$M_{s2i} = \frac{S^i V^T}{\sqrt{d_k}} \tag{11}$$

$$\widetilde{M}_{i2s} = softmax\left(\psi\left(M_{s2i}^T\right)\right) \tag{12}$$

$$\widetilde{M}_{s2i} = softmax\left(reshape(M_{s2i})\right) \tag{13}$$

$$M_{s2s} = \widetilde{M}_{s2i}\widetilde{M}_{i2s} \tag{14}$$

$$VGA(S^i, V, S^i) = M_{s2s}S^i \tag{15}$$

where V is obtained by Equation (8), S^i is the sequence self-attention output of the i^{th} decoder layer, $softmax$ is a activated function for normalization, and ψ represents the time mask operation, as shown in Figure 5c. Reshape is an operation of changing the dimension. $VGA(S^i, V, S^i)$ is the output of the VGA module.

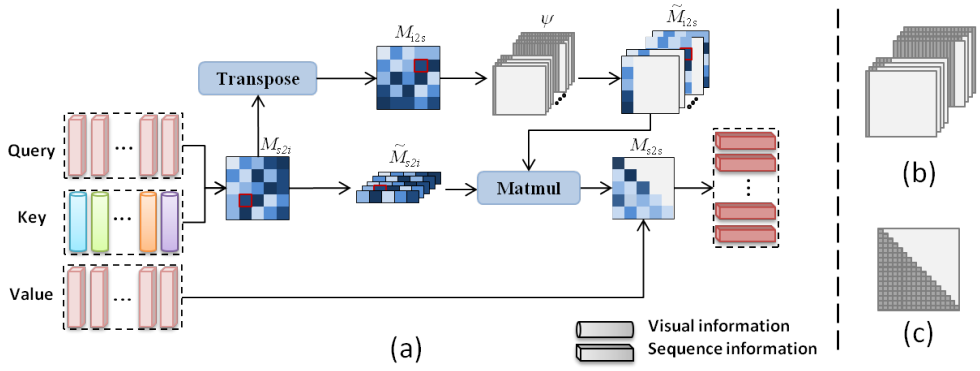

Figure 5. (**a**) The completed computational process of Vision-Guided Attention (VGA). (**b**) "Time mask" adjusts the image-to-seq attention map dynamically over time to keep the view of visual features within the time limitation. (**c**) The lower triangular masked matrix ensures only the previously-generated sequence information can be utilized, and this masked matrix is used in sequence self-attention. The dark parts of the masks mean retaining status, and the others are set to $-\infty$.

4. Experiments

4.1. Datasets

We used the COCO [22] dataset as the evaluation benchmark to measure the performance of our proposed module, which is the largest benchmark dataset for image captioning. This dataset contains 82,783, 40,504, and 40,775 images for training, validation, and testing, respectively, with 5 human-produced captions for each image approximately. To make the results comparable with others fairly, we used the publicly available "Karpathy" splits [23] following previous works. These

splits contain 5000 validation images and 5000 test images taken from the original validation set. The remaining validation set and the whole original training set, 113,287 images, were used for training. We followed the standard preprocessing procedure on all captions: transforming all sentences to lowercase and truncating to less than 16 words. The words occurring less than five times in the training captions were filtered and replaced by a special token 'UNK'. Finally, we built a vocabulary containing 9487 words. Then, we evaluated the performance of our caption model in several different evaluation metrics, including BLUE [24], ROUGEL [25], CIDEr [26], METEOR [27], and SPICE [28], which have different emphases on the evaluation of generated captions quantitatively and are denoted as B@N (N=1, 2, 3, 4), R, C, M, and S, respectively.

4.2. Experiments Details

In order to avoid the effects whereby different backbones can extract discrepant image features, we obtained the same visual features as [29], including 10–100 regional features with a dimension of 2048 for each image. For the aim of speeding up and reducing memory, our image encoder and caption decoder stacked 4 layers with 8 attention heads. The hidden unit dimension of multi-head attention was 512, and that of the feed-forward was 2048. We used an independent word embedding to represent the semantic concepts with an embedding size of 128. In the training stage, we first trained our model with the cross-entropy loss for 15 epochs. After that, the pre-trained model continued to adjust its parameters under the proposed Reinforcement Learning (RL) method [30] for another 10 epochs. The RL-based method was optimized for the CIDEr metric, but it has been proven that this process can improve other metrics incidentally. We used the Adam optimizer [31] with a momentum parameter of 0.9. The initial learning rate was set to $min(1t^{-4}, 3e^{-4})$ for 6 epochs and then decayed by 0.5 after every 3 epochs, where t is the current epoch and starts from 1. The scheduled sampling probability [32] was initialized as 0.05 and multiplied by 2 to the maximal value 0.25 after every 2 epochs. When testing, beam search with a beam size of 3 was used.

4.3. Ablation Studies

In this section, we conduct extensive ablative experiments for the BT model on the MSCOCO datasets. We first explore the CGA's influence with different structures, i.e., input types, layer choices, and integration strategies. Then, we validate the effectiveness of VGA with different layer choices. The baseline is an implementation of vanilla transformer [6].

4.3.1. Influence of CGA

Single source input: Given an image, we can obtain the projected visual features V and the aligned semantic concepts Y as Section 3.3. A naive way to verify the mutual effectiveness between them is solely using the single representations as the input with the comparison of both. As shown in Table 1, the baseline only used the visual features V as the input of the model. CGA-c represents that the model only uses the aligned concept representations (c) computed by Equation (7) as the input. CGA-y means only regarding the embedding vectors of concepts as the input without associating with visual features. We can see that individual visual features, aligned concept representations, and concept vectors were not enough to express the semantic information of the images compared with the integrated model CGA_1st-GS, which considers the visual features and the concept representations . In order to exclude the effect of increasing the parameter, we conducted a comparison experiment CGA_1st-GS(vv), whose inputs in two branches were identical visual features. Comparing CGA_1st-GS(vv) and the baseline, the result showed that the architecture of the CGA module could also further develop the image information, similar to multi-head attention.

Layer choice: Concept representations can enhance the semantic expression of visual features effectively, but this does not mean that the more times concept information is utilized, the better our model. We explore which encoder layers should use CGA in Table 1(c) and find that equipping all the encoder layers with CGA, i.e., CGA_4-GS, resulted in the degradation of performance because

of overfitting. Moreover, equipping the first layer with CGA, i.e., CGA_1st-GS, can achieve the best performance in the evaluation metrics. CGA_xth means that we only apply the CGA module to the xth layer of the image encoder, and the other layers are the same as the baseline, as shown in Figure 3b.

Integration strategy: We show the performance comparison between two integration strategies in Table 1, (c) Gated Sum (GS) and (d) Channel Connect (CC). We can see that the gated sum strategy achieved better performance with all metrics since the image representations and concept representations were already aligned.

Table 1. The results of different ways of using the CGA module, evaluated on the COCO Karpathy test split. B@N, BLUE; M, METEOR; R, ROUGEL; C, CIDEr; S, SPICE; CGA-c, Concept-Guided Attention with aligned concept representations; CGA-y, Concept-Guided Attention with concept embedding vectors; CGA_1st-GS(vv), Concept-Guided Attention with two identical visual features; GS, Gated Sum; CC, Channel Connect.

	Model	B@1	B@2	B@3	B@4	M	R	C	S
(a)	baseline	80.2	64.9	50.5	38.4	28.6	58.2	128.1	22.6
(b)	CGA-c	79.8	64.3	49.8	37.9	28.6	58.1	126.6	22.5
	CGA-y	75.4	58.5	43.5	31.9	25.6	53.9	106.3	19.4
	CGA_1st-GS(vv)	80.4	65.2	51.0	38.6	28.6	58.5	128.7	22.7
(c)	CGA_1st-GS	**81.0**	**65.9**	**51.5**	**39.5**	**29.3**	58.9	**130.9**	**23.1**
	CGA_2nd-GS	80.9	65.7	51.3	39.3	29.0	58.8	130.5	22.9
	CGA_3rd-GS	80.9	65.7	51.3	39.4	29.1	**59.0**	130.3	23.0
	CGA_4th-GS	80.7	65.4	51.2	39.2	29.1	58.9	130.3	22.8
	CGA_4-GS	80.1	64.8	50.3	38.3	28.8	58.2	127.4	22.6
(d)	CGA_1st-CC	80.6	65.2	50.6	38.5	28.8	58.5	128.5	22.7

4.3.2. Influence of VGA

From Table 2, we can see that with VGA (b), all the metrics obtained improvement to some extent. VGA_xth had the similar means as that in Table 1. VGA re-attended to all the positional sequence information with the guidance of visual information, achieving a more reasonable attention distribution. The overuse of VGA can result in overfitting, so we only applied VGA to the fourth layer, which can obtain the best performance according to the ablative experiments. Some visualizations are shown in Section 4.5, which is better to reflect the effectiveness of VGA. In order to validate how much visual information VGA involves, we got rid of the sequence-to-image attention module, which was the sole way to involve visual information. From the results in (c) and (d), VGA utilized the image information in another way and could work well with the sequence information, where (c) s2s-VGA means the decoder layer was composed of sequence self-attention and VGA. (d) s2s means the decoder layer only contained sequence self-attention without the guidance of visual features.

Table 2. The results of different ways of using the VGA module, evaluated on the COCO Karpathy test split. s2s, sequence self-attention.

	Model	B@1	B@2	B@3	B@4	M	R	C	S
(a)	baseline	80.2	64.9	50.5	38.4	28.6	58.2	128.1	22.6
(b)	VGA_1st	80.8	65.5	51.1	39.1	29.0	58.8	130.3	22.9
	VGA_2nd	80.8	65.7	51.4	39.4	29.1	58.9	130.7	22.9
	VGA_3rd	81.0	65.7	51.3	39.3	29.2	59.1	130.8	23.1
	VGA_4th	**81.1**	**65.9**	**51.4**	**39.4**	**29.3**	**58.9**	**131.0**	**23.1**
	VGA_4	80.8	65.5	51.1	39.2	29.0	58.8	129.8	22.9
(c)	s2s-VGA	79.5	63.9	49.6	37.8	28.4	57.9	124.4	21.8
(d)	s2s	44.5	24.8	14.1	8.1	13.3	34.8	11.0	4.1

4.4. Comparing with the State-of-the-Art

In this section, we regard BT as our overall model for convenience, which was based on transformer and comprised of a CGA module in the first encoder layer and a VGA module in the fourth decoder layer. We conducted the test evaluations on the offline "Karpathy" split (5000 images) and the online MSCOCO test server (40,775 images), which have been widely adopted in prior works. We compared BT with the state-of-the-art methods, e.g., Google NICv2 [9], AdaATT [11], SCST [30], StackCap [33], Up-Down [29], DA [34], SGAE [7], and so on, as shown in Table 3 for the offline test and Table 4 for the online test. Among these works, SCST used the reinforcement learning for training; Up-Down applied the object-level features to the model; DA used a two-layer LSTM to refine the prediction results; SGAE utilized the object, attribute, and relation to construct a semantic graph, helping to learn a richer contextual image representation. It is clear that our model performed better on the evaluation metrics.

Table 3. The performance comparison with the state-of-the-art methods on the COCO Karpathy test split.

Model	B@1	B@4	M	R	C	S
Google NICv2 [9]	-	32.1	25.7	-	99.8	-
AdaATT [11]	74.2	33.2	26.6	-	108.5	-
Att2all [30]	-	34.2	26.7	55.7	114.0	-
StackCap [33]	78.6	36.1	27.4	56.9	120.4	20.9
Up-Down [29]	79.8	36.3	27.7	56.9	120.1	21.4
GCN-LSTM [16]	80.5	38.2	28.5	58.3	127.6	22.0
DA [34]	79.9	37.5	28.5	58.2	125.6	22.3
SGAE [7]	80.8	38.4	28.4	58.6	127.8	22.1
CGA_1st-GS	81.0	39.5	29.3	58.9	130.9	23.1
VGA_4th	81.1	39.4	29.3	58.9	131.0	23.1
BT	**81.2**	**39.7**	**29.4**	**59.1**	**131.5**	**23.2**

Table 4. Leaderboard of the published state-of-the-art image captioning models on the online COCO test server, where c5 and c40 denote using 5 and 40 references for testing, respectively.

Model	BLEU -1		BLEU-2		BLEU-3		BLEU-4		METEOR		ROUGE-L		CIDEr	
Metric	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40
SCST [30]	78.1	93.7	61.9	86.0	47.0	75.9	35.2	64.5	27.0	35.5	56.3	10.7	114.7	116.0
LSTM-A [15]	78.7	93.7	62.7	86.7	47.6	76.5	35.6	65.2	27.0	35.4	56.4	70.5	116.0	118.0
StackCap [33]	77.8	93.2	61.6	86.1	46.8	76.0	34.9	64.6	27.0	35.6	56.2	70.6	114.8	118.3
Up-Down [29]	80.2	95.2	64.1	88.8	49.1	79.4	36.9	68.5	27.6	36.7	57.1	72.4	117.9	120.5
CAVP [35]	80.1	94.9	64.7	88.8	50.0	79.7	37.9	69.0	28.1	37.0	58.2	73.1	121.6	123.8
DA [34]	79.4	94.4	63.5	88.0	48.7	78.4	36.8	67.4	28.2	37.0	57.7	72.2	120.5	122.0
SGAE [7]	80.6	95.0	65.0	88.9	50.1	79.6	37.8	68.7	28.1	37.0	58.2	73.1	122.7	125.5
BT	80.5	94.7	**65.2**	88.8	**50.6**	**80.1**	**38.6**	**69.6**	**28.8**	**37.9**	**58.5**	**73.5**	**125.0**	**126.8**

4.5. Qualitative Analysis

In order to further demonstrate the effectiveness of BT, we show some visualizations of the attention map, comparing the attention map changes with VGA, as shown in Figure 6. The result revealed that VGA can effectively attend to some new focuses digressing from the language prior. For example, in Fig ATT_s2s, the attention map was obtained by sequence self-attention, where the color of the position corresponding to "standing" and "two" or "on" and "two" are relatively dark, indicating higher focus. This is because the situation that those words co-appeared in happened frequently in the training set. After the processing of VGA, the attention distribution was changed and became more reasonable, as shown in ATT_VGA, where "standing" had a higher focus on "people". ATT_s2i is the attention map between caption tokens and the visual areas. In addition,

we present some examples of images and captions in Figure 7. We can see that the generated descriptions of BT contained more image-associated tokens compared with the baseline.

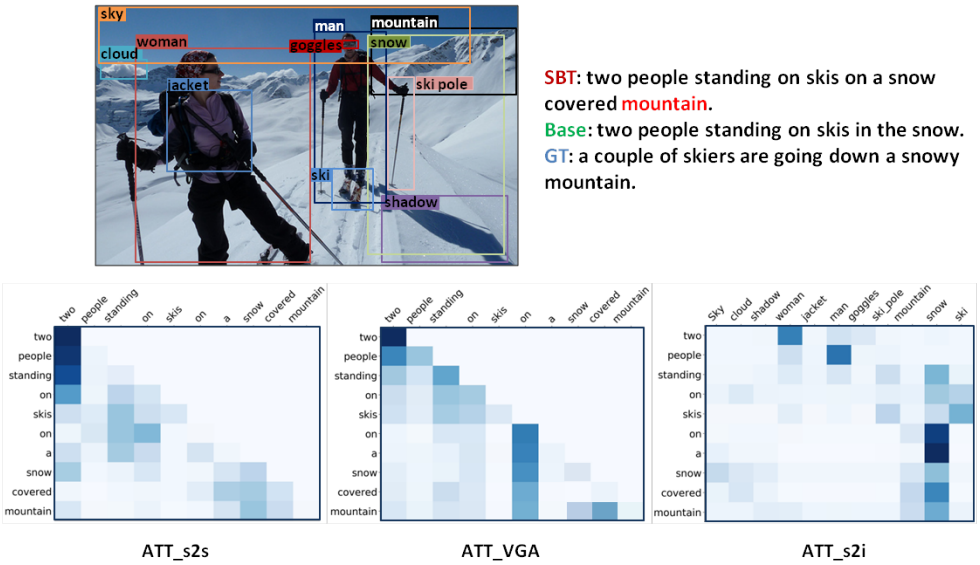

Figure 6. Visualizations of the attention maps, including the seq-to-self attention map, i.e., ATT_s2s, the boosted seq-to-self attention map, i.e., ATT_VGA, and the seq-to-image attention map, i.e., ATT_s2i. For a better visualization effect, we only highlight some more caption-related objects (10-100 objects in total, dependent on the specific image).

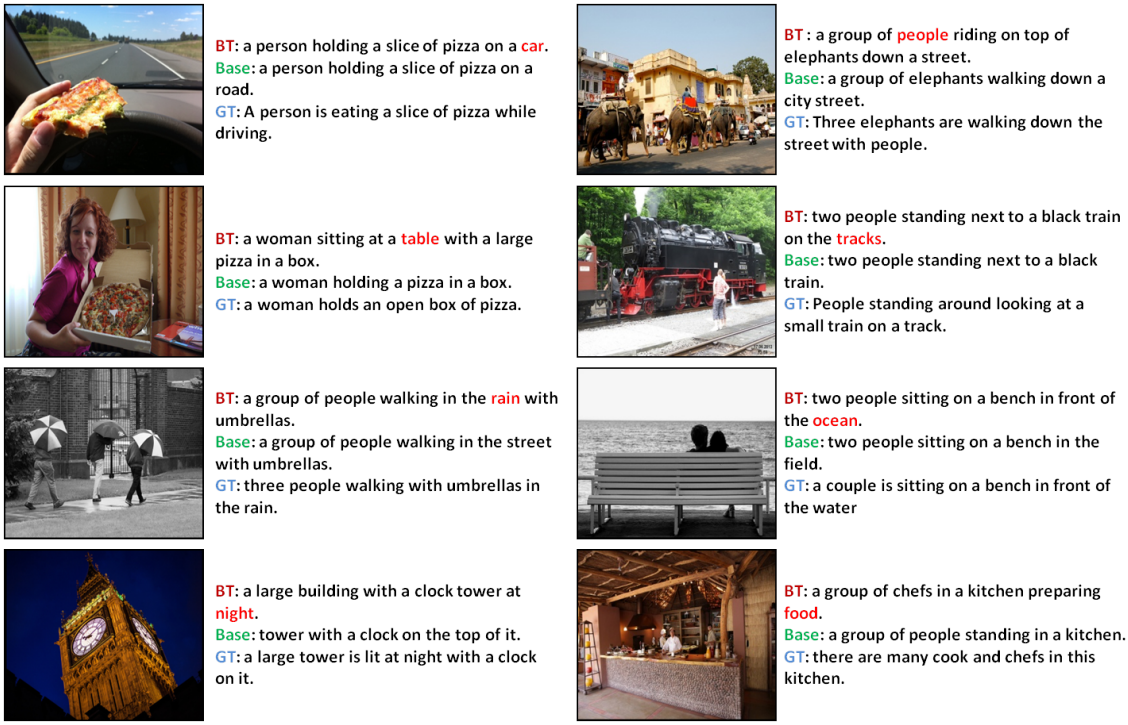

Figure 7. Examples generated by the BT model on the Microsoft COCOvalidation set. GT is the ground-truth chosen from one of five references. Base and BT represent the descriptions generated from the vanilla transformer and our model, respectively. The red words reflect that our model can generate more image-associated descriptions.

5. Conclusions

In this paper, we presented a novel model that makes full use of semantic and visual information for more accurate and image-associated captioning generation. We devised a semantic boost module in the encoder, i.e., CGA, and a visual boost module in the decoder, i.e., VGA. The former utilized the instance-level semantic concepts to boost the visual features, and the latter achieved the double attention under the influence of the visual information. With these innovations, we obtained a better performance than the state-of-the-art approaches on the MSCOCO benchmark.

Author Contributions: Conceptualization, J.L., P.Y., and L.G.; data curation, P.Y.; formal analysis, J.L. and W.Z.; funding acquisition, J.L.; methodology, P.Y.; project administration, J.L. and W.Z.; software, P.Y.; supervision, J.L.; validation, W.Z.; visualization, P.Y. and L.G.; writing, original draft, P.Y.; writing, review and editing, J.L. and L.G.; all authors interpreted the results and revised the paper.

Acknowledgments: This work was supported in part by the National Nature Science Foundation of China No. 61671054 and in part by Beijing Natural Science Foundation No. 4182038.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Zhu, X.; Li, L.; Jing, L.; Peng, H.; Niu, X. Captioning Transformer with Stacked Attention Modules. *Appl. Sci.* **2018**, *8*, 739. [CrossRef]
2. Zhu, X.; Li, L.; Liu, J.; Guo, L.; Fang, Z.; Peng, H.; Niu, X. Image Captioning with Word Gate and Adaptive Self-Critical Learning. *Appl. Sci.* **2018**, *8*, 909. [CrossRef]
3. Yang, Z.; Yuan, Y.; Wu, Y.; Cohen, W.W.; Salakhutdinov, R.R. Review networks for caption generation. In Proceedings of the Advances in Neural Information Processing Systems 29 (NIPS 2016), Barcelona, Spain, 5–10 December 2016; pp. 2361–2369.
4. Socher, R.; Karpathy, A.; Le, Q.V.; Manning, C.D.; Ng, A.Y. Grounded compositional semantics for finding and describing images with sentences. *Trans. Assoc. Comput. Linguist.* **2014**, *2*, 207–218. [CrossRef]
5. Guan, Z.; Liu, K.; Ma, Y.; Qian, X.; Ji, T. Middle-Level Attribute-Based Language Retouching for Image Caption Generation. *Appl. Sci.* **2018**, *8*, 1850. [CrossRef]
6. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*; Neural Information Processing Systems Foundation, Inc.: Long Beach, CA, USA, 2017; pp. 5998–6008.
7. Yang, X.; Tang, K.; Zhang, H.; Cai, J. Auto-encoding scene graphs for image captioning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 10685–10694.
8. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436. [CrossRef] [PubMed]
9. Vinyals, O.; Toshev, A.; Bengio, S.; Erhan, D. Show and Tell: Lessons learned from the 2015 MSCOCO Image Captioning Challenge. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 652–663. [CrossRef] [PubMed]
10. Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhutdinov, R.; Zemel, R.; Bengio, Y. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In Proceedings of the International Conference on Machine Learning, Lille, France, 6–11 July 2015; pp. 2048–2057.
11. Lu, J.; Xiong, C.; Parikh, D.; Socher, R. Knowing when to look: Adaptive attention via a visual sentinel for image captioning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 375–383.
12. Fang, H.; Gupta, S.; Iandola, F.; Srivastava, R.K.; Deng, L.; Dollár, P.; Gao, J.; He, X.; Mitchell, M.; Platt, J.C.; et al. From captions to visual concepts and back. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1473–1482.
13. You, Q.; Jin, H.; Wang, Z.; Fang, C.; Luo, J. Image captioning with semantic attention. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 4651–4659.

14. Lu, J.; Yang, J.; Batra, D.; Parikh, D. Neural baby talk. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7219–7228.
15. Yao, T.; Pan, Y.; Li, Y.; Qiu, Z.; Mei, T. Boosting image captioning with attributes. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 4894–4902.
16. Yao, T.; Pan, Y.; Li, Y.; Mei, T. Exploring visual relationship for image captioning. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 684–699.
17. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
18. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
19. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In Proceedings of the Advances in Neural Information Processing Systems 28 (NIPS 2015), Montreal, QC, Canada, 7–12 December 2015; pp. 91–99.
20. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. *arXiv* **2013**, arXiv:1301.3781.
21. Chaudhari, S.; Polatkan, G.; Ramanath, R.; Mithal, V. An Attentive Survey of Attention Models. *arXiv* **2019**, arXiv:1904.02874.
22. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In Proceedings of the European Conference on Computer Vision; Zurich, Switzerland, 6–12 September 2014; pp. 740–755.
23. Karpathy, A.; Fei-Fei, L. Deep visual-semantic alignments for generating image descriptions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3128–3137.
24. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.J. BLEU: A method for automatic evaluation of machine translation. In Proceedings of the 40th Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, Philadelphia, PA, USA, 7–12 July 2002; pp. 311–318.
25. Flick, C. ROUGE: A Package for Automatic Evaluation of summaries. In Proceedings of the Workshop on Text Summarization Branches Out, Barcelona, Spain, 25–26 July 2004.
26. Vedantam, R.; Zitnick, C.L.; Parikh, D. CIDEr: Consensus-based Image Description Evaluation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015.
27. Lavie, A.; Agarwal, A. METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments. In Proceedings of the Second Workshop on Statistical Machine Translation. Association for Computational Linguistics, Prague, Czech Republic, 23 June 2007; pp. 228–231.
28. Anderson, P.; Fernando, B.; Johnson, M.; Gould, S. SPICE: Semantic Propositional Image Caption Evaluation. *Adapt. Behav.* **2016**, *11*, 382–398.
29. Anderson, P.; He, X.; Buehler, C.; Teney, D.; Johnson, M.; Gould, S.; Zhang, L. Bottom-up and top-down attention for image captioning and visual question answering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6077–6086.
30. Rennie, S.J.; Marcheret, E.; Mroueh, Y.; Ross, J.; Goel, V. Self-critical sequence training for image captioning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7008–7024.
31. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
32. Bengio, S.; Vinyals, O.; Jaitly, N.; Shazeer, N. Scheduled Sampling for Sequence Prediction with Recurrent Neural Networks. In Proceedings of the International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015.
33. Gu, J.; Cai, J.; Wang, G.; Chen, T. Stack-captioning: Coarse-to-fine learning for image captioning. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018.

34. Gao, L.; Fan, K.; Song, J.; Liu, X.; Xu, X.; Shen, H.T. Deliberate Attention Networks for Image Captioning. In Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019.
35. Liu, D.; Zha, Z.J.; Zhang, H.; Zhang, Y.; Wu, F. Context-aware visual policy network for sequence-level image captioning. *arXiv* **2018**, arXiv:1808.05864.