IEICE TRANSACTIONS

on Information and Systems

DOI:10.1587/transinf.2024EDP7036

Publicized:2024/10/02

This advance publication article will be replaced by the finalized version after proofreading.

A PUBLICATION OF THE INFORMATION AND SYSTEMS SOCIETY

The Institute of Electronics, Information and Communication Engineers

Kikai-Shinko-Kaikan Bldg., 5-8, Shibakoen 3 chome, Minato-ku, TOKYO, 105-0011 JAPAN

PAPER

UTStyleCap4K: Generating Image Captions with Sentimental Styles

Chi ZHANG[†a)], **Li TAO**[†b)], ***Nonmembers***, ***and*** **Toshihiko YAMASAKI**[†c)], ***Member***

SUMMARY Stylized image captioning is the task of generating image captions that have a description style, such as positive or negative sentiments. Recently, deep learning models have reached high performance in this task, but they still lack description accuracy and diversity, and they often suffer from the small size and the low descriptiveness of existing datasets. In this paper, we introduce a new dataset, UTStyleCap4K, which contains 4,644 images with three positive and three negative captions for every image (27,864 captions in total), collected by a crowdsourcing service. Experimental results show that our dataset is accurate in meaning and sentiments, diverse in the ways to describe the styles, and less similar to the base dataset, the MSCOCO dataset, than existing stylized image captioning datasets. We train multiple models on our dataset to set a baseline. We also propose a new Bidirectional Encoder Representations from Transformers (BERT) based model, StyleCapBERT, that controls the length and style of the generated captions at the same time, by introducing length and style information into the embeddings of caption words. Experimental results show that our model is capable of generating captions of three sentimental styles, positive, factual, and negative, at the same time, and achieving the best performance on our dataset.

key words: *stylized image captioning, dataset, BERT*

1. Introduction

Image captioning aims at generating natural language descriptions for images. It is an important task in artificial intelligence, asking for the ability to understand image contents and transfer such understandings to natural language descriptions. Since deep learning was introduced, a lot of deep learning based image captioning models have been proposed [1]–[3], reaching high performance in terms of accuracy in descriptiveness and correctness in grammar and syntax.

While the expression power of image captioning models keeps increasing, researchers turn to evaluation metrics besides accuracy or correctness [4]. Human beings are capable of telling different stories given the same image, based on their subjective emotions. Such an ability to control descriptions is something artificial intelligence research wants to achieve.

Based on these needs, the task of stylized image captioning [5] was proposed. While traditional image captioning models aim to describe images in a plain and direct manner, stylized image captioning seeks to generate captions with additional styles, such as incorporating positive or negative sentiments or telling romantic or adventurous stories. Mathews et al. [5] propose the first paper to introduce the task of stylized image captioning, together with the most important dataset in stylized image captioning, SentiCap. Gan et al. [6] propose another dataset, FlickrStyle10K, which introduces another two styles, humorous and romantic.

- A person is peacefully riding a bike across a brick road on a sunny day.
- An old man riding a bike on a gorgeous, bricked road.
- A happy person riding their bike down a city street center as people walk in the background.
- A clumsy fool rides a bike down an ugly brown road.
- a person rides a bike on a dirty city street
- An older women cautiously rides her bike on a brick city road.

Fig. 1: **One example from our dataset UTStyleCap4K.** For every image, we annotate three positive and three negative captions. Here we use the red color to denote words related to positive styles and the blue color to denote words related to negative styles.

A lot of stylized image captioning models have been proposed [7]–[10] using these two datasets. While they achieve good results, their performance is significantly lower than traditional image captioning models [2], [11] on conventional datasets like MSCOCO [12] in terms of evaluation metric scores. One reason for this deficiency we believe is the low quality of captions in existing stylized image captioning datasets. For example, SentiCap contains many unreasonable captions such as "a dead man doing a trick on a skateboard on a sidewalk", where "dead man" conveys negative sentiments but contradicts the facts. Some captions are wrongly annotated, such as "man doing a clever trick on a skateboard on a sidewalk", which is annotated negative. The size of the dataset is also quite small, with only 8,869 captions in total, only about 1% of the size of the MSCOCO dataset [12].

Owing to such facts, we construct a new dataset, UTStyleCap4K, as shown in Figure 1. UTStyleCap4K contains 4,644 images with three positive and three negative captions for each, totaling 27,864 stylized captions, about three times the size of SentiCap. The statistics of our dataset also show that it is more diverse in the ways to describe the sentiments, and less similar to its base dataset, the MSCOCO dataset, than SentiCap. Therefore, we believe that our dataset is better than SentiCap in terms of quantity and quality.

With the revolution of BERT [13] in natural language processing, standard image captioning models also turn to

[†]The University of Tokyo
a) E-mail: zhangchi@cvm.t.u-tokyo.ac.jp
b) E-mail: taoli@cvm.t.u-tokyo.ac.jp
c) E-mail: yamasaki@cvm.t.u-tokyo.ac.jp

BERT to improve the performance [14], [15]. Thus, we propose our model, StyleCapBERT, based on BERT. StyleCapBERT is capable of controlling the style and the length of the generated captions at the same time. Experimental results show that our model is capable of generating captions of different styles with only one model, and our model outperforms the state-of-the-art models in terms of standard image captioning evaluation metrics such as CIDEr [16] or ROUGE [17].

In this paper, we make the following contributions:

- We construct a new dataset, UTStyleCap4K. Our dataset is larger than any existing stylized image captioning datasets, correct in meanings and styles, diverse in the ways of describing the styles.
- We introduce a BERT based model that is capable of controlling the length level and the style of the generated caption on demand, only one model needs to be trained for multiple styles.
- Experimental results show that our dataset is capable of training various kinds of image captioning models, and our model reaches the state-of-the-art results on our dataset on different styles.

2. Related Works

2.1 Stylized Image Captioning

Since the two stylized image captioning datasets SentiCap [5] and FlickrStyle10K [6] are proposed, various models have been introduced in stylized image captioning based on them. Chen et al. [7] introduce SF-LSTM, using gated attention in Long Short Term Memory (LSTM) to help the model pay attention to different styles. Guo et al. [8] introduce MSCap, a generative adversarial network based model, whose generator tries to generate a stylized caption, discriminator tries to find whether the caption is fake and classify its style when it is real, enhanced by a back-translation module to get original caption given the stylized caption. Nezami et al. [9] use the generative adversarial network to solve this problem, proposing ATTEND-GAN, which uses reinforcement learning strategies to improve the model besides supervised training and adversarial training. Zhao et al. [18] introduce scene graphs as structural information from the images and sentences, and add a style memory module to encode the style-related information during training, which can be used to generate captions of different styles at the inference time. Tan et al. [10] use two ways of Transformer decoder to represent the factual and stylized caption decoding, and by multi-task learning the model can learn to generate stylized captions. Wu et al. [19] extract prior knowledge from sentimental corpus to obtain sentimental textual information and design a multimodal Transformer for sentimental visual captioning. Achlioptas et al. [20] collect a large dataset, Affection, which contains 526,749 emotional responses for 85,007 images. These responses contain the explanations for various emotional feelings towards the images, which is a bit different from the settings of stylized image captioning, but requires further study in the future.

2.2 Controllable Image Captioning

As the expression power of image captioning models increases, another criterion, controllability of the generated captions, is taken into concern [4]. While humans can tell different stories based on the same image, previous models lack the ability to control the captions they generate, relying solely on probabilities calculated for each word during inference.

Some image captioning models try to use scene graphs [21], which capture the structural information of objects in both images and captions, to control the generation of captions. Li et al. [22] extract visual features along with semantic features from scene graphs, and introduce a hierarchical-attentionbased module to learn discriminative features for word generation at each time step. Zhong et al. [23] detect the full scene graph of the image, then extracts different subgraphs of the full graph to generate different captions. Zhao et al. [24] construct a multi-modal knowledge graph to associate the visual objects with named entities in entity-aware image captioning.

There are also models directly modifying the captions. Sammani et al. [25] propose a Copy-LSTM with a Selective Copy Memory Attention mechanism (SCMA) to select the words in original captions generated by traditional image captioning models to copy, and then use an LSTM-based denoising auto-encoder to do minor fixes in the new sentence. Deng et al. [15] propose LaBERT, introducing length information into the captioning embeddings during training, so that the Transformer encoder model can identify captions of multiple lengths during the training time, thus being capable of controlling the lengths of captions during inference time.

Recently, Large Language Models (LLM) [26], [27] have achieved great success in natural language processing, and Large Vision Language Models (VLM) [28]–[30] have also become state-of-the-art models in various vision-language tasks. By using instruction tuning during training, these VLMs can effectively control the captions generated for images and prompts. Still, existing VLMs are very large in size, which often limits their practical applications. Recently, Zhang et al. [31] train a rather small model SEVLM of GPT-2 [32] size, and achieves competitive results compared with LLaVA-7B after fine-tuning and GPT-4 [33]. This demonstrates that while VLMs perform well with extensive data and model sizes, small models can still be competitive on specific tasks.

Some metrics are also proposed to encourage the diversity of generated captions. Want et al. [4] propose the CIDErBtw criterion to evaluate the distinctiveness of a caption with respect to those of similar images. Shi et al. [34] introduce the max-CIDEr criterion to serve as the reward for promoting diversity during reinforcement learning. Padmakumar et al. [35] proposes the homogenization score to evaluate the similarity of a group of captions. Meister et al. [36] calculates the corpus-level n-gram diversity based

Table 1: Comparison between our UTStyleCap4K and three existing dataset, MSCOCO, FlickrStyle10K, and SentiCap.

Datasets	Styles	Number of images			Number of captions		
		Training	Validation	Test	Training	Validation	Test
MSCOCO [12]	Factual	82,783	40,504	40,775	413,915	202,520	203,875
FlickrStyle10K [6]	Humorous	7,000	2,000	1,000	7,000	2,000[a]	5,000[a]
	Romantic	7,000	2,000	1,000	7,000	2,000[a]	5,000[a]
SentiCap [5]	Positive	824	174	673	2,464	409	2,019
	Negative	823	174	503	2,039	429	1,509
UTStyleCap4K	Positive	3,644	500	500	10,932	1,500	1,500
	Negative	3,644	500	500	10,932	1,500	1,500

[a] Not open to public.

on unique n-grams. Generally speaking, higher diversity allows the models to select a caption among a wider range of good captions, improving the controllability of these models.

3. Dataset

We summarize the existing datasets used in stylized image captioning in Table 1. The MSCOCO dataset [12] doesn't contain sentimental captions, and can be viewed as of "Factual" style. The FlickrStyle10K dataset [6] only releases its train split. The SentiCap dataset [5] is too small and contains some incorrect annotations. Therefore, we construct a new dataset, UTStyleCap4K. Since humorous and romantic captions in FlickrStyle10K involve much imagination of the image, which is a bit beyond image captioning, we use the settings of SentiCap, collecting captions of positive and negative styles.

3.1 Dataset Construction

We start with the MSCOCO dataset by using images from it. As there are 80 object classes annotated in the MSCOCO Captioning 2014 dataset, we randomly select 80 images for all of the 80 classes, and then remove the duplicate images to get 4,644 images in total. Using Amazon Mechanical Turk (AMT), we first ask workers to generate 10 positive and 10 negative captions for every image, giving one caption from the MSCOCO dataset as the sample caption. Then we ask 15 workers to select the best caption for each style for every image. Based on the choices, we automatically select the three positive and three negative captions with the largest number of votes. When different candidate captions share the same frequency, we choose the longest captions. We randomly distribute the images into training, validation, and test split.

In all, we collect 4,644 images with three positive and three negative captions for every image. There are 3,644 images with 10,932 positive and 10,932 negative captions in the train split (500 images with 1,500 positive and negative captions are in the validation split), and 500 images with 1,500 positive and negative captions in the text split. As shown in Table 1, our dataset contains much more captions than existing stylized image captioning datasets.

Table 2: Results of human evaluation on the comparison of quality between MSCOCO [12] and the two sections of our dataset UTStyleCap4K. Here we randomly select 500 images and choose 1,500 captions from each section. The DESC means descriptiveness score rated as 1, 2, 3, 4 and averaged across three AMT workers for each caption, higher is better. The "SENTI" column records the number of captions receiving 3, 2, 1 and 0 votes for having the correct style, voted by AMT workers.

	#caps	DESC	SENTI #votes			
			3	2	1	0
MSCOCO [12]	1500	2.90±1.05	1,288	203	9	0
Ours positive	1500	2.92±0.94	1,432	68	0	0
Ours negative	1500	2.90±0.95	1,358	138	4	0

3.2 Quality Control and Validation

As we are asking workers to generate captions for the image, one problem is that workers often write meaningless sentences, along with sentences of low quality or containing wrong sentiments. To solve this problem, Gan et al. [6] give some factual captions as the sample with corresponding stylistic modifications, Mathews et al. [5] also give factual captions as samples with candidate adjective-noun pairs (ANP). In this way, we can say that the workers are working on sentence editing, which can improve the quality of the captions, but also tends to collect captions similar to the sample sentence. During our experiments, we find many workers just add only some words to the original sentence, making sentences very similar to the sample sentence, which is often found in the SentiCap dataset as well. Therefore, we just give one factual caption as the sample and ask the workers with at least 90% HIT accuracy to create a positive or negative caption on their own. We reject all the sentences less than two words or just copying the sample sentence.

To validate the quality of captions in UTStyleCap4K, we conduct a crowdsourcing task to evaluate the descriptiveness and the correctness of emotions of the captions, given the 500 images randomly selected from UTStyleCap4K, along with their corresponding 1,500 captions of three styles from the MSCOCO dataset, the positive section and the negative section of UTStyleCap4K. Table 2 shows the result. In terms of descriptiveness, both sections of UTStyleCap4K are no worse than that of the MSCOCO dataset, with a smaller vari-

Table 3: Cosine similarities between the BERT embedding vectors of captions in SentiCap or UTStyleCap4K with the embedding vectors of captions in MSCOCO belonging to the same image. The smaller, the more different from MSCOCO.

	Positive (↓)	Negative (↓)
SentiCap [5]	0.800	0.749
UTStyleCap4K	**0.761**	**0.656**

Table 4: ANP dependability between captions in SentiCap and UTStyleCap4K. Here, the fraction number x/y represents x out of y captions contain an ANP. The smaller, the better.

	Positive(↓)	Negative(↓)
SentiCap [5]	4,823/4,892	3,917/3,977
UTStyleCap4K	1,715/13,932	682/13,932

Table 5: The diversity score of SentiCap and UTStyleCap4K.

	homogenization (↓)	n-gram diversity (↑)
SentiCap [5], Positive	0.257	3.264
UTStyleCap4K, Positive	**0.163**	**3.692**
SentiCap [5], Negative	0.239	3.316
UTStyleCap4K, Negative	**0.139**	**3.749**

ance. In terms of sentiment correctness, most workers vote correctly, with both sections of UTStyleCap4K have higher rate of correct votes. The results show that UTStyleCap4K includes sentiments correctly, with only four captions with one or less vote, and 93% of the captions receive all three votes.

Apart from subjective evaluation, we also compare our dataset with the SentiCap dataset in an objective way. We use an official pretrained BERT model [13] to evaluate the similarity of captions in UTStyleCap4K and SentiCap with captions in MSCOCO, since both our dataset and the SentiCap dataset use images from the MSCOCO dataset. For every image, we encode all the captions of it in UTStyleCap4K or SentiCap and MSCOCO, then we compute the cosine similarity between its embedding vector of captions in UTStyleCap4K or SentiCap and the embedding vectors of all five captions in MSCOCO dataset. We retain the max similarity scores of the five caption pairs, and average them over the whole UTStyleCap4K and SentiCap. Table 3 shows the result of the similarity scores, proving that our dataset is less similar to MSCOCO than SentiCap.

The SentiCap dataset depends too much on adjectives as they provide 10 adjective-noun pairs (ANPs) for every worker and ask them to include at least one ANP in the sentence, so we didn't limit the range of words that the workers use during data collection. Thus, we calculate the number of captions including an adjective-noun pair in UTStyleCap4K, given the ANP list provided by Mathews et al. [5]. Table 4 shows the result, proving that our dataset is less dependent on ANPs than SentiCap. Along with the results of sentimental correctness of our dataset shown in Table 2, it proves

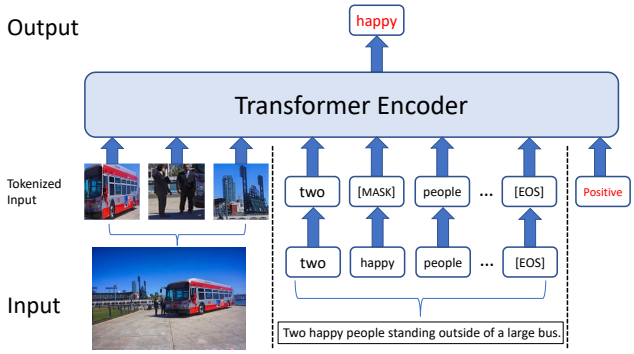

Fig. 2: **The pipeline of our model, StyleCapBERT**, for stylized image captioning. The bottom shows input as images and captions from the dataset, then we tokenize different object regions of images and every word of captions, with some words being masked, followed by a label embedding for the style. The Transformer encoder is trained to predict the masked words.

that sentiments don't depend only on adjectives, and our dataset provides captions with correct sentiments expressed in a wider range of words such as nouns and adverbs.

Finally, we calculate the diversity of UTStyleCap4K and SentiCap. We use the diversity[†] package of Python to calculate the homogenization score [35] and n-gram diversity [36] of each dataset. The results in Table 5 proves that UTStyleCap4K is more diverse than SentiCap under both evaluation metrics.

In all, our dataset reaches a high level of descriptiveness and correctness in sentiments, while being less similar to the original MSCOCO dataset and more diverse in describing sentiments, than the SentiCap dataset. Therefore, we believe that our dataset can perform better for training stylized image captioning models.

4. Proposed Method

4.1 Caption and Image Embeddings

The overall structure of StyleCapBERT is shown in Figure 2, given the input data as image-caption pairs, during the pre-training phase, we embed the images and captions into tokens, which are the inputs to the Transformer encoder. During this process we include the length and style information in these tokens.

Caption Embeddings For a caption $S = \{s_i\}_{i=1}^{L}$ of length L, based on the Length-aware BERT (LaBERT) [15] model, we encode four kinds of information for every word s_i, which are the token embeddings representing different words in a predefined dictionary, the position embeddings representing the position that every word is located at, and the length and style embeddings representing the length level and style of this sentence. The process of caption embeddings is shown in Figure 3.

[†]https://pypi.org/project/diversity/

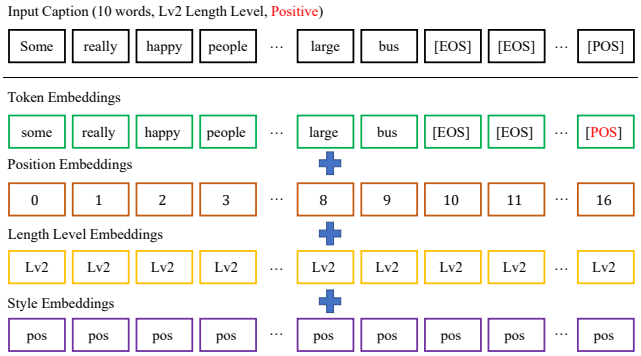

Fig. 3: **Our method of embedding caption words.** Given a caption, we will generate four kinds of embeddings for every word, which are the token embeddings, position embeddings, length level embeddings of Lv2, and style embeddings of positive. Then we add them together as the caption embedding. Notice that there is a [**POS**] token after the padded [**EOS**] tokens, representing the style of this caption.

To embed the length information, we assign different lengths into different length levels, where length levels are defined as regions in the number of words of a sentence. Suppose we are concerning l kinds of length levels, then we will use l-dimension one-hot vectors t_l to represent these length levels. For a sentence S of length L, suppose it falls into the length level of $[L_{\text{low}}, L_{\text{high}}]$, then we will use a length embedding matrix $W_l \in \mathbb{R}^{l\times d}$ to project its length level (i.e. the corresponding one-hot vector) into embedding space.

To embed the style information, for every word we will add a style embedding. Suppose we need to treat s kinds of styles, we use s-dimension one-hot vectors t_s to represent these styles, then we multiply t_s by a style embedding matrix $W_s \in \mathbb{R}^{s\times d}$, d being the embedding dimension.

In all, the tokenized input for a word s_i is represented as:

$$x_{s_i} = e_{w,s_i} + e_{p,i} + W_l^T t_l + W_{ws}^T t_s, \tag{1}$$

where $e_{w,s_i} \in \mathbb{R}^d$ represents the word embedding, $e_{p,i} \in \mathbb{R}^d$ represents the position embedding, the third term represents the length embedding, and the fourth term represents the style embedding, with $W_s \in \mathbb{R}^{s\times d}$ being the style embedding matrix. One example of our caption embedding method can be seen in Figure 3.

Image Embeddings Given an image **I** as input, we first use a pretrained object detection model to detect N region proposals in **I**, denoted as $\{r_i\}_{i=1}^N$. For every proposal, the detector will output the region visual features f_i^v, classification probabilities vector f_i^c and localization features f_i^l for every region r_i. Then the tokenized input for an image region r_i is represented as:

$$x_{r_i} = W_e^T f_i^e + W_o^T [\text{LN}(f_i^c), \text{LN}(f_i^l)] + e_{img}, \tag{2}$$

where the first term represents visual embeddings, the second term represents localization embeddings, the third term $e_{img} \in \mathbb{R}^d$ is a learnable embedding that distinguish the image region embeddings from text embeddings. Here $[\cdot,\cdot]$ represents concatenation, LN represents Layer Normalization [37], W_e and W_o represents the parameter matrix for visual and localization features respectively.

4.2 Training and Inference Procedure

Training During training, we have the image-caption pairs as input, then we mask some words of the caption and let the Transformer encoder to predict their real values. Given an image and its corresponding caption, we first encode the image regions $\{r_i\}_{i=1}^N$, then treat the caption denoted as S^*. We identify the length level $[L_{\text{low}}, L_{\text{high}}]$ and the style of the sentence, then we pad S^* with [**EOS**] token until the max length L_{high}, followed by a style token [**STYLE**], making the padded sequence reaching the length of $L_{\text{high}} + 1$, as can be seen in Figure 3. In this paper, we use three style tokens, [**FAC**], [**POS**] and [**NEG**], representing the factual, positive and negative styles respectively. Here the factual style represents captions from the MSCOCO dataset that don't contain sentiments and describe the image in a plain manner. Thus, we have the input tokens for the image and caption.

Next, we randomly mask m tokens in this padded sequence of caption tokens, replacing the true tokens s_i^* by the [**MASK**] token. Given the image region embeddings shown in Eqn. 2 and the masked caption word embeddings shown in Eqn. 1, StyleCapBERT is asked to predict the m words s_i at the masked position. We use the cross-entropy loss function to train StyleCapBERT:

$$\min \sum_{i=1}^{L_{\text{high}}+1} -\mathbb{1}(s_i) \log p(s_i = s_i^*), \tag{3}$$

where $\mathbb{1}(\cdot)$ is an indicator function that returns 1 if s_i = [**MASK**] and 0 otherwise. It can be seen that this loss function asks for StyleCapBERT's ability to recover the masked words based on the vision-language context.

Inference At inference time, the situation is different. Now we only have the images as input, and we ask our trained Transformer encoder to generate captions of all length levels and styles. Following LaBERT [15], we do the inference in an iterative fashion. At the first time step $t = 1$, we first embed the image regions using Eqn. 2. Then we initialize the caption as a sequence of L_{high} consecutive [**MASK**] tokens, followed by a [**STYLE**] token. This masked caption is then embedded using Eqn. 1. Thus, we feed the image and caption tokens into the Transformer encoder, and the Transformer encoder can predict a probability distribution over a predefined dictionary for every position in the caption. Then we would replace the [**MASK**] tokens by the words of maximum probability at all positions.

At each time step $t > 1$, given the sequence S' from the last time step $t - 1$, we replace the n words that has the lowest confidence scores by [**MASK**] tokens, and input them into the Transformer encoder to update these words with low

1. a is a in a a of in a a kitchen.
2. a great of nice a a and in kitchen kitchen kitchen.
3. a great of nice a a and a kitchen kitchen kitchen.
4. a woman of nice with a food nice of kitchen.
5. a wonderful woman running in a kitchen room of kitchen.
6. a pretty woman filled through a great room with kitchen.
7. a pretty woman kitchen through a kitchen room with kitchen.
8. a nice woman comes through a doorway of a kitchen.
9. a pretty woman comes through a doorway of a kitchen.
10. a pretty woman comes through a doorway of a kitchen.

Fig. 4: **One example of the generated caption of StyleCapBERT during inference.** For this image StyleCapBERT takes 10 steps to finally get the answer with the Lv2 length level and positive style. Here we use the green color to denote the words that will be changed in the next time step.

certainty. Then we update the confidence scores c_i of every words s_i by:

$$c_i \longleftarrow \begin{cases} \max\limits_{s} p_i(s_i = s), & i \text{ is a masked position,} \\ (c_i + \max\limits_{s} p_i(s_i = s))/2, & \text{otherwise,} \end{cases} \tag{4}$$

and we iterate this process until the results converge.

Figure 4 shows one example of the inference procedure in our experiments. For this image it takes 10 steps for StyleCapBERT to get the result on this length level [10,14] with positive style. We can see that in the beginning steps, the syntax of the caption is quickly fixed, and it takes several more steps for the model to decide the important adjectives and nouns. It also starts to generate positive words from the second iteration. Through this example, we believe that the inference procedure is correct, and our model is capable of controlling the length and the style of the generated captions.

5. Experiments

5.1 Experimental Settings

Baseline We train or use the following models on UTStyleCap4K to set a baseline on it:

- **Neural Image Caption (NIC)** [1] The classical encoder-decoder model, here we use a standard LSTM as the decoder.
- **att2in2** [38] Based on the Show, Attend and Tel model [42], but the image attention features are only input to the cell node of the LSTM, and we are using adaptive attention mechanism [43].
- **Transformer** [39] We follow the standard structure and treat the image features as query and value, caption features as key.
- **AoANet** [2] One important baseline image captioning model, which uses two layers of attention mechanism in the decoder.
- **LaBERT** [15] The base model of our model StyleCapBERT, uses a BERT-based model to control the length of the generated captions.
- **VisualGPT** [40] A GPT-2 [32] based model that can quickly adapt the pre-trained language model with a small amount of in-domain image-text data, which fits our task very well.
- **DIFNet** [11] A Transformer based model enhanced by a segmentation network generating segmentation features to improve the contribution of visual information for prediction. One baseline in traditional image captioning.
- **InstructBLIP** A instruction-tuning based VLM, which achieves state-of-the-art zero-shot generalization performance on a wide range of vision-language tasks.
- **LLaVA** A GPT-4 [33] based model and one of the most important VLMs in vision-language area.

Realizing the fact that some models need to be pre-trained on the MSCOCO dataset, to ensure a fair comparison, we set two tracks of tasks. One is training using data only from our UTStyleCap4K dataset, the other is using data from our dataset and the MSCOCO dataset. For traditional image captioning models, we train two models for the two styles. For our two-style and three-style models, we train one model for both styles (positive and negative) or three styles (positive, negative, factual).

Evaluation Metrics We evaluate all the models under standard image captioning metrics, BLEU [44], METEOR [45], ROUGE [17], CIDEr [16], and SPICE [46]. These metrics have different focuses on the sentences, some on correctness of grammar and syntax, some on the use of words and n-grams in the sentence, some on the correctness in meaning of the captions.

For a subjective evaluation, we also ask AMT workers to vote the description level and correctness of sentiments of each caption, in the same manner as in Table 2.

Implementation Details We initialize StyleCapBERT using the official pre-trained BERT model [13], which uses 12 layers of Transformer with 12 attention heads, the hidden size being 768. For every image, we detect 100 object

Table 6: Results of different models on UTStyleCap4K, after training using data only from UTStyleCap4K.

		BLEU-1	BLEU-4	ROUGE	METEOR	CIDEr	SPICE	DESC	SENTI
POS	fc [1]	48.2	8.6	33.7	14.4	33.4	10.3	3.08±1.00	98.2%
	att2in2 [38]	48.1	8.7	35.2	15.0	39.3	12.5	2.93±0.92	98.6%
	Transformer [39]	48.9	10.3	35.9	15.8	47.1	15.3	2.93±0.91	98.6%
	AoANet [2]	**53.4**	**11.0**	37.7	16.1	50.2	**16.1**	2.97±0.91	98.4%
	LaBERT [15]	51.7	9.1	**42.1**	21.1	64.4	15.6	2.73±1.12	87.1%
	Ours, 2 styles	48.8	8.6	**42.1**	**21.2**	**65.3**	15.3	2.96±0.90	98.2%
NEG	fc [1]	45.1	7.8	31.6	12.2	29.2	9.9	3.02±1.04	94.6%
	att2in2 [38]	48.4	9.3	34.3	13.1	39.9	12.7	2.91±0.91	94.6%
	Transformer [39]	49.0	10.1	34.2	14.9	43.3	14.8	2.95±0.90	94.8%
	AoANet [2]	**51.0**	**11.8**	35.7	15.2	50.4	15.8	2.94±0.95	93.2%
	LaBERT [15]	47.1	8.5	39.4	18.6	57.0	14.8	2.77±1.08	93.1%
	Ours, 2 styles	47.0	9.3	**42.5**	**20.5**	**68.5**	**16.5**	3.01±0.92	94.4%

Table 7: Results of different models on UTStyleCap4K, after training using data from both the MSCOCO and UTStyleCap4K dataset. InstructBLIP is directly used without finetuning, and for LLaVA we include the results with and without finetuning using UTStyleCap4K dataset. Here 'ft' means finetuning.

		BLEU-1	BLEU-4	ROUGE	METEOR	CIDEr	SPICE	DESC	SENTI
POS	NIC [1]	46.9	8.9	33.6	13.4	33.2	10.1	2.64±1.19	88.6%
	att2in2 [38]	49.5	9.5	35.2	15.2	39.5	13.0	2.62±1.08	87.8%
	Transformer [39]	50.9	10.1	35.8	15.6	43.8	14.8	2.66±1.12	86.5%
	AoANet [2]	53.2	10.1	37.5	15.7	47.4	15.9	2.64±1.10	84.5%
	VisualGPT [40]	56.9	11.9	36.7	15.3	57.3	-	2.82±1.12	84.0%
	DIFNet [11]	54.5	11.0	37.2	16.1	52.9	-	2.88±1.10	83.4%
	InstructBLIP [41]	40.5	8.9	38.6	20.6	61.1	**22.4**	**3.43**±0.91	95.9%
	LLaVA [29] w/o ft	50.0	11.5	38.9	19.9	59.2	20.3	3.28±0.90	95.7%
	LLaVA [29] w/ ft	58.0	**13.1**	40.4	20.0	69.2	20.0	3.31±0.93	95.8%
	Ours, 3 styles	**61.2**	10.4	**49.5**	**25.3**	**93.6**	21.6	3.08±0.85	**98.4%**
NEG	NIC [1]	45.3	8.3	32.0	12.2	31.7	10.3	2.70±1.16	92.2%
	att2in2 [38]	46.3	8.1	33.5	12.6	35.4	12.1	2.69±1.07	92.7%
	Transformer [39]	47.5	9.1	33.6	14.4	42.2	14.3	2.63±1.11	92.7%
	AoANet [2]	51.1	10.9	36.1	15.2	47.6	15.9	2.74±1.09	93.2%
	VisualGPT [40]	56.0	11.3	36.9	15.3	56.5	-	2.83±1.03	92.8%
	DIFNet [11]	52.6	12.3	37.3	15.7	57.6	-	2.90±1.06	91.6%
	InstructBLIP [41]	53.0	**13.0**	39.5	20.0	67.8	**21.4**	**3.45**±0.93	87.6%
	LLaVA [29] w/o ft	44.5	9.3	35.8	18.5	46.0	18.7	3.27±0.93	87.7%
	LLaVA [29] w/ ft	51.3	11.5	37.8	18.2	60.0	19.4	3.18±1.03	88.5%
	Ours, 3 styles	**56.9**	11.1	**45.6**	**23.4**	**79.5**	19.9	3.09±0.87	**93.4%**

regions using a Faster R-CNN [47] model pretrained on the Visual Genome dataset [48], and we use the feature maps of its fc6 layer as the region visual features, the classification probability of 1,600 classes predicted by Faster R-CNN as the region classification features, and the top-left and bottom-right coordinates along with the relative area of the object region as the region localization features. We train the two-style StyleCapBERT for 100,000 iterations with a batch size of 256. We use the AdamW optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and a weight decay of 10^{-2}. We use the tokenized embedding vector of the official pre-trainied BERT model given the word "positive", "negative" and "factual" as the [**POS**], [**NEG**] and [**FAC**] token. We assign four length levels, Lv1 as 1 to 9 words, Lv2 as 10 to 14 words, Lv3 as 15 to 19 words and Lv4 as 20 to 25 words. We ignore captions longer than 25 words.

For the three-style StyleCapBERT we add all the 23,220 MSCOCO captions for all the 4,644 UTStyleCap4K images into the training dataset, which will make five factual three positive and three negative captions for every image. Then we also train for 100,000 iterations with a batch size of 256, and other settings are the same as two-style StyleCapBERT.

For the NIC, att2in2, and AoANet models, we set the batch size as 64. When using only our data, we train for 60 epochs. When training using data from the MSCOCO and UTStyleCap4K dataset, we first train for 30 epochs using MSCOCO and then 30 epochs using UTStyleCap4K. For the Transformer and MFAN model, we set six layers of at-

Table 8: The performance of StyleCapBERT on the SentiCap [5] dataset, compared with state-of-the-art stylized image captioning models.

(a) On the positive section of SentiCap.

	B-4	R	M	C	S
ATTEND-GAN [9]	12.5	44.3	18.8	61.6	15.9
MFAN [10]	**18.0**	44.1	20.2	88.3	**21.0**
CNM [50]	-	-	16.3	55.0	-
Ours, 3-style	13.2	**51.3**	**26.3**	**109.9**	20.8

(b) On the negative section of SentiCap.

	B-4	R	M	C	S
ATTEND-GAN [9]	13.6	44.6	17.9	64.1	16.2
MFAN [10]	**18.0**	44.1	20.2	88.3	21.0
CNM [50]	-	-	17.0	54.8	-
Ours, 3-style	14.1	**51.0**	**26.0**	**112.4**	**21.6**

tention blocks with 8 attention heads, the input and output embeddings size as 512, and the inner-layer dimensionality as 2,048. We still set the batch size as 64 and train for the same number of epochs as NIC. For these models we use codes from Luo et al. [49] for experiment. For LaBERT, we use the official codes and train for 30 epochs using UTStyleCap4K. For VisualGPT and DIFNet, we use their official Github repository and follow their standard settings, training 30 epochs using UTStyleCap4K based on the pretrained models they provide.

For InstructBLIP, the finetuning script is missing, so we use the original version based on Vicuna-7b-v1.1 without finetuning on our dataset UTStyleCap4K, and we use the prompts as "Please describe this image in less than 25 words which expresses positive/negative feelings." For LLaVA, we use the v1.6-Vicuna-7b version, and finetune it using UTStyleCap4K for 20 epochs with a batch size of 16. We use the prompts as "Please write a concise sentence describing this image using positive/negative sentiments." for LLaVA with/without finetuning.

5.2 Experimental Results on UTStyleCap4K

Table 6 shows the results of six baseline models trained only using UTStyleCap4K, which are NIC, att2in2, Transformer, AoANet, LaBERT with our two-style StyleCapBERT (positive and negative), on the six objective evaluation metrics and two human evaluation metrics. For our two-style model and LaBERT, we show the 4-ensemble results on ROUGE, METEOR, and CIDEr, and we use the Lv2 results on BLEU-1, BLEU-4, and SPICE because they cannot be calculated in an ensemble way. In terms of CIDEr, ROGUE, and METEOR, our two-style model performs the best in both sections. On BLEU-n, AoANet is better than our model, but in the human evaluation results, our model still rates higher than AoANet. The comparison between our two-style model with LaBERT also shows the importance of including style information into embeddings, since on most objective evaluation metrics our two-style model is better than LaBERT, and on human evaluation our model is much better than LaBERT. It should be noticed that we train only one model for both styles, while we train two models for two styles for other kinds of models.

Table 9: The performance of three-style StyleCapBERT trained using SentiCap or UTStyleCap4K data, on the MSCOCO Karpathy's test split, which only contains factual style. Results show that our model StyleCapBERT is also capable of generating factual captions of high quality.

	B-4	R	M	C	S
VLP [14]	36.5	-	28.4	116.9	21.2
AoANet [2]	**37.2**	57.5	28.4	119.8	21.3
LaBERT [15], Lv2	35.3	57.4	28.4	118.2	**21.8**
Ours using SentiCap [5]	32.2	59.6	29.9	120.2	20.0
Ours using UTStyleCap4K	33.6	**63.0**	**34.1**	**123.8**	20.4

Table 7 shows the results of 10 baseline models capable of generating captions of three styles. NIC, att2in2, Transformer, AoANet, VisualGPT, and DIFNet are pretrained on the MSCOCO dataset and then trained on UTStyleCap4K, while our model, three-style StyleCapBERT (positive, negative, and factual) is trained only using on MSCOCO and UTStyleCap4K at the same time. InstructBLIP is directly used without finetuning, and LLaVA is used with or without finetuning. We still show the 4-ensemble results on ROUGE, METEOR, and CIDEr, and we use the Lv2 results for BLEU-1, BLEU-4, and SPICE. Although a lot of models nowadays would pretrain on the unstylized MSCOCO dataset and then train on stylized image captioning dataset, here from the results of the NIC, att2in2, Transformer and AoANet we can see that pretraining doesn't help these models very much, and even lowering the performance of these models. Although our three-style model performs much better than the two-style model, it should be noticed that we only include a small amount of MSCOCO data in such a change. While this shows the data efficiency of our three-style StyleCapBERT, it may also suggest that during pretraining one shouldn't include too much data from the MSCOCO dataset since they are way larger than and dissimilar to our dataset UTStyleCap4K.

It can also be seen that our model performs much better than state-of-the-art image captioning models VisualGPT and DIFNet, and is even better than large VLMs, InstructBLIP and LLaVA, on various evaluation metrics. Although our model is much simpler, by including the styles, our model learns how to really add styles onto the original factual descriptions, not just fine-tune on two sections of sentiments, or do instruction tuning based on prompts. Since our method of introducing styles into caption embeddings is universal, it can be transferred to other BERT-based models as well.

In Figure 5, we show one sample of the generated captions of different baseline models trained on our dataset. It can be seen that most models learn to tell the story with a sentiment, with our model's results being more descriptive than all other models. Still we have to say that ground truth captions collected by crowdsourcing tasks remain the best in fluency and quality, thus we think there is still a long way for

Positive Section
GT: the cat and dog were content being comfortable on the bed
FC: a cute cat is sitting on the bed
Att2in2: a cat is sleeping on a bed
Transformer: a cute cat sleeping on the bed
AoANet: a cute cat sleeping on the bed
LaBERT: a cute color dog is sleeping peacefully on the bed
VisualGPT: a cute dog sleeping on the bed
DIFNet: a cute cat sleeping on the bed
InstructBLIP: a pug and a black cat are laying on a bed together, with the pug holding a stuffed teddy bear
LLaVA: A stuffed teddy bear is laying on a bed next to a black cat.
LLaVA (with finetune): A cute puppy and a black cat are lying on the bed with a teddy bear.
Ours: a very cute dog is sleeping on the bed and a cute black cat looks on the bed

Negative Section
GT: a cat plans to attack a dog from behind who is ruining a stuffed animal
FC: a cat is sitting on the bed
Att2in2: a cat is sitting on a dirty bed
Transformer: a cat is laying on a bed
AoANet: a cat is laying on the bed
LaBERT: a cat is anxiously sleeping on the messy bed
VisualGPT: a cat is very sad
DIFNet: a cat is sleeping on the bed
InstructBLIP: a dog and a cat laying on a bed with a teddy bear
LLaVA: A stuffed teddy bear is laying on a bed next to a black cat.
LLaVA (with finetune): A black cat and a dog are laying on a messy bed.
Ours: an ugly and dirty dog laying on a bed next to a black cat

Fig. 5: One sample showing the generated captions of different baseline models on the test split of our dataset. The LaBERT is only trained on UTStyleCap4K and other models are trained on both UTStyleCap4K and MSCOCO. Here we use the Lv3 result from our three-style StyleCapBERT and the Lv2 result from LaBERT.

stylized image captioning models to really talk like human.

Our Results
Lv1: a man riding a bike past a train.
Lv2: a man is riding a bike past a passenger train.
Lv3: a man is riding a bike down a street in front of a passenger train.
Lv4: a man in a black jacket is riding a bike down a street in front of a red train.

MSCOCO Ground Truth
1. a person is riding a bicycle but there is a train in the background.
2. a red and white train and a man riding a bicycle
3. a man riding a bike past a train traveling along tracks.
4. a guy that is riding his bike next to a train
5. a man on a bicycle riding next to a train

Fig. 6: One sample of inference results of captions in factual style generated by StyleCapBERT. The image on the left is an input image, and the result consisted of four captions divided by four length levels in factual style and five captions from ground truth.

5.3 Extra Experimental Results for StyleCapBERT

To show that our model StyleCapBERT is capable of training on different dataset, we also train our three-style StyleCapBERT on the Senticap [5] dataset. As can be seen in Tables 8a and 8b, in terms of ROUGE, METEOR and CIDEr, our model is much better than state-of-the-art stylized image captioning model MFAN [10]. In terms of SPICE the performance is about the same, and in BLEU-4 the performance is worse. But we are using much less data than MFAN, and we achieve better results. Figure 6 shows this sample of generated captions of StyleCapBERT on the test split of the MSCOCO dataset. It can be seen that on all four levels our model generate quite correct and descriptive captions, and as the length level increases the model also tends to generate captions with more details.

Table 10: The performance of three-style StyleCapBERT on all four length levels, trained using MSCOCO and UTStyleCap4K data, on the test split of UTStyleCap4K.

		B-1	B-4	R	M	C	S
POS	NIC [1]	46.9	8.9	33.6	13.4	33.2	10.1
	DIFNet [11]	54.5	11.0	37.2	16.1	52.9	-
	Ours, Lv1	50.6	11.3	39.8	16.9	61.1	17.9
	Ours, Lv2	**61.2**	**13.1**	42.0	19.5	70.8	**21.6**
	Ours, Lv3	53.5	11.9	40.3	20.6	63.9	21.0
	Ours, Lv4	46.3	8.8	37.3	20.8	40.6	21.1
	Ours, 4-level	**61.2**	10.4	**49.5**	**25.3**	**93.6**	**21.6**
NEG	NIC [1]	45.3	8.3	32.0	12.2	31.7	10.3
	DIFNet [11]	52.6	**12.3**	37.3	15.7	57.6	-
	Ours, Lv1	44.9	8.0	34.6	14.7	49.4	16.5
	Ours, Lv2	**56.9**	11.1	38.8	18.1	62.3	**19.9**
	Ours, Lv3	46.7	8.4	36.1	19.0	48.9	19.8
	Ours, Lv4	40.4	6.6	33.0	18.5	28.8	19.2
	Ours, 4-level	**56.9**	11.1	**45.6**	**23.4**	**79.5**	**19.9**

Table 9 shows how stylized image captioning can help traditional image captioning, and our dataset is better than SentiCap in doing this, as our three-style StyleCapBERT trained on UTStyleCap4K performs better than its original model LaBERT, and better than the same StyleCapBERT

trained on SentiCap. This may prove that sentimental data also helps the model understand the factual style of describing images.

Since we also train StyleCapBERT to generate captions on four length levels, 1-9, 10-14, 15-19, and 20-25, in Table 10 we also show the results of different length levels compared with two baseline models and the ensemble model. We can see that generally speaking Lv2 of 10-14 words perform the best and even better than DIFNet on most metrics. Lv3 results are a bit better than Lv1 results, and Lv4 results are the worst in general. From these results we can see that by combining different length levels into an ensemble, we get a model better than all four length levels.

6. Conclusion

In this paper, we propose a stylized image captioning dataset, UTStyleCap4K, together with a stylized image captioning model, StyleCapBERT. Our dataset is larger in data size, less similar to base dataset MSCOCO, and less dependant on adjective-noun pairs. Our model takes styles into caption embeddings and reaches high scores in multiple experiments. Experimental results show the high correctness and descriptiveness of our dataset, and our mode reaches the best performance on our datasets.

References

[1] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[2] L. Huang, W. Wang, J. Chen, and X.Y. Wei, "Attention on attention for image captioning," Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019.

[3] X. Xiao, L. Wang, K. Ding, S. Xiang, and C. Pan, "Deep hierarchical encoder–decoder network for image captioning," IEEE Transactions on Multimedia, vol.21, no.11, pp.2942–2956, 2019.

[4] J. Wang, W. Xu, Q. Wang, and A.B. Chan, "Compare and reweight: Distinctive image captioning using similar images sets," Proceedings of the European Conference on Computer Vision (ECCV), 2020.

[5] A.P. Mathews, L. Xie, and X. He, "Senticap: Generating image descriptions with sentiments," Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), 2016.

[6] C. Gan, Z. Gan, X. He, J. Gao, and L. Deng, "Stylenet: Generating attractive visual captions with styles," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[7] T. Chen, Z. Zhang, Q. You, C. Fang, Z. Wang, H. Jin, and J. Luo, ""factual" or "emotional": Stylized image captioning with adaptive learning and attention," Proceedings of the European Conference on Computer Vision (ECCV), 2018.

[8] L. Guo, J. Liu, P. Yao, J. Li, and H. Lu, "Mscap: Multi-style image captioning with unpaired stylized text," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

[9] O.M. Nezami, M. Dras, S. Wan, C. Paris, and L. Hamey, "Towards generating stylized image captions via adversarial training," Pacific Rim International Conference on Artificial Intelligence (PRICAI), 2019.

[10] Y. Tan, Z. Lin, H. Liu, and F. Zuo, "Improving stylized image captioning with better use of transformer," International Conference on Artificial Neural Networks, pp.347–358, 2022.

[11] M. Wu, X. Zhang, X. Sun, Y. Zhou, C. Chen, J. Gu, X. Sun, and R. Ji, "Difnet: Boosting visual information flow for image captioning," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.18020–18029, 2022.

[12] T.Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C.L. Zitnick, "Microsoft coco: Common objects in context," Proceedings of the European Conference on Computer Vision (ECCV), 2014.

[13] J. Devlin, M.W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2019.

[14] L. Zhou, H. Palangi, L. Zhang, H. Hu, J. Corso, and J. Gao, "Unified vision-language pre-training for image captioning and vqa," Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), 2020.

[15] C. Deng, N. Ding, M. Tan, and Q. Wu, "Length-controllable image captioning," Proceedings of the European Conference on Computer Vision (ECCV), 2020.

[16] R. Vedantam, C. Lawrence Zitnick, and D. Parikh, "Cider: Consensus-based image description evaluation," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[17] C.Y. Lin, "Rouge: A package for automatic evaluation of summaries," Proceedings of the Workshop on Text Summarization Branches Out, 2004.

[18] W. Zhao, X. Wu, and X. Zhang, "Memcap: Memorizing style knowledge for image captioning," Proceedings of the AAAI Conference on Artificial Intelligence, pp.12984–12992, 2020.

[19] X. Wu and T. Li, "Sentimental visual captioning using multimodal transformer," International Journal of Computer Vision, vol.131, no.4, pp.1073–1090, 2023.

[20] P. Achlioptas, M. Ovsjanikov, L. Guibas, and S. Tulyakov, "Affection: Learning affective explanations for real-world visual data," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.6641–6651, 2023.

[21] J. Gu, S. Joty, J. Cai, H. Zhao, X. Yang, and G. Wang, "Unpaired image captioning via scene graph alignments," Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019.

[22] X. Li and S. Jiang, "Know more say less: Image captioning based on scene graphs," IEEE Transactions on Multimedia, vol.21, no.8, pp.2117–2130, 2019.

[23] Y. Zhong, L. Wang, J. Chen, D. Yu, and Y. Li, "Comprehensive image captioning via scene graph decomposition," Proceedings of the European Conference on Computer Vision (ECCV), 2020.

[24] W. Zhao and X. Wu, "Boosting entity-aware image captioning with multi-modal knowledge graph," IEEE Transactions on Multimedia, pp.1–12, 2023.

[25] F. Sammani and L. Melas-Kyriazi, "Show, edit and tell: A framework for editing image captions," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[26] T. Brown, B. Mann, N. Ryder, M. Subbiah, J.D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, *et al.*, "Language models are few-shot learners," Advances in neural information processing systems, vol.33, pp.1877–1901, 2020.

[27] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, *et al.*, "Llama 2: Open foundation and fine-tuned chat models," arXiv preprint arXiv:2307.09288, 2023.

[28] W.L. Chiang, Z. Li, Z. Lin, Y. Sheng, Z. Wu, H. Zhang, L. Zheng, S. Zhuang, Y. Zhuang, J.E. Gonzalez, I. Stoica, and E.P. Xing, "Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality," March 2023.

[29] H. Liu, C. Li, Q. Wu, and Y.J. Lee, "Visual instruction tuning," Advances in neural information processing systems, vol.36, 2024.

[30] N. Rotstein, D. Bensaid, S. Brody, R. Ganz, and R. Kimmel, "Fusecap: Leveraging large language models for enriched fused image captions," 2023.
[31] J. Zhang, L. Zheng, D. Guo, and M. Wang, "Training a small emotional vision language model for visual art comprehension," arXiv preprint arXiv:2403.11150, 2024.
[32] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, *et al.*, "Language models are unsupervised multitask learners," OpenAI blog, vol.1, no.8, p.9, 2019.
[33] OpenAI, "Gpt-4 technical report," 2024.
[34] J. Shi, Y. Li, and S. Wang, "Partial off-policy learning: Balance accuracy and diversity for human-oriented image captioning," Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp.2187–2196, October 2021.
[35] V. Padmakumar and H. He, "Does writing with language models reduce content diversity?," The Twelfth International Conference on Learning Representations, 2024.
[36] C. Meister, T. Pimentel, G. Wiher, and R. Cotterell, "Locally typical sampling," Transactions of the Association for Computational Linguistics, vol.11, pp.102–121, 2023.
[37] J.L. Ba, J.R. Kiros, and G.E. Hinton, "Layer normalization," arXiv preprint arXiv:1607.06450, 2016.
[38] S.J. Rennie, E. Marcheret, Y. Mroueh, J. Ross, and V. Goel, "Self-critical sequence training for image captioning," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.
[39] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L.u. Kaiser, and I. Polosukhin, "Attention is all you need," Advances in Neural Information Processing Systems (NeurIPS), 2017.
[40] J. Chen, H. Guo, K. Yi, B. Li, and M. Elhoseiny, "Visualgpt: Data-efficient adaptation of pretrained language models for image captioning," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.18030–18040, June 2022.
[41] W. Dai, J. Li, D. LI, A. Tiong, J. Zhao, W. Wang, B. Li, P.N. Fung, and S. Hoi, "Instructblip: Towards general-purpose vision-language models with instruction tuning," Advances in Neural Information Processing Systems, ed. A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, pp.49250–49267, Curran Associates, Inc., 2023.
[42] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," Proceedings of the 32nd International Conference on Machine Learning (ICML), 2015.
[43] J. Lu, C. Xiong, D. Parikh, and R. Socher, "Knowing when to look: Adaptive attention via a visual sentinel for image captioning," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.
[44] K. Papineni, S. Roukos, T. Ward, and W.J. Zhu, "Bleu: a method for automatic evaluation of machine translation," Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL), 2002.
[45] S. Banerjee and A. Lavie, "Meteor: An automatic metric for mt evaluation with improved correlation with human judgments," Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, pp.65–72, 2005.
[46] Z. Wang, B. Feng, K. Narasimhan, and O. Russakovsky, "Towards unique and informative captioning of images," Proceedings of the European Conference on Computer Vision (ECCV), ed. A. Vedaldi, H. Bischof, T. Brox, and J.M. Frahm, 2020.
[47] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in Advances in Neural Information Processing Systems 28, ed. C. Cortes, N.D. Lawrence, D.D. Lee, M. Sugiyama, and R. Garnett, pp.91–99, 2015.
[48] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.J. Li, D.A. Shamma, *et al.*, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," International Journal of Computer Vision, vol.123, no.1, pp.32–73, 2017.
[49] R. Luo, B. Price, S. Cohen, and G. Shakhnarovich, "Discriminability objective for training descriptive captions," arXiv preprint arXiv:1803.04376, 2018.
[50] X. Wu, W. Zhao, and J. Luo, "Learning cooperative neural modules for stylized image captioning," International Journal of Computer Vision, vol.130, no.9, pp.2305–2320, 2022.

Chi Zhang received the B.S. degree in physics from Peking University, and the M.S. degree in information and communication engineering from The University of Tokyo. He is currently a doctor student at the Department of Information and Communication Engineering, Graduate School of Information Science and Technology, The University of Tokyo.

Tao Li received the M.S. degree in computer applied technology from Peking University, and the Ph.D. degree in information and communication enginneering from The University of Tokyo. He is currently working at NVIDIA. His research interests include video understanding, self-supervised learning.

Toshihiko Yamasaki received the Ph.D. degree from The University of Tokyo. He is currently a Professor at Department of Information and Communication Engineering, Graduate School of Information Science and Technology, The University of Tokyo. He was a JSPS Fellow for Research Abroad and a visiting scientist at Cornell University from Feb. 2011 to Feb. 2013. His current research interests include attractiveness computing based on multimedia data, pattern recognition, machine learning, and so on.