CLIP-DIFFUSIONLM: APPLY DIFFUSION MODEL ON IMAGE CAPTIONING [*]

Shitong Xu
Imperial College London
shitong.xu19@imperial.ac.uk

ABSTRACT

In this work, we applied denoising diffusion probabilistic models to text generation in image captioning tasks. We show that our CLIP-diffusionLM *** On the flickr8k dataset, the model showed. By combining samples from flickr8k and flickr30k dataset, our model showed ... performance. In addition, the model achieved ... zero shot performance in COCO 2015 image caption task. Our code is available at ...

contribution: experiment on learning rate, optimizer, adding mechanism, cosine schedule, using classifier free or not. using model to predict image feature or not, scale up to larger dataset

Keywords Diffusion model · CLIP · Non auto-regressive generation

1 Introduction

Image captioning has being a focus of research over the recent years. Previous text encoder used could be split to 2 general classes, which are autoregressive and non-autoregressive. Most of the sota models falls in the autoregressive class[]. However, autoregressive generation suffer from 1) the slow generation speed due to the generation step is token by token; and 2) not capable of refining prefix of sentences based on the later generated tokens. Multiple attempts have experimented using a non-autoregressive model in the text generation steps[]. The closest to our work is 2019 Masked Non-Autoregressive Image Captioning[], which used a BERT model as generator and involves 2 steps to refine the generated sequence. However, these work still used a discrete generation process, which means masking out certain tokens and train model to refine words in these certain positions. To the best of our knowledge, there has not been other work on generating caption embedding based on continuous generations steps. Our work aim at employing a model to refine generated token continuously on their embedding. In particular, we used pretrained CLIP model for generating image and text features, and distilbert model based on diffusion-lm for text sequence generation. Our contribution could be summarized as follow: - apply diffusion model in image captioning tasks - experiments with multiple feature fusion methods, in particular the relative importance of restoring the token feature and certainty of generated sequence.

2 Related Work

2.1 Autoregressive image captioning

2015 deep caption with multimodal rnn[] proposed the mRNN model, which used CNN for image extraction and RNN for text generation. 2016 show attend tell[] employed the LSTM for text generation and exerimented on soft and hard attention for early fusion between image feature and text feature. Based on this early fusion method, 2016 knowing where to look[] experimented the late fusion of image and text features, allowing model to attend on either image or text modality during generation. 2017 cascade recurrent nn[] experimented on reversing the generated caption, allowing its backend model to refine the former tokens based on later caption tokens. 2018 gla[] used attention model to combine local and global feature from images, so that captions can more accurately identify occluded objects. Similarly, 2019 stack vs[] also used image features from both high and low generaliaty, and combined them using cross attention. Their

[*] *Citation*: **Authors. Title. Pages.... DOI:000000/11111.**

work also involves multi step refining of the generated text caption. Caption in each step the caption is autoregressively generated. 2019 unsupervised image caption[] trained image caption in a GAN style, with a LSTM discriminator reproducing the original image feature from generated text sequence. 2019 Variational Autoencoder-Based Multiple ImageCaptioning Using a Caption Attention Map[] used variational auto encoder for extracting image information, their model allows(?) changing the image feature by sampling from the learned distribution, thus produce various captions for different images. 2021 CLIP cap[] experimented on using pretrained CLIP image feature for sequence generation. The CLIP features are transformed to a sequence of token and used as prefix for a GPT-2 model in generation.

2.2 Non autoregressive image captioning

In contrast, non-autoregressive models benefits from the attention models' ability to pass textural information in both direction during generation. The text generated in former timesteps could adjust based on text in later timesteps, thus is expected to achieve better performance. 2019 Masked Non-Autoregressive Image Captioning[] used BERT[] as text decoder and employed a 2 step generation method. Based on this work, Partially Non-Autoregressive Image Captioning [] and semi Non-Autoregressive Image Captioning[] partitioned the generated text in subgroups, words in the same group are generated non-autoregressively and different groups are generated in autoregressive way. Our method falls in this catogory and most close to the 2019 Masked Non-Autoregressive Image Captioning[]. The difference is we chose to use diffusion model as the non-autoregressive generation model. 2022 GRIT[] experimented changing the cross attention part of transformer decoder to use both Reginal feature from Faster RCNN and Grid features from swin transformer.

2.3 Diffusion models

Diffusion models aims at training a model that denoise a feature from Gaussial noise to original features. [] proposed the DDPM model to simplified the loss function by only letting models to predict the noise in generation steps, and proposed alternative loss functions by removing the weight terms. Based on DDPM, improved ddpm [] proposed several improvments based on DDPM, including setting variance to be learnable parameters, apply cosine instead of linear noise schedule, and speed up forward process by reducing forward steps.

Diffusion model beat GAN[] is another work which used larger model, more attention heads, multiscaled attention and classifier guidance(?is the classifier guidance reduce generation step or higher score) for reducing generation steps. Classifuer guided diffusion is a technique of using a classifier to help generating features with higher . The classifier is pretrained on dataset with noise added. To guide the generation, the model's output is feed into the classifier, to get gradient on which direction to optimise parameter in order to generate image closer to target class.

To avoid training classifier for guiding model, classifier free guidance technique is proposed, the task is to provide generator both guided or unguided samples, and train an additional model to predict the relationship in 2 models' output. Example of applying classifier free guidance includes GLIDE[], (other class free guide work?)

Diffusion lm[] which is a recent work on applying continuous diffusion model on text generation, this paper provides various techniques to improve the performance of continuous diffusion model on text generation.

ddim[] reduced the variance in forward process. The result showed that by reducing variance to 0, the deterministic model achieved higher FID score in image generation on both CIFAR10 and CelebA.

By using diffusion model as text-to-image generation, DALL-E 2 and GLIDE model achieved significant image generation performance. Dall-e 2[] is a recent work on using CLIP and diffusion model for image generation task. The model used CLIP model for extracting feature from text, predict the corresponding image CLIP feature through prior network, then use predicted image CLIP feature for final image generation. The model achieved significant novelty in generated images. The innovativity of generated of image from DALL-E 2 also provided us the inspiration to train a image-to-text model with diffusion model in generation step.

3 Background

3.1 Diffusion models

The training of denoise diffusion probabilistic model involves generation of noised samples (forward process), and denoising based on model's output (backward process). Let x$_0 betheoriginalfeature, theforwardprocessincrementlyaddnoiseto$x$_0$ to generates a sequence of T noised features $[x_1, ...x_T]$. Each x$_t atsteptissampledfromprobabilitydistribution$q(x$_t$|$x_{t-1}$) $= N(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I)$.

From reparameterization trick, the x$_tatanysteptcouldbedirectlygeneratefromx_0$:x$_0$ = ... The backward process is using a trained model with parameter θ to denoise the samples generated in the forward process. The training objective is to minimize the negative log-likelihood of generating x$_0, thatistominimize$E(-$\log(p_\theta(x_0))) = E(-\log(\int p_\theta(x_0,...,x_T), d(x_1,...,x_T))) = E(-\log(\int p_\theta(x_0,...,x_T), d(x_1,...,x_T)))$. Where $p_\theta(x_{t-1}|x_t) = N(x_{t-1}, \mu_{theta}(x_t),...)$ and μ_θ is model's prediction on mean of x_{t-1} conditioned on x_t. By modeling the backward process as a Markov Process, $p_\theta(x_1,...,x_T)$ is simplified to $p_\theta(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}|x_t)$. From variational lower bound, $E(-\log(p_\theta(x_0))) \leq E_q[\log(\frac{p_\theta(x_0..x_T)}{q(x_1..x_T|x_0)})] = E_[\log(p(x_T)) + \sum_{t=1}^{T}\log(\frac{p_\theta(x_{t-1}|x_t)}{q(x_t|x_{t-1})})]$. From the work of [], expanding and reweighting each term of the negative log-likelihood gives a concise loss function $L = \sum_{t=1}^{T} E_{q(x_t|x_0)}||\mu_\theta(x_t,t) - \mu_{x_t,x_0}||^2$

Due to the large generation step number (T = 1000 as proposed in []), and the generation step being autoregressive on the denoised feature in the previous step, the reverse diffusion is significantly slower than the other generative models (... for gan and ... for diffusion). Multiple stetegies were proposed to accelerate the generation process. In Improved DDPM [] a subset of generation steps is selected. Model is trained to predict ... . In diffusion-lm the model is trained directly predict the x$_0insteadoftheintermediatestepscontainingnoise.Inourexperiments, the...showedbetterreproducequalitycomparedwith....I$ $decoderstructure., whichhasgenerationstepspropotionaltotheoutputsequenselength.autoregressivelyapplytheoutputofx_0tr$

Based on the propose from, we added an additional rounding term[] in our loss function, parameterized by $E_{p_\theta(\hat{x}|x_t)} - \log(p_\theta(w|\hat{x})) = E_{p_\theta(\hat{x}|x_t)} - \log(\prod_{i=1}^{L} p(w_i|\hat{x}_i))$. L represent the generated sequence length, w represent the gound truth sentence and $\hat{x}$ is the predicted sequence embedding from the input x_t. $p_\theta(w_i|\hat{x}_i)$ follows the softmax distribution. The training objection change to the following function:

$L = \sum_{t=1}^{T} E_{q(x_t|x_0)}||\mu_\theta(x_t,t) - \mu_{x_t,x_0}||^2 + -\log(p_\theta(w|\hat{x}))$

In our experiments, we found this term significantly influence the model performance.

4 CLIP-DiffusionLM

5 Experiments

Our model is based on Distilbert[] model, Distilbert is a model trained by distiling on BERT[] and has around 40

5.1 fusion: concatenation or elementwise adding

5.2 relative importance of confidency of prediction and restored feature

5.3 guidance free training

5.4 learning rate

5.5 x$_0predictionorx_{t-n}prediction$

5.6 number of x$_tpredictions$

5.7 Conclusion

We present the application of diffusion in image caption task, and proved its validity in limited dataset. Particularly, we identified the certainty term to be an important term in loss function to help model converge, and introduced the adaptive ratio adjustment to balance its importance with other terms. There are various improvements to the model and training process: - Experiment on output the attention graph of the model, to check the model did focus on the correct region of the image. - In various cases, the model failed to identify the correct object colour. For example, after correctly identify a girl wearing dress and a shirt, the model mistake the colour of shirt to the colour of the dress. - The output text sequence suffers from apparent grammar mistakes, for example, missing subject, repeated words. Additional supervision on the output text grammar might help model reduce such error. - We trained on raw image data of the dataset. However, image argumentation has proven to be a valid method to improve the performance[]. Performing data argumentation might improve the model's generalizability and help reduce the wrong colour alignment problem as discussed above. We believe analysing and improving based on the above observations, diffusion as text generation step could achieve comparable or better performance than auto-regressive models.

$$\xi_{ij}(t) = P(x_t = i, x_{t+1} = j | y, v, w; \theta) = \frac{\alpha_i(t) a^{w_t}_{ij} \beta_j(t+1) b^{v_{t+1}}_j(y_{t+1})}{\sum^N_{i=1} \sum^N_{j=1} \alpha_i(t) a^{w_t}_{ij} \beta_j(t+1) b^{v_{t+1}}_j(y_{t+1})} \tag{1}$$

Paragraph Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

6 Examples of citations, figures, tables, references

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetuer at, consectetuer sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui. [**?**, **?**] and see [**?**].

The documentation for `natbib` may be found at

`http://mirrors.ctan.org/macros/latex/contrib/natbib/natnotes.pdf`

Of note is the command `\citet`, which produces citations appropriate for use in inline text. For example,

```
   \citet{hasselmo} investigated\dots
```

produces

> Hasselmo, et al. (1995) investigated…

`https://www.ctan.org/pkg/booktabs`

6.1 Figures

Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetuer odio sem sed wisi. See Figure **??**. Here is how you add footnotes. [2] Sed feugiat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Ut pellentesque augue sed urna. Vestibulum diam eros, fringilla et, consectetuer eu, nonummy id, sapien. Nullam at lectus. In sagittis ultrices mauris. Curabitur malesuada erat sit amet massa. Fusce blandit. Aliquam erat volutpat. Aliquam euismod. Aenean vel lectus. Nunc imperdiet justo nec dolor.

6.2 Tables

Etiam euismod. Fusce facilisis lacinia dui. Suspendisse potenti. In mi erat, cursus id, nonummy sed, ullamcorper eget, sapien. Praesent pretium, magna in eleifend egestas, pede pede pretium lorem, quis consectetuer tortor sapien facilisis magna. Mauris quis magna varius nulla scelerisque imperdiet. Aliquam non quam. Aliquam porttitor quam a lacus. Praesent vel arcu ut tortor cursus volutpat. In vitae pede quis diam bibendum placerat. Fusce elementum convallis neque. Sed dolor orci, scelerisque ac, dapibus nec, ultricies ut, mi. Duis nec dui quis leo sagittis commodo. See awesome Table **??**.

6.3 Lists

- Lorem ipsum dolor sit amet

[2] Sample of the first footnote.

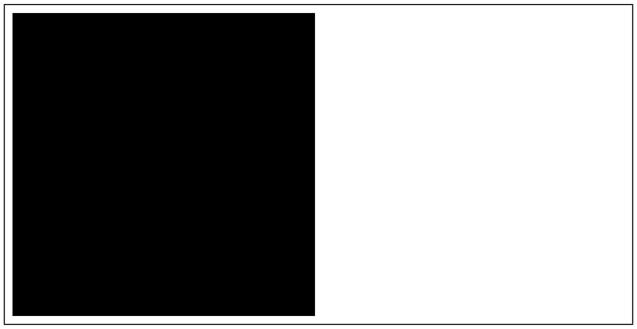

Figure 1: Sample figure caption.

Table 1: Sample table title

Part		
Name	Description	Size (μm)
Dendrite	Input terminal	$\sim$100
Axon	Output terminal	$\sim$10
Soma	Cell body	up to 10^6

- consectetur adipiscing elit.
- Aliquam dignissim blandit est, in dictum tortor gravida eget. In ac rutrum magna.

7 Conclusion

Your conclusion here

Acknowledgments

This was was supported in part by......