GRIT: Faster and Better Image captioning Transformer Using Dual Visual Features (ECCV 2023)

Rohan Prasad
rpp5524@psu.edu

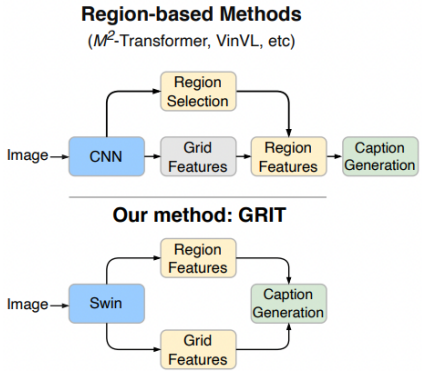

Figure 1. Image Captioning using GRIT with Region and Grid Features

1. Task

The task involves the generation of correct and semantic image captions using a machine learning model. Image captioning requires substantial understanding of scenes, objects, and their relationships for meaningful textual descriptions. The different challenges include correctly capturing contextual relationships between objects, dealing with noisy or missing data from detection processes, and ensuring computationally efficient captions without sacrificing quality. Existing methods often rely on two types of visual features: region-based and grid-based. Region-based features, derived from object detection models like Faster R-CNN, focus on capturing object-level information. These features often fail to capture spatial relationships between objects. They are also computationally expensive and prone to inaccuracies in object detection.

GRIT (Grid- and Region-based Image Captioning Transformer) (Figure 1) proposes a solution by combining these two types of features within a Transformer-only [8] architecture. By using a DETR-based [6] framework for region features and a Swin[5] Transformer for grid features, GRIT achieves a seamless integration of object-level details and contextual relationships. Its end-to-end design eliminates the need for complex preprocessing steps like non-maximum suppression (NMS), enabling efficient training and inference. This innovative approach improves captioning accuracy.

2. Related Work

The "Show and Tell"[1] model was one of the first deep learning approaches to picture captioning, using an encoder-decoder structure. The encoder, a Convolutional Neural Network (CNN), extracted global visual information from the input image, and the decoder, a Long Short-Term Memory (LSTM) network, created captions consecutively. The model trained both the encoder and the decoder from start to finish, and it performed admirably on benchmark data. However, this was based on a whole representation of an image, which often resulted in information loss or loss scene of spatial relations. This challenge prevented the generation of captions that represent fragile aspects, such as the objects interaction within the image. Furthermore, because LSTMs are sequential, they can have difficulty identifying long-term dependencies in data, especially for complex images involving multiple entities or activities.

Anderson et al. [2] introduced the Bottom-Up and Top-Down Attention model, a major advancement in image captioning that effectively combined attention mechanisms with region-based features. The model was able to extract region-specific features, allowing it to isolate and identify unique objects by using Faster R-CNN. The Top-Down mechanism included a layer of contextual understanding by stressing on contextually relevant regions while the Bottom-Up approach moved the model's focus toward important objects. These processes significantly improved caption accuracy and relevancy, especially for complex image compositions. This strategy did have some serious disadvantages, though. The significant processing lag brought on the Faster R-CNN's computing demands made real-time testing challenging. Issues in object detection like misclassifying items or missing important regions, typically made it way through into the captioning stage, which reduced the overall performance. It was challenging for the model to fully comprehend object interactions like actions or spatial dynamics due to its low contextual integration across regions.

The M2 Transformer [3] utilized a memory-augmented attention mechanism to enhance the representation of long-term dependencies and inter-object relationships in image captioning. It replaced the traditional encoder-decoder paradigm with a Transformer-based architecture [8], allowing the model to leverage self-attention for encoding object relationships within an image. The memory approach dynamically stores and gets image regions information, which lets the model generate contextually sensitive captions. This approach showed significant improvements in processing complicated scenarios when compared with earlier methods. However, the model was still limited by its reliance on region-based features extracted using Faster R-CNN, which made it vulnerable to errors from the object detection process. While the memory augmentation improved information retention, it often lacked the ability to seamlessly integrate object-specific and contextual features, which restricted its ability to generate highly descriptive captions in some scenarios.

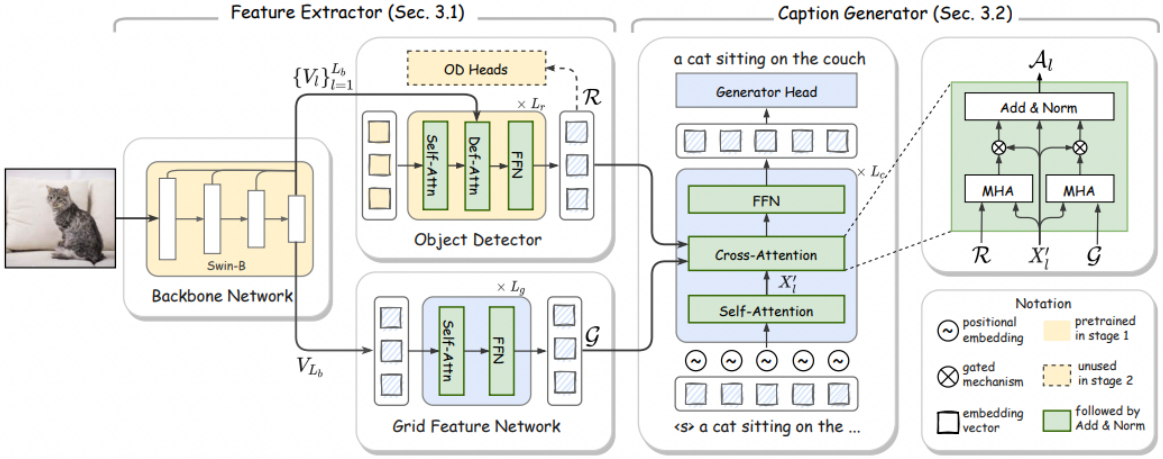

Figure 2. GRIT Architecture

SimVLMhuge [4] advanced the field of image captioning by employing a large-scale vision-language pretraining approach. It trained on 1.8 billion image-caption pairs, allowing the model to get a thorough understanding of multimodal semantics. The model's architecture made good use of Transformers to merge visual and textual representations, resulting in cutting-edge performance across a variety of vision-language tasks such as captioning and visual question answering. SimVLMhuge was supreme at producing highly accurate and contextually nuanced captions, even for delicate/unusual scenes, majorly due to its extensive pretraining data. However, the model depends on large-scale data which brought limitations. It requires significant compute which makes it difficult for researchers and practitioners to perform experiments, raising concerns about its feasibility and the scalability of reproducing results. Furthermore, its dependency on pretraining limited its applicability to domains with limited or specialized data, highlighting a gap in its generalization capabilities compared to resource-efficient models.

The Swin [5] Transformer, a cutting-edge vision transformer intended for great performance and computational efficiency in computer vision tasks, forms the foundation of the GRIT concept This is the **State of the Art Architecture** used in thus paper.. Unlike traditional Vision Transformers (ViT), which compute global attention over image patches, the Swin [5] Transformer incorporates patch reduction and shifted window mechanisms to enable local attention. Through the use of Deformable DETR [6] for object detection and Swin [5] Transformers for feature extraction, GRIT achieves end-to-end training efficiency, greatly lowering computational overhead while improving accuracy. One potential approach to image captioning is the combination of grid and area features, which helps to bridge the gap between contextual relationships and object-specific knowledge. The development of picture captioning toward more effective, precise, and scalable solutions is demonstrated by GRIT's capacity to learn from dual visual features.

3. Approach

The first stage of the GRIT (Figure 2) model involves feature extraction using the Swin Transformer and Deformable DETR. From these multi-scale outputs, the last feature map is processed to extract “grid features”. This is achieved by passing the feature map through a self-attention Transformer, which models spatial interactions to capture contextual information across the entire image.

$$\mathcal{L}_v(y,\hat{y}) = \sum_{i=1}^{N}[\underbrace{-\log\hat{p}_{\hat{\sigma}(i)}(c_i) + \mathbf{1}_{c_i\neq\varnothing}\mathcal{L}_{box}(b_i,\hat{b}_{\hat{\sigma}(i)})}_{\text{object detection}} \underbrace{-\log\hat{p}_{\hat{\sigma}(i)}(a_i)}_{\text{attribute prediction}}] \quad (1)$$

For “region-based features”, the model employs Deformable DETR [6], an improved version of the Detection Transformer (DETR). Deformable DETR extracts region features by processing the Swin Transformer's multi-scale outputs with a stack of deformable attention layers. This approach avoids computationally expensive operations like non-maximum suppression (NMS) used in traditional CNN-based detectors, enabling end-to-end training with loss given in equation (1). The resulting region features provide detailed object-specific information, which complements the broader contextual insights from the grid features.

The caption generation module in GRIT is a lightweight Transformer that combines the grid and region features through a “parallel cross-attention” mechanism as shown in Figure 3 (c). This mechanism allows the model to process both feature types simultaneously, preserving their distinct contributions while leveraging their complementary nature. For each word in the generated caption, the model attends to both grid and region features using separate attention heads, ensuring that contextual and object-specific information is integrated effectively.

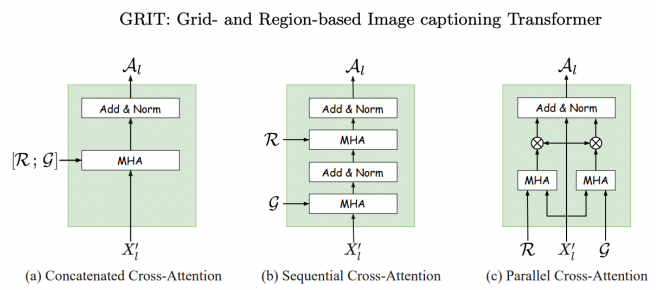

Figure 3. Three designs of cross-attention mechanism to use dual visual features

The caption generation process is autoregressive. This implies that each word is predicted in sequence, based on previously generated visual features and words. This ensures that the generated captions are coherent and grammatically correct. To enhance performance, the model incorporates a gating mechanism in the cross-attention layers, weighting the contributions of grid and region features dynamically based on the input image and current caption context.

The implementation of GRIT builds on existing libraries for Deformable DETR [6] and Swin Transformer. These pre-existing components are adapted to work seamlessly within the GRIT architecture, while custom modules are developed for feature fusion and caption generation. The integration of these components ensures computational efficiency and facilitates end-to-end training. The project setup includes the installation of necessary dependencies such as PyTorch, TensorBoard, SpaCy, and Timm. The training process involves downloading and preprocessing the COCO [7] dataset, which is organized into training, validation, and test splits. The region and grid features are extracted and processed during the training phase to optimize performance. The evaluation is performed on the Karpathy[9] test split using standard metrics like CIDEr, BLEU, and SPICE. Inference scripts are provided for generating captions on individual images, either through command-line scripts or interactive Jupyter notebooks.

In addition to utilizing existing libraries and implementations, custom code was written for preprocessing the dataset and building essential components of the GRIT pipeline. The dataset is preprocessed using SpaCy. To guarantee compliance and consistency with the model's input needs, it tokenizes and cleans the data. A setup script also that simplifies the process of compiling and deploying the MultiScaleDeformableAttention module is also used.

4. Dataset

The GRIT model leverages two primary datasets: COCO [7] (Common Objects in Context) and Visual Genome, each serving distinct purposes within the training pipeline. The COCO [7] dataset is widely regarded as the benchmark for image captioning tasks. It contains 123,287 images, with each image annotated with five diverse and descriptive captions. This richness in annotations makes COCO [7] particularly suitable for training and evaluating caption generation models. For offline evaluation, the widely adopted Karpathy[9] split, where 113,287, 5,000, and 5,000 images are used for training, validation, and testing respectively. The authors have taken a combination of 4 datasets. But due to compute and storage challenges, I have considered just the COCO14 dataset.

Data preprocessing plays a vital role in ensuring the consistency and quality of the input data. For both datasets, images are resized to standard dimensions to ensure uniformity during training. Captions are lowercased, and punctuation is removed to simplify the textual input.

Method	V.E Type	#VL Data	Performance Metrics					
			Bleu@1	Bleu@4	Meteor	RougeL	CIDEr	Spice
$VinVL_{large}$	R	8.9M	-	41.0	31.1	-	140.9	25.2
$SimVLM_{huge}$	G	1.8B	-	40.6	33.7	-	143.3	**25.4**
M^2 Transformer	R	-	80.8	39.1	29.2	58.6	131.2	22.6
RSTNet	G	-	81.8	40.1	29.8	59.5	135.6	23.0
SAT[1]	G	-	-	31.9	25.5	54.3	106.3	-
Up Down	R	-	79.8	36.3	27.7	56.9	120.1	21.4
GRIT (vg)	R+G	-	83.5	41.9	30.5	60.5	142.2	24.2
GRIT(4ds)	**R+G**	**-**	**84.2**	**42.4**	**30.6**	**60.7**	**144.2**	**24.3**

Table 1. Comparison of GRIT with baseline and other approaches in literature (in the paper)

Method	V.E Type	#VL Data	Performance Metrics					
			Bleu@1	Bleu@4	Meteor	RougeL	CIDEr	Spice
GRIT (vg)	R+G	-	83.5	41.9	30.5	60.5	142.2	24.2
GRIT (vg) Replicated*	**R+G**		**82.6**	**40.8**	**30.2**	**60.0**	**138.7**	**-**
GRIT(4ds)	R+G	-	84.2	42.4	30.6	60.7	144.2	24.3
GRIT (4ds) Replicated	**R+G**	**-**	**83.4**	**41.6**	**30.2**	**60.2**	**140.8**	**-**

Table 2: Comparison of GRIT with replicated GRIT
'*' evaluation done using checkpoint provided by author since, the dataset is hard preprocess and requires intensive cleaning.

Tokenization is performed using the SpaCy library, which efficiently converts text into tokens while preserving linguistic structures. The preprocessing stage ensures that the model works with clean and well-structured data. This enabled it to build stronger connections between visuals and text. We create a solid foundation for GRIT training, enabling the generation of captions that are both accurate and meaningful by pairing this approach with thoughtfully selected datasets and preparation steps.

5. Results

The authors evaluate the full set of captioning metrics: BLEU@N, METEOR, ROUGE-L, CIDEr, and SPICE. In their model, they set the dimension of each layer to 512, the number of heads to 8. They set a dropout rate of 0.2 on the output of each MHA and FFN sub-layer. They set 6 regional feature layers, 3 grid feature layers and 3 layers for the caption generator. Following earlier studies, the authors remove punctuation characters, convert all the captions to lower-case, and perform tokenization with the SpaCy. They build the vocabularies, without the words which appear fewer than five times in the training and validation splits. All the models are trained with the XE loss and finetuned by the CIDEr optimization. The optimizer uses a warmup learning rate schedule along with cosine annealing learning rates for XE. weight decay of 0.01. Training includes self-critical training, each lasting 10 epochs.

Table 1 shows the comparison of GRIT and GRIT pretrained on the 4 datasets (COCO, Visual Genome, Open Images, and Object365 with other baselines and models in the related work and literature review. The GRIT model was evaluated against a wide range of baseline methods, both with and without large-scale vision-language (V&L) pretraining, to demonstrate its superior performance in image captioning tasks. These baseline models encompass diverse approaches, including those relying on region-based and grid-based features, as well as combinations of both. GRIT achieved a CIDEr score of 142.2, outperforming all methods without V&L pretraining and most methods with V&L pretraining, except SimVLMhuge. GRIT, with the object detector pretrained on multiple datasets, further improved the CIDEr score to 144.2, surpassing SimVLMhuge

While trying to replicate the results, I trained the model only on the COCO 14 dataset instead of the 4 combined dataset which includes COCO14, Visual Genome, Open Images and Object365. The reason being that COCO itself being 12.6GB in training data poses training challenges. Despite this challenge, the replicated model achieved decent scores during evaluation with the validation set. Furthermore, the authors have provided the results of the model trained on just the COCO 14 dataset in Table 1 which gives a basis for a good and fair comparison of the replicated results. These results demonstrate the method's robustness and adaptability

even with limited training data. Table 2. shows the comparison of the results posted by the authors and the results obtained after replicating their work.

6. Possible Improvements and Results

The GRIT model employs Deformable DETR to extract region-based features, which are critical for detecting object-specific information inside an image. However, the quality and granularity of these attributes are dependent on the underlying object identification framework, which can essentially be improved. The GRIT model currently views grid and region features as complementary, but it does not dynamically modify their relative contributions. **Adaptive weighting algorithms** enable the model to prioritize grid or area data based on image complexity or the word being created.

In my experiments, I tested various hyperparameter configurations to fine-tune the GRIT model. While several adjustments led to minor improvements or degradation in certain scores, one particularly impactful change emerged when I altered the way region features, and grid features were combined in the ensemble. I observed that the training loss had improved at a better rate. I noticed notable improvements in the model's performance after adding a gating mechanism to the feature integration module. The gating mechanism dynamically adjusted the contributions of region and grid features based on the complexity of the image and the context of the generated caption, this improved the loss calculations. This adaptive approach allowed the model to better leverage complementary strengths of both feature types, leading to more accurate and contextually rich captions. This gating module could potentially improve CIDEr scores as well and provides the basis for future work.

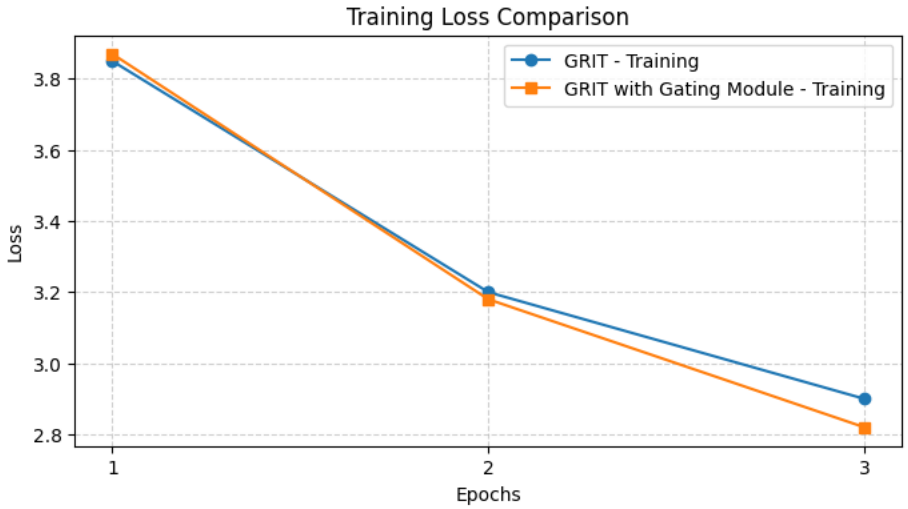

Table 3 Improvements in training loss in GRIT with Gating module

A learnable gating module can be introduced in the cross-attention layers of the caption generator. For each word in the caption, a weight for region features and a weight for grid features can be computed using a simple fully connected network as follows:

$$w_r, w_g = Softmax(FC(x)) \qquad (2)$$

where x is the current decoder state. These weights control the relative influence of region and grid features. Following this, attention scores can be used over region and grid features to compute dynamic weights as follows:

$$w_r = \frac{\sum_i \alpha_{r,i}}{\sum_i (\alpha_{r,i} + \alpha_{g,i})}, \qquad w_g = 1 - w_r \ (3)$$

Then a gating module can be added between the feature fusion and caption generation stages. The decoder's hidden state is then passed through the gating module to compute w_r, w_g. The cross-attention computation is then modified by taking the weighted sum of region features and grid features.

Next, I used the training and validation loss to compare the performance of the updated GRIT model with the SOTA GRIT versions. I analyzed the gating mechanism's performance and flexibility in handling different image complexity levels. The experiment demonstrated that the gating module's capacity to dynamically assemble region and grid characteristics greatly enhances the model's generating power and accuracy.

7. Code Repository

The code for this work can be found at https://github.com/rpp5524/CSE_597_rpp5524. I extend my gratitude to the authors for their clarity in the instructions in their codebase which helped me in understanding their research and guided me in successfully replicating the results.

References

1) Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2015). "Show and tell: A neural image caption generator". *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3156–3164.
2) Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., & Zhang, L. (2018). "Bottom-up and top-down attention for image captioning and visual question answering". Proceedings of the *IEEE Conference on Computer Vision and Pattern Recognition (CVPR*), 6077–6086.
3) Cornia, M., Stefanini, M., Baraldi, L., & Cucchiara, R. (2020). "Meshed-memory transformer for image captioning". Proceedings of the *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),* 10578–10587.
4) Wang, Z., Wang, W., Wang, Z., Zhou, Y., Li, H., Yang, M., & Lu, H. (2021). "SimVLM: Simple visual language model pretraining with weak supervision". *arXiv preprint arXiv:2108.10904*.
5) Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: "Swin transformer: Hierarchical vision transformer using shifted windows". In: Proceedings of the *IEEE International Conference on Computer Vision.* pp. 10012–10022 (2021)
6) Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: "End-to-end object detection with transformers". In: Proceedings of *European Conference on Computer Vision*. pp. 213–229 (2020)
7) Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Doll´ar, P., Zitnick, C.L.: "Microsoft coco: Common objects in context". In: Proceedings of *European Conference on Computer Vision*
8) Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: "Attention is all you need". *arXiv:1706.03762* (2017)
9) Karpathy: Karpathy/neuraltalk: Neuraltalk is a python+numpy project for learning multimodal recurrent neural networks that describe images with sentences., https://github.com/karpathy/neuraltalk