Image Caption Generation with Visual Attention

Project of AI3611 Intelligent Perception and Cognition Practice, 2024 Spring, SJTU

Xiangyuan Xue (521030910387)

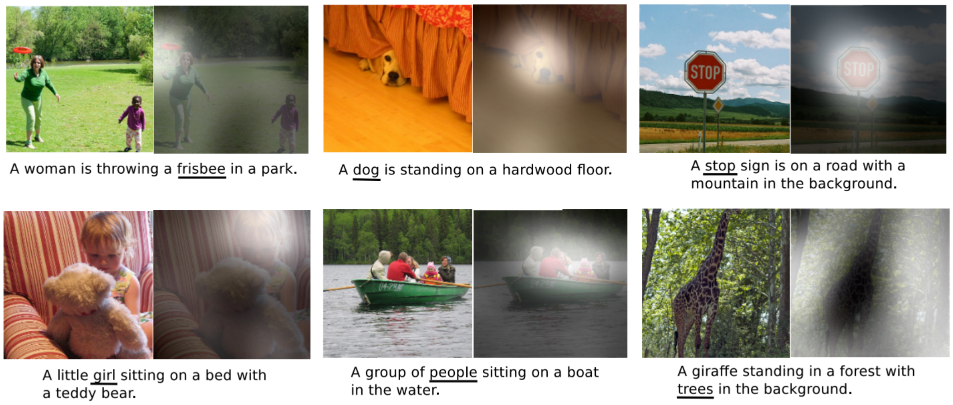
School of Electronic Information and Electrical Engineering

Abstract

Image caption generation is a cross-modal task that requires the model to understand both the visual and textual information and establish the relationship between them. In this project, we explore solving the image caption generation task by using the encoder-decoder framework with visual attention. We introduce the image captioner model, the scheduled sampling and beam search methods, and the various evaluation metrics. We conduct experiments to investigate how the hyperparameters affect the performance of the model. We also conduct ablation studies to verify the effectiveness of the scheduled sampling and beam search methods.

1 Introduction

In recent years, computer vision has witnessed the development from simple tasks like image classification to complex tasks like image generation and comprehension. At the same time, the significant progress made in natural language processing (NLP) makes it possible to combine vision and language to solve challenging cross-modal tasks [8]. Image caption generation is a representative task in this field, which aims to generate a natural language description of the given image [20]. It not only requires the model to understand both the visual and textual information but also to establish the relationship between them [14]. Image caption generation to some extent bridges the gap between computer vision and human perception and can be applied in various real-world applications, such as image classification, image retrieval and content analysis.



A woman is throwing a frisbee in a park.

A dog is standing on a hardwood floor.

A stop sign is on a road with a mountain in the background.

A little girl sitting on a bed with a teddy bear.

A group of people sitting on a boat in the water.

A giraffe standing in a forest with trees in the background.

Figure 1: An overview of the image caption generation task. The model takes an image as input and generates a natural language description of the given image as output.

Before the rise of deep learning, traditional methods for image caption generation mainly rely on hand-crafted features and statistical models [1]. However, these methods cannot achieve satisfactory performance due to the lack of representation power and generalization ability. With the development of deep learning, a method based on the visual detector and language model is proposed to generate image captions [16], where the visual detector extracts words from the image and the language model stitches them together to form a sentence. However, this method fails to model the interaction between different objects in the image. This problem remains unsolved until the introduction of the encoder-decoder framework [13], which uses a convolutional neural network (CNN) to encode the image and a recurrent neural network (RNN) to decode the feature into a sentence. Later, the attention mechanism is introduced to the encoder-decoder framework [21], which significantly enhances the model performance by allowing the model to focus on different parts of the image when generating each word. The encoder-decoder framework with attention mechanism is still one of the state-of-the-art methods for image caption generation nowadays.

In this project, we will explore solving the image caption generation task by using the encoder-decoder framework with visual attention [21]. Such a framework follows an end-to-end learning paradigm and promises a satisfactory performance, which can help us better understand the essence of the cross-modal task. In Section 2, we will introduce the pipeline of the image captioner model. In Section 3, we will present the experimental results and analyze the model performance. Finally, we will summarize our findings and provide some insights in Section 4.

2 Method

In this section, we will describe our method in four main parts: image captioner, scheduled sampling, beam search, and evaluation metrics. The overall pipeline of the encoder-decoder framework with visual attention is illustrated in Figure 2, where a convolutional neural network (CNN) [15] is used as the encoder to extract the feature and a recurrent neural network (RNN) [11] with attention mechanism is used as the decoder to generate the caption [21]. The scheduled sampling and beam search techniques are introduced to improve the model performance. Multiple metrics are introduced in our experiments to evaluate the model performance comprehensively.

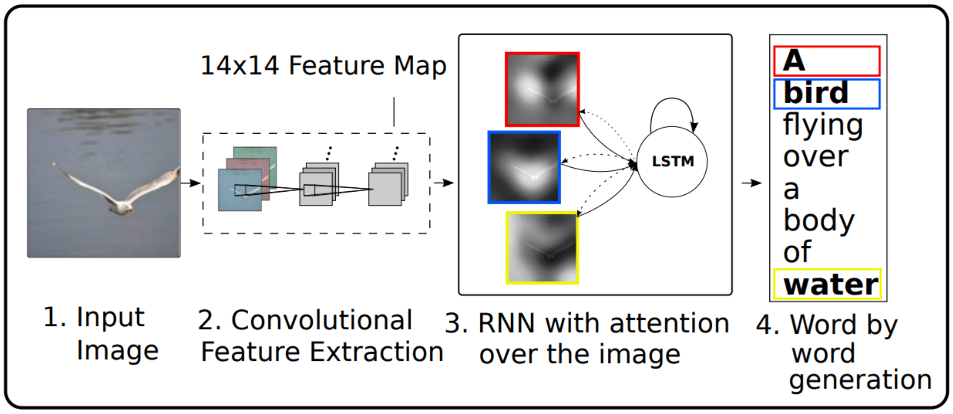

14x14 Feature Map
LSTM
A
bird
flying
over
a
body
of
water
1. Input Image
2. Convolutional Feature Extraction
3. RNN with attention over the image
4. Word by word generation

Figure 2: The overall pipeline of our framework. A convolutional neural network (CNN) is used as the encoder to extract the feature of the image, and a recurrent neural network (RNN) with attention mechanism is used as the decoder to generate the caption.

2.1 Image Captioner

Image encoder. The image encoder is responsible for extracting the feature of the given image, which contains the visual information needed for generating the caption. We adopt a ResNet-101 model [10] pretrained on the ImageNet dataset [6] as the image encoder to ensure adequate representation power, encoding the input image into a 14×14 feature map $\boldsymbol{g}$ with 2048 channels.

Visual attention. The visual attention mechanism is introduced to enhance the model performance by allowing the model to focus on different parts of the image when generating each word. Given the image feature $\boldsymbol{g}$ and the hidden state of $\boldsymbol{h}_{t-1}$, the visual attention mechanism computes the weight

$\alpha_{t,i}$ for each time step t and each spatial location i, which is formulated as

$$\alpha_{t,i} = \frac{\exp\left[\phi(\boldsymbol{g}_i, \boldsymbol{h}_{t-1})\right]}{\sum_{k=1}^{L} \exp\left[\phi(\boldsymbol{g}_k, \boldsymbol{h}_{t-1})\right]} \tag{1}$$

where L is the location number and ϕ is an attention model, which can be implemented as a multi-layer perceptron (MLP). Then the weighted feature map $\boldsymbol{z}_t$ is computed as

$$\boldsymbol{z}_t = \sum_{i=1}^{L} \alpha_{t,i} \cdot \boldsymbol{g}_i \tag{2}$$

Note that the weighted feature map $\boldsymbol{z}_t$ will be concatenated with the predicted word embedding $\boldsymbol{y}_{t-1}$ after gating and fed into the decoder to generate the next word.

Text decoder. The text decoder is responsible for generating the caption based on the feature extracted by the image encoder. We adopt a long short-term memory (LSTM) network [11] which is fused with the visual attention mechanism as the text decoder. Besides the visual information, the prediction should also consider the textual information, namely the previous generated words in the sentence. Given the last hidden state $\boldsymbol{h}_{t-1}$, the weight β_t is computed as

$$\beta_t = \frac{\exp\left[\psi(\boldsymbol{h}_{t-1})\right]}{1 + \exp\left[\psi(\boldsymbol{h}_{t-1})\right]} \tag{3}$$

g where ψ can be a fully connected layer. Then the context vector $\boldsymbol{c}_t$ is computed as

$$\boldsymbol{c}_t = \beta_t \cdot \boldsymbol{z}_t \tag{4}$$

Finally, the context vector $\boldsymbol{c}_t$ will be concatenated with the predicted word embedding $\boldsymbol{y}_{t-1}$ and fed into the LSTM cell as is shown in Figure 3 to generate the next hidden state $\boldsymbol{h}_t$, which is further fed into a prediction head to generate the probability distribution of the next word.

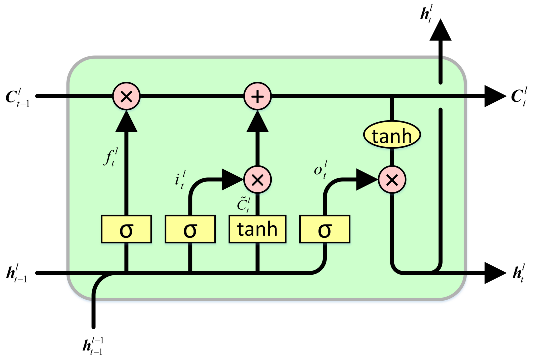


Figure 3: An overview of the LSTM cell. The input is constructed by concatenating the context vector $\boldsymbol{c}_t$ and the predicted word embedding $\boldsymbol{y}_{t-1}$, where the context vector is computed by the attention model based on both the visual and textual information.

2.2 Scheduled Sampling

The scheduled sampling technique [4] is introduced to alleviate the exposure bias problem, which is caused by the discrepancy between the training and inference process. In the traditional setting, the model is trained to maximize the likelihood of each token given the ground truth at the previous time step. However, during the inference process, the ground truth at the previous time step is replaced by the last predicted token, which may lead to the model suffering from compounding errors. To address this issue, the scheduled sampling technique modifies the training strategy by randomly choosing whether to use the ground truth or the predicted token at each time step. At the i-th epoch, we use the ground truth with probability ϵ_i and the predicted token with probability $1-\epsilon_i$. It is clear that if $\epsilon_i = 1$, the model is trained with the traditional strategy. If $\epsilon_i = 0$, the model is trained the same as inference. Therefore, we prefer to dynamically decay the sampling probability ϵ_i during the training process, so that the model can converge faster during the training process and generalize

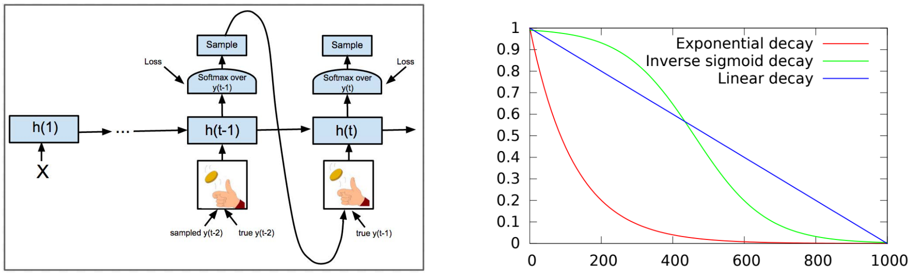


Figure 4: An illustration of the scheduled sampling. The left figure shows the training process with the scheduled sampling. The right figure shows the decay schedule of the sampling probability.

better during the inference process. There are multiple decay schedules. As is shown in Figure 4, we mainly introduce linear decay, exponential decay, and inverse sigmoid decay.

Linear decay. The linear decay schedule can be formulated as

$$\epsilon_i = \max\left(\epsilon, k - ci\right) \tag{5}$$

where $0 \leq \epsilon < 1$ determines the lower bound of the sampling probability, k and c provide the offset and slope of the decay respectively, which depend on the expected speed of convergence.

Exponential decay. The exponential decay schedule can be formulated as

$$\epsilon_i = k^i \tag{6}$$

where $k < 1$ determines the decay rate of the sampling probability.

Inverse sigmoid decay. The inverse sigmoid decay schedule can be formulated as

$$\epsilon_i = \frac{k}{k + \exp\left(\frac{i}{k}\right)} \tag{7}$$

where $k \geq 1$ determines the decay rate of the sampling probability.

2.3 Beam Search

The beam search technique [9] is a heuristic search algorithm that is widely used in sequence generation tasks [7]. When decoding the caption, we need to search all the possible word sequences to find the global optimal solution. However, the search space grows exponentially with the sequence length, which makes exhaustive search computationally infeasible. In contrast, greedy search always selects the word with the highest probability at each time step, which tends to generate suboptimal results. The beam search technique is introduced to balance the trade-off between the computational complexity and the quality of the generated caption. At each time step, the beam search maintains a set of k partial hypotheses, which are extended by appending each possible word from the vocabulary. Then the hypotheses are ranked based on the accumulated log-likelihood, and only the top k hypotheses are retained for the next time step, which is illustrated in Figure 5.

2.4 Evaluation Metrics

To evaluate the performance of the model comprehensively, we introduce multiple metrics including Bleu [17], Rouge [5], METEOR [3], CIDEr [19], SPICE [2], and SPIDEr. Different metrics focus on different aspects when evaluating the quality of the generated caption.

Bleu. Bilingual evaluation understudy (Bleu) is a commonly used metric for machine translation tasks, which measures the n-gram overlap between the prediction and the reference. The core idea is to measure the proximity of the prediciton to the reference. Bleu considers an n-gram rather than a word, but all the n-grams are weighted equally, which may bring about some bias.

Rouge. Recall-oriented understudy for gisting evaluation (Rouge) is a set of metrics for text summarization algorithms. We mainly use the Rouge-L score as a metric in this project, which measures the length of the longest common subsequences between the prediction and the reference. Rouge is similar to Bleu, but it focuses on recall rather than precision.

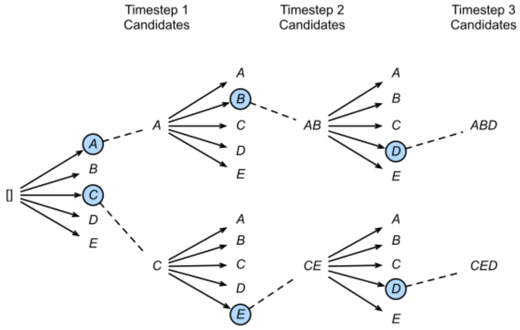


Figure 5: An illustration of the beam search. The beam search maintains a set of k partial hypotheses. Only the hypotheses with the highest accumulated log-likelihood are retained for the next time step.

METEOR. Metric for evaluation of translation with explicit ordering (METEOR) is also a metric for machine translation tasks, which aligns the prediction and the reference and matches the accuracy, recall, and F-score of various cases. Note that METEOR will penalize the prediction if the word order is different from the reference, so it has a high correlation with human judgment.

CIDEr. Consensus-based image description evaluation (CIDEr) is a metric speicially designed for image annotation tasks, which performs a term frequency-inverse document frequency (TF-IDF) analysis to represent the sentence into a vector. Then CIDEr is computed as the cosine similarity between the prediction and the reference, which compensates for the limitations of Bleu.

SPICE. Semantic propositional image caption evaluation (SPICE) is a metric speicially designed for image captioning tasks, which parses the sentence into a scene graph and computes the precision, recall, and F-score of the extracted tuple sets. Compared to CIDEr, SPICE focuses more on the semantic meaning of the sentence, which is more consistent with human judgment.

SPIDEr. SPIDEr takes the average of SPICE and CIDEr, which provides a comprehensive evaluation balancing the term frequency and the semantic meaning of the generated caption.

3 Experiments

In this section, we will train the models on the given dataset and compare the performance of different settings. We will research how the hyperparameters affect the performance and try to optimize the generation results. We also conduct ablation studies to explore the effectiveness of the scheduled sampling and beam search methods. In the following parts, we will present the results in detail.

3.1 Experiment Setup

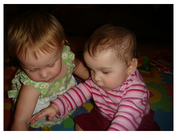
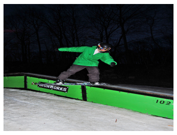
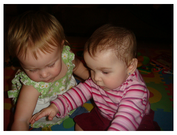
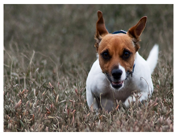
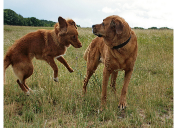
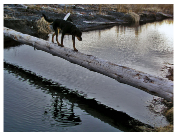
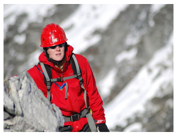
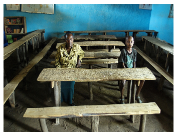
We employ the Flickr8k dataset [12] to train and evaluate the models. The dataset contains $8,000$ images from Yahoo's photo album site Flickr, among which $6,000$ images are used for training, $1,000$ images are used for validation, and $1,000$ images are used for testing. Most images depict humans performing various activities. Each image is associated with 5 sentences as the annotations.

In the default setting, we do not employ the scheduled sampling and beam search methods. We use the image captioner model with 384 embedding size and 256 hidden size. We train the model for 30 epochs with a batch size of 128 using RMSprop [18] optimizer with a learning rate of 10^{-3}. The model with the best validation metrics will be selected for evaluation.

3.2 Model Performance

We first train the model with the default setting and evaluate the performance on the test set. We select 4 representative metrics to present the image captioning results with the best and worst scores, including Bleu-4, METEOR, CIDEr and SPICE. This can help us intuitively analyze the performance of the model and better understand the focus of different metrics.

Table 1: Sample images and captions with the best and worst scores in Bleu-4, METEOR, CIDEr and SPICE metrics. The blue texts are the reference. The green text presents the prediction with the best score, and the red text presents the prediction with the worst score.

Metric	Image	Caption
Bleu-4		A toddler girl plays with another younger toddler girl. Two babies sitting on a playmat reaching for something. Two babies are sitting close together while reaching for something. Two children sitting on the floor. Two young children sitting together playing. Baby is playing in a backyard. (1.000)
		A helmeted boy flies through the air on a snowboard. A snowboarder balancing on a wall. A snowboarder in green grinds along the edge of a rail at night. A snowboarder wearing a green jacket jumping a green railing. A snowboarder wearing a green jacket jumps above a low gate. Man in a red jacket is skateboarding on a ramp. (0.000)
METEOR		A toddler girl plays with another younger toddler girl. Two babies sitting on a playmat reaching for something. Two babies are sitting close together while reaching for something. Two children sitting on the floor. Two young children sitting together playing. Baby is playing in a backyard. (0.574)
		A little white and brown dog runs through brown grass. A small dog runs through a field. Brown and white dog runs through brush. Small dog running through a field. The dog is running full speed through the meadow. Small dog is running through the grass. (0.021)
CIDEr		A dog hops in a field while another dog stands next to it. One dog is jumping up at another dog in a grassy field. The two tan colored dogs are in a field, and one is jumping in the air. Two brown dogs in a field. Two brown dogs play with one another in the field. Brown dog is standing on a grassy field. (3.892)
		A black dog is slowly crossing a fallen log that is outstretched over a stream of water. A dog crossing a river on a bridge made of a fallen tree. A dog walks on a log across a small river. The black dog is walking along a tree trunk bridge over water. The dog walks across the stream on a fallen log. Black dog is jumping over a stream. (0.000)
SPICE		A hiker in a red on a snowy peak. A person in a red jacket and a red hardhat stands near a mountain. A person wearing a red jacket and helmet walks up near the large rock. A woman wearing a red hat and red jacket is hiking in the snow topped mountains. The person wearing the red hardhat is on the mountain. Man in a red jacket and a red jacket is standing in the snow. (0.632)
		An African man and a boy are standing behind a coarse wooden table in a blue room. A young boy and an older woman sitting at a brown table in a blue room. The person in the yellow shirt is sitting with a child at a wooden picnic table. Two people sit in front of tables in a room with blue walls. Two people sitting in a blue room with ratty wooden seats and tables. Woman is sitting on a bench next to a picnic table. (0.000)

As is shown in Table 1, the model can sometimes generate reasonable captions that are similar to the reference. However, it may also generate wrong information such as mistaking the red color for the green one or recognizing a man as a woman. Similarly, the metrics do not always faithfully reflect the quality of the generated captions. For example, the caption with the worst METEOR score is actually accurate and informative. This indicates the significance of combining multiple metrics to evaluate the performance of the model. In addition, we can conclude that Bleu-4 and CIDEr are more sensitive to the key words and phrases, while METEOR and SPICE are more sensitive to the sentence structure and grammar.

To compare the performance of the models with different architectures, we implement the captioner model with different embedding sizes and hidden sizes. All the hyperparameters follow the default setting except for the embedding size and hidden size. Multiple metrics of the models with different architectures are reported in Table 2.

Table 2: The performance of the models with different architectures. The numbers in the model name represent the embedding size and hidden size (e.g., Captioner-128-128 denotes a captioner model with 128 embedding size and 128 hidden size). The metrics including Bleu-4, Rouge-L, METEOR, CIDEr, SPICE and SPIDEr are reported.

Model	Bleu-4	Rouge-L	METEOR	CIDEr	SPICE	SPIDEr
Captioner-128-128	0.171	0.410	0.194	0.497	0.141	0.319
Captioner-128-256	0.186	0.416	0.196	0.513	0.139	0.326
Captioner-256-128	0.171	0.407	0.194	0.496	0.142	0.319
Captioner-256-256	**0.188**	**0.417**	0.196	**0.518**	**0.144**	**0.331**
Captioner-384-128	0.169	0.412	0.196	0.492	0.141	0.317
Captioner-384-256	0.174	0.413	**0.198**	0.503	0.143	0.323

From the table we can see that the model with 256 embedding size and 256 hidden size achieves the best performance in terms of Bleu-4, Rouge-L, CIDEr, SPICE and SPIDEr. The model with 384 embedding size and 256 hidden size achieves the best METEOR score, which is slightly better than the model with 256 embedding size and 256 hidden size. This result indicates that increasing the hidden size can significantly enhance the representation ability of the model. However, increasing the embedding size may not always improve the performance and a 256 embedding size is sufficient.

3.3 Ablaion Study

To explore the effectiveness of the scheduled sampling and beam search methods, we conduct ablation studies based on the default setting. We compare the performance of the models with different decay schedules and beam sizes. The results are reported in Table 3 and Table 4.

Table 3: The performance of the models with different decay schedules. The methods respectively represent the baseline without scheduled sampling, linear decay, exponential decay and inverse sigmoid decay. The metrics include Bleu-4, Rouge-L, METEOR, CIDEr, SPICE and SPIDEr.

Method	Bleu-4	Rouge-L	METEOR	CIDEr	SPICE	SPIDEr
Baseline	0.174	0.413	**0.198**	**0.503**	**0.143**	**0.323**
Linear	0.165	0.413	0.186	0.477	0.138	0.307
Exponential	**0.180**	**0.416**	0.187	0.495	0.140	0.318
Sigmoid	0.172	0.411	0.180	0.457	0.127	0.292

Table 4: The performance of the models with different beam sizes. The methods respectively represent the baseline without beam search, beam size of 3, beam size of 5 and beam size of 7. The metrics include Bleu-4, Rouge-L, METEOR, CIDEr, SPICE and SPIDEr.

Method	Bleu-4	Rouge-L	METEOR	CIDEr	SPICE	SPIDEr
Baseline	**0.174**	**0.413**	**0.198**	**0.503**	**0.143**	**0.323**
Beam-3	0.170	0.398	0.186	0.480	0.139	0.309
Beam-5	0.170	0.397	0.182	0.482	0.135	0.309
Beam-7	0.168	0.394	0.182	0.467	0.133	0.300

From the table we can see that the scheduled sampling method can effectively improve the performance of the model, especially in terms of Bleu-4 and Rouge-L. The exponential decay schedule shows the best effect among the three methods. This indicates that the scheduled sampling method can effectively alleviate the bias between training and inference and improve the generalization ability of the model. On the other hand, the beam search method brings no improvement to the performance of the model regardless of the beam size. We owe this result to the training strategy, where the model always maximizes the likelihood of the next word given the ground truth, so the greedy search tends to generate better sentences than the beam search. This hypothesis is supported by the fact that the best model combines the scheduled sampling and beam search methods.

4 Conclusion

In this project, we research on solving the image caption generation problem by using the encoder-decoder framework with visual attention. We first introduce the background of the image captioning task and the motivation of using encoder-decoder framework with visual attention. We then present the pipeline of the proposed framework, including the image captioner model, the scheduled sampling and beam search methods, and the various evaluation metrics. We implement the proposed framework and conduct experiments on the Flickr8k dataset. We compare the performance of the models with different architectures to investigate how the hyperparameters affect the results. We find that increasing the hidden size can significantly enhance the representation ability of the model. However, increasing the embedding size may not always improve the performance and a 256 embedding size is sufficient for the image captioning task.

Besides, we conduct ablation studies to investigate the effectiveness of the scheduled sampling and beam search methods. We find that the scheduled sampling method can effectively alleviate the bias between training and inference and improve the generalization ability of the model. In contrast, the beam search method brings no improvement to the performance of the model regardless of the beam size. We claim that the beam search method should be combined with the scheduled sampling method to achieve the best performance.

References

[1] Ahmet Aker and Robert Gaizauskas. Generating image descriptions using dependency relational patterns. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 1250–1258, 2010.

[2] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Spice: Semantic propositional image caption evaluation. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V 14*, pages 382–398. Springer, 2016.

[3] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005.

[4] Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. *Advances in neural information processing systems*, 28, 2015.

[5] Lin Chin-Yew. Rouge: A package for automatic evaluation of summaries. In *Proceedings of the Workshop on Text Summarization Branches Out, 2004*, 2004.

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[7] Markus Freitag and Yaser Al-Onaizan. Beam search strategies for neural machine translation. *arXiv preprint arXiv:1702.01806*, 2017.

[8] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[9] Alex Graves. Sequence transduction with recurrent neural networks. *arXiv preprint arXiv:1211.3711*, 2012.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

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

[12] Micah Hodosh, Peter Young, and Julia Hockenmaier. Framing image description as a ranking task: Data, models and evaluation metrics. *Journal of Artificial Intelligence Research*, 47:853–899, 2013.

[13] MD Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, and Hamid Laga. A comprehensive survey of deep learning for image captioning. *ACM Computing Surveys (CsUR)*, 51(6):1–36, 2019.

[14] Lun Huang, Wenmin Wang, Jie Chen, and Xiao-Yong Wei. Attention on attention for image captioning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4634–4643, 2019.

[15] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[16] Girish Kulkarni, Visruth Premraj, Vicente Ordonez, Sagnik Dhar, Siming Li, Yejin Choi, Alexander C Berg, and Tamara L Berg. Babytalk: Understanding and generating simple image descriptions. *IEEE transactions on pattern analysis and machine intelligence*, 35(12):2891–2903, 2013.

[17] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

[18] Tijmen Tieleman and G Hinton. Divide the gradient by a running average of its recent magnitude. coursera: Neural networks for machine learning. *Technical report*, 2017.

[19] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575, 2015.

[20] Haoran Wang, Yue Zhang, and Xiaosheng Yu. An overview of image caption generation methods. *Computational intelligence and neuroscience*, 2020, 2020.

[21] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057. PMLR, 2015.