Computers, Materials & Continua
DOI: 10.32604/cmc.2022.029313
Article

Tech Science Press

Efficient Image Captioning Based on Vision Transformer Models

Samar Elbedwehy[1,*], T. Medhat[2], Taher Hamza[3] and Mohammed F. Alrahmawy[3]

[1]Department of Data Science, Faculty of Artificial Intelligence, Kafrelsheikh University, Egypt
[2]Department of Electrical Engineering, Faculty of Engineering, Kafrelsheikh University, Egypt
[3]Department of Computer Science, Faculty of Computer and Information Science, Mansoura, Egypt
*Corresponding Author: Samar Elbedwehy. Email: samarelbedwehy@ai.kfs.edu.eg
Received: 01 March 2022; Accepted: 12 April 2022

Abstract: Image captioning is an emerging field in machine learning. It refers to the ability to automatically generate a syntactically and semantically meaningful sentence that describes the content of an image. Image captioning requires a complex machine learning process as it involves two sub models: a vision sub-model for extracting object features and a language sub-model that use the extracted features to generate meaningful captions. Attention-based vision transformers models have a great impact in vision field recently. In this paper, we studied the effect of using the vision transformers on the image captioning process by evaluating the use of four different vision transformer models for the vision sub-models of the image captioning The first vision transformers used is DINO (self-distillation with no labels). The second is PVT (Pyramid Vision Transformer) which is a vision transformer that is not using convolutional layers. The third is XCIT (cross-Covariance Image Transformer) which changes the operation in self-attention by focusing on feature dimension instead of token dimensions. The last one is SWIN (Shifted windows), it is a vision transformer which, unlike the other transformers, uses shifted-window in splitting the image. For a deeper evaluation, the four mentioned vision transformers have been tested with their different versions and different configuration, we evaluate the use of DINO model with five different backbones, PVT with two versions: PVT_v1and PVT_v2, one model of XCIT, SWIN transformer. The results show the high effectiveness of using SWIN-transformer within the proposed image captioning model with regard to the other models.

Keywords: Image captioning; sequence-to-sequence; self-distillation; transformer; convolutional layer

1 Introduction

One of the most difficult problems in artificial intelligence is automatic caption synthesis for images, i.e., image captioning. Models of picture captioning, as shown in Fig. 1, not only handles computer vision difficulties of object recognition, but also describes relationships between them in

plain language. Automatic generation of image captions has a huge impact in the fields of information retrieval, accessibility for the vision impaired, categorization of images, and image indexing. There are applications for image captioning that do not have enough data and require methods to deal with this amount of data and help in producing a smaller model that does not need many parameters, one of these methods used transfer learning that exists in this reference [1]. The advances in research in both language modeling and object identification have a direct influence on the image captioning. Many researches in image classification, object detection and segmentation have achieved promising results using deep convolutional neural networks. The image captioning can be applied even to images with low resolution that can be improved by using the modern method for example this research [2].

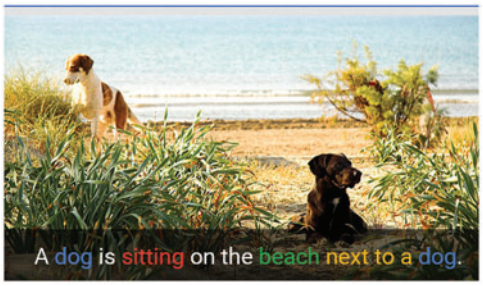

Figure 1: Example of image captioning

Modern convolutional neural networks require a massive computations and massive storage capacities to achieve good performance. This challenge has been thoroughly researched in recent years and one of possible solutions is using attention. Attention methods can be divided into two category; Global (or Soft) Attention which is placed on all positions in the image and Local (or Hard) Attention which is placed only on a few positions in the image. Transformer designs are built on a self-attention mechanism that learns the connections between sequence parts, which is a model that uses *attention* to improve the speed of training data. The transformer architecture can be used to detect objects, as it enables the model to distinguish between foreground and background objects in the encoder part to caption an image. Also, it can predict locations and categories for these objects that exist in the image. This aids image captioning models in predicting the bounding boxes and category labels for each object. Vision models with self-attention are classified into two categories [3]: The models which use single-head self-attention Transformer and the models which employ multi-head self-attention Transformer into their architectures. Self-Supervised Vision Transformers which have achieved significant success for CNN(convolutional neural network)-based vision tasks, have also been investigated for ViTs (vision transformer) [4] which is the first captioning work that used Transformers instead of standard convolutions in deep neural networks on large image datasets. They applied the original Transformer model with some changes on a sequence of image 'patches' flattened as vectors [5] and extended by the preceding works. We consider solving the problems of image captioning with four new proposed methods. We concentrated on image phase in the captioning stages as it is the first phase and it should be done by for extracting the features from the images with high accuracy. This paper is organized as follows; Section 2 for the related work and Section 3 for discussing the proposed framework. Section 4 presents the evaluation stage; then, we compare the results of the four proposed framework with the ViT, ResNet50 (residual neural network) and VGG16-LSTM (Visual Geometry Group–Long short term memory) model. Finally, Section 5 presents the conclusion and future work.

2 Background

The common machine translation tasks include classical operations such as translating words, aligning words, reordering . . . etc. Image captioning using deep learning models is now considered a part of the current machine translation breakthroughs as it *translates* the visual image into its corresponding textual description. The goal any machine translation process is to maximize p (T |S), which is used for estimating the parameters of an assumed probability distribution given some observed data, to translate a sentence S in the source language into the target language T. Machine captioning is very similar to machine translation as the encoder part commonly used in machine translation is replaced in image captioning by CNN instead of an RNN (*recurrent neural network)* [6]. This is because recent research has shown a rich representation of an input image can be obtained by embedding its contents in a fixed-length vector using CNN, which is very useful for many computer vision applications.

Vision Transformer now is the most popular model commonly used nowadays in computer vision; it uses attention mechanism to build enhanced vision models. We study in this paper using them as sub-models for the vision part of the image captioning model.

2.1 Vision Transformer

The Transformer model was first created to help solving natural language processing problems. It enables modeling long dependencies between input sequence elements and it supports parallel processing of sequences. The transformer model has been effectively recently in the field of computer vision. Transformers, unlike convolutional networks, are designed with minimal inductive biases and are naturally suited as set-functions. Transformer architectures are based on a self-attention mechanism that learns the relationships between elements of a sequence [5], which we'll go over in the next section.

2.1.1 Attention Mechanism on Vision Transformer

The attention mechanism is currently used increasingly in deep learning models with neural machine translation applications for improving the performance. Attention is how we can focus on different parts of an image or related words in one sentence. Fig. 2 is an example; Human visual attention allows us to pay attention to a specific part like focusing on two resolutions; one is high and the other is low. High resolution in the yellow box and the low is in the background. This attention makes detecting the whole image correct as in the yellow box indicates to an animal from the ear of the nose so the blanket and T-shirt doesn't mean anything for us and this attention can be done from the small patch of an image.

Figure 2: An example for human visual attention [7]

Sneha et al. proposed [8] four categories for attention depending on the number of sequences, abstraction, positions and representations as follows:

a) **Number of sequences** has three types: *distinctive* when key and query state belong to two distinct single input and single output sequences, *co-attention* when multiple input sequences are presented simultaneously and attention weights are learned to find the interactions between these inputs, and the last type is **self-attention** or **intra**-attention when the key and query state are from the same input sequence, this type is the most popular used in the recent research [8].
b) **Number of abstraction levels** has two types Single level when the weights of the attention are calculated only for the original input, and the multi-level when attention is applied sequentially on multiple abstraction levels of the input by using the output of certain level as the query state for the next higher level either a top-down or bottom-up [8].
c) **Number of positions,** this category defines the types of attention depending on the positions of input sequences where the attention is computed. These are three types. *Soft* or *global attention* [8] which is compute using all the data at on all positions in the input sequence *local* attention type that uses soft shading for focusing on a window in the image to calculate the attention more computationally-efficient and the last type in this category is *hard attention* which was proposed by Xu et al. [9] to compute attentions using stochastically sampled hidden states in the input sequence, i.e. On certain predicted positions of the input sequence.
d) **Number of representations;** this category has two types. *Multi-representation* in which different aspects of the input are captured through multiple feature representations and attention is used to assign importance weights to these different representations, and the other type is multi-dimensional that computes the relevance of each dimension of the input embedding vector, extracts contextual meaning of input dimensions.

2.1.2 Image Captioning with Attention

The most popular architectures that are used in most recent image captioning research are based on Encoder-Decoder and Transformer models. The attention is used in Encoder-Decoder model by converting first the input to a single fixed-length vector to reduce the length of the input, and then this vector is passed to the decoding step. The most commonly used Encode-Decoder is CNN-LSTM in which CNN represents the encoder and LSTM represents the decoder. It uses attention not only to select relevant regions and locations within the input image but also to reduce the complexity that with the size of the image. Attention is used to reduce this complexity and help to control it with considering the size of the image by dealing with selected regions at high resolution only. On contrary, Transformer model is built using multiple Encoder and Decoder layers stacked together and connected to each other. Transformer is based entirely on *self-attention* to compute representations of its input and output without using sequence-aligned RNNs or convolution. It means making a relation between tokens and their positions that exist in the same input.

2.1.3 Transformer Models

Several vision transformer models have been presented in the literature, we focus in this paper on four of 5hese models. In the following, we present the Transformer models that we used in this paper.

a) ViT

ViT was proposed by Dosovitskiy et al. [10]. Instead of using CNN in its architecture. It is directly uses the original Transformer architecture on image patches along with positional embeddings for image classification task. It has outperformed a comparable state-of-the-art CNN with four times fewer computational resources that most of classification tasks in computer vision used it [11–13].

b) PVT

PVT [14] inherits the advantages of both CNN and Transformer and try to solve their problems by *combining the advantages of CNN and ViT, without convolution. It* employes a variant of self-attention called SRA (Spatial-Reduction Attention) to overcome the quadratic complexity of the attention mechanism used in ViT. Also, unlike ViT, PVT can be used for dense prediction as it can be trained on dense partitions of an image to achieve high output resolution. Moreover, PVT uses a progressive shrinking pyramid to reduce the computations of large feature maps *The second version of PVT*, PVT_v2 [15] that solved some problems of fixed-size of PVT_v1 in position encoding, Also it reduces the complexity that come with PVT_v1, and finally it fixes losing of local continuity of the image to a certain extent due to using non-overlapping patches that exists in PVT_v1

c) DINO

DINO model [16] is a self-supervised vision transformer that applies knowledge distillation with no labels, where a student model is trained to match the output of a supervising teacher mode. In this transformer, teacher knowledge is distillated from the student during the training which is called dynamic training. The meaning of knowledge distillation is training a student network model to match the output of a teacher network model which means using a single trained model with identical architecture. Self-training means sending small annotations to a large unlabeled instances and this way help in improving the quality of the features and can work with soft-labels which referred to as knowledge distillation.

d) XCIT

XCIT [17] replaced the self-attention model with a transposed attention model called cross-covariance attention *(XCA)*. This is done as core self-attention operations which have relatively high time and memory complexity that increases with the number of input tokens or the patches. The *proposed cross-covariance attention modifies the transformer model by adding a transposed attention to deal with feature dimensions instead of token dimensions. These make reduction in the computational complexity to become better than self-attention and make dealing with flexibility with high resolution as XCIT focus on using a fixed number of channels instead of the number of tokens. The policy of this model is to take the features and divide them into heads and apply cross-covariance on each one of them separately then the weights that will be obtained from one of them in the tensors are retted*.

e) Swin Transformer

Feature maps in this model [18] are built with a hierarchical Transformer by integrating the patches of the image in deeper layers. The representation is computed with (SWIN) shifted **win**dow. It has linear computation complexity relative to the size of the input image due to computation of self-attention only within each local window. This strategy brings greater efficiency by reducing the computation of self-attention to non-overlapping local windows while also allowing for cross-window connection. *The model uses layers for patch merging with linear layer to reduce the number of tokens. Swin transformer is applied to get the features and those steps are repeated to produce the hierarchical structure. This model also change the standard multi-head self-attentions "MSA" with shifted window and this is the core of the model that makes the change in the results as the traditional connections uses window self-attention that lacks connections among windows and this effect on power. The advantage of this model exists in shift-window manner as it cure the previous layer that may lacks in window-based manner and this makes enhancement in power. The shifted window manner has much lower latency than the sliding window as it makes the connections between neighboring non-overlapping windows in the previous layer. The model takes the image as an input and divided it into patches with non-overlapping manner by using patch*

splitting module as exists in ViT [18]. It looks to every patch as a token. It applies SWIN transformer on each patch which is a modification on self-attention.

3 Related Work

Image captioning has been the subject in many research papers. For example; Lu et al. in [19] found that most attention methods ignore the phenomenon that words such as "the" or "an" in the captioning text cannot match the image parts and force each word to match an image part only. So the authors proposed an adaptive attention model that solve this problem and improve the mandatory matching between words and image areas. Huang et al. [20] proposed AoA (Attention on Attention) mechanism which uses conventional attention mechanisms for determining the relationship between queries and results of the attention. The model generates a vector for information and an attention gatewhich uses the attention results and the current context. After that, they apply element-wise multiplication to them by adding another attention. Then, the attended information and the expected useful knowledge are obtained. AoA was applied in this work for both the encoder and decoder of the image captioning model. The main advantage of the model that it counts objects of the same type accurately. Another mechanism presented that uses top-down soft attention was proposed by He et al. [21], it uses a top-down soft attention for simulating the human attention in captioning tasks and show that the behavior of human attention is different in seeing the image and in describing the tasksand there is a strong relevance between described and attended objects. This mechanism used CNN as feature encoder and integrated the soft attention model with the salience of the image by using a top-down soft attention mechanism for automatic captioning systems. In [22], Wang et al. proposed a novel method to model is the relevance between important parts of the image using a graph neural network, where features are extracted first from the image using deep CNN; then, GNN (Graph Neural Network) model is used to learn the relationship between visual objects. After that, the selection of the important, relevant objects is done by using a visual context-aware attention model. Finally, sentences are generated using an LSTM-based language model. This mechanism is used to guide attention selection by memorizing previously attended visual content as it takes into consideration the historical context information of the previous attention, besides it can learn relation-aware visual representations of image captioning. The work in Biswas et al. [23] is concerned with improving the level of image features. The authors proposed a novel image captioning system with a mechanism of bottom-up attention. The model combines low-level features like contrast, sharpness, contrast and colorfulness with high-level features like classification of motion or face recognition for detecting the attention regions in the image and for detecting regions that adapt to the bottom-up attention mechanism. Then, the weights of the impact factors for each region are adjusted by using a Gaussian filter. Then, a Faster RCNN (Region-based Convolutional Neural Network) is used for detecting objectsCompared to the "CNN + Transformer" paradigm, Liu introduced a CPTR (CaPtion TransformeR) in [24] which is a simple and effective method that totally avoids convolution operations. Due to the local operator essence of convolution, the CNN encoder has limitations in global context modeling, which can only be fulfilled by gradually enlarging the receptive field gradually as the convolution layers go deeper. However, the encoder of CPTR can utilize long-range dependencies among the sequentialized patches from the very beginning via a self-attention mechanism. During the generation of words, CPTR models "words-to-patches" attention in the cross attention layer of decoder which is proved to be effective.

4 The Proposed Framework

The task of captioning images can be separated into two sub-models; the first is a vision-based sub-model, which acts as a vision encoder that uses computer vision model to extract features from input images, and the second is a language-based sub-model, which acts as a decoder that translates the features and objects given by the image sub- model into natural sentences. Fig. 3 shows our proposed image captioning model, where the block labelled "Attention-based Transformer" refers to the vision transformers that extracts the features vectors from the input images and feeds them to the language decoder that generates the captions.

We propose for our work experimenting vision transformer (encoder) to evaluate them in order to find the most efficient transformer to be used for image captioning. We experimented four different attention-based visions Transformer for extracting the features from the images and for each Transformer, we applied it with its known, in the following subsections we illustrate more details on how these transformers have been used in our proposed model. Regarding the language-based (decoder), we used only (LSTM) Long Short Term Memory model to predict the sequences of the generated captions, from the feature vectors obtained after applying the vision transformer.

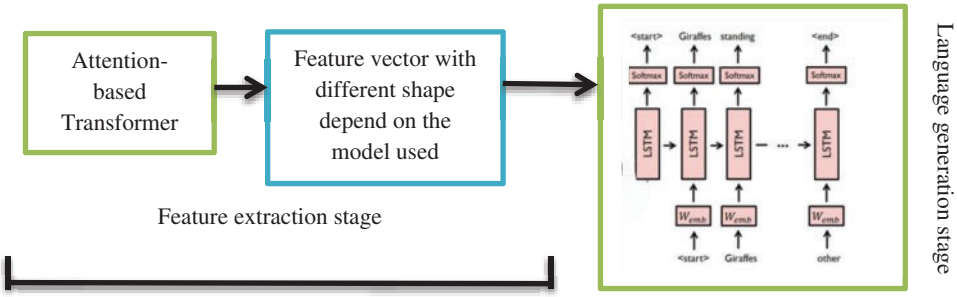

Figure 3: The proposed attention based

4.1 Vision-Transformer Encoder Model

We present here the different attention based vision transformer models that are tested in out proposed model. These transformers are DINO transformer [16] that have been used with different backbones in our proposed captioning model including (ResNet50, ViT s/8, ViT s/16, Xcit_meduim_24/p8 and ViT b/8).The second tested transformer is PVT, and we tested two different versions of it as presented in [14,15], and they have been tested with different configurations in our captioning model includingPVT_v1_Small, PVT_v1_Large, PVT_v2_b5 and PVT_v2_b2_Linear. The third transformer is XCIT model [17] and we tested its XCIT-Large version only. Finally, the fourth transformer is SWIN-transformer presented in [18], and we tested its SWIN-Large version.

4.2 The Proposed Language Model

For our proposed image captioning to predict the word sequences corresponding to the image contents, i.e., captions, we used a single language decoder model, as we focus only on this paper on evaluating the attention-based vision transformers. The fixed language decoder is as LSTM-based model that uses the feature vectors obtained from the vision transformer proceeding to generate the captions. In Fig. 4, the blue arrows correspond to the recurrent connections between the LSTM memories. All LSTMs share the same parameters. After receiving the image and all preceding words as defined by $P(S_t|I, S_0, S_1, \ldots S_{t-1})$, each word in the sentence is predicted by the LSTM model, which

has been trained, where I is an image, S its correct transcription, and W_e is word embedding. If we denote $S = S_0, S_1, \ldots S_N$ is a true sentence describing this image where S_0 a special start word like (*startseq*) and by S_N a special stop word like (*endseq*). For the image and each sentence word, a copy of the LSTM memory is produced so that all LSTMs have the identical settings and the output of the LSTM at time $t-1$ is supplied to the LSTM at time$t$. In the unrolled version, all recurrent connections are converted to feed forward connections [25].

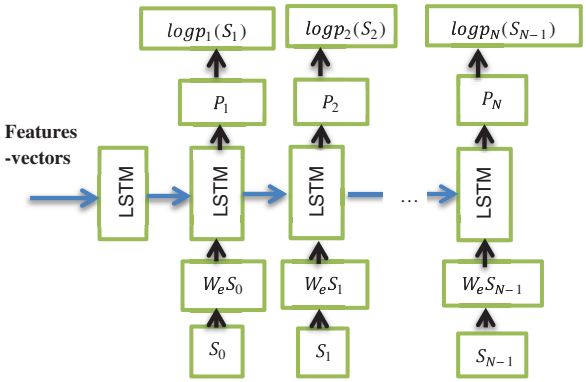

Figure 4: LSTM model structure [25]

5 Evaluation and Configurations

We evaluated our proposed model on the MS COCO (Microsoft Common Objects in Context) dataset [26], which is the most used image captioning benchmark. To be consistent with previous work, we used 30.000 images for training and 5000 images for testing. We trained our model in an end-to-end using Keras model using a laptop with one GPU (2060 RTX). Fig. 5 shows the Plot of the Caption Generation Deep Learning Model for using ViTs/16 and s/8 with DINO, where input1 is the input of image features, input2 is the text sequences or captions and dense is a vector of 384 elements that are processed by a dense layer to produce a 256 element representation of the image as all the settings are the same in five backbones with DINO, ResNet50, VGG16, PVT_v1, PVT_v2, ViT, XCIT and SWIN with their different methods except the shape of the image will be change upon the model. We used netron site [27] for plot the model by uploading the file of the model.

One is the VGG16 model with LSTM, the second is the PVT_v1 with their methods (Small and Large) model with LSTM and PVT_v2 with (b5 and b2_linear), and the other is the ResNet50-LSTM model and DINO with 5 different backbones (ResNet50, ViTs/8, ViTs/16, ViTb/8 and DINO-XCIT-m24_p8) with LSTM and XCIT, SWIN, ViT with LSTM.

The hyper parameter settings for our model are as follows:

Language model layers: 1-Layer LSTM, Word Embedding Dimensionality: 512, Hidden Layer Dimensionality: 256, Maximum Epochs: 20, LSTM dropout settings: [0.5], learning rate: [4e−4], **Optimizer:** We used Adam optimizer and the batch size is 16.

Vision Transformer, We compared different transformer models with different versions configurations as follow:

– DINO with five different backbones which are (ResNet50, ViTs16, ViTs8, ViTb8 and XCIT-m24_p8) with:-

1- ViT model which is proposed as a replacement of CNN that achieved results better than convolutional networks as it is applied directly to patches of the image as a sequence. Self-attention allows ViT to concatenate the information among image even in the lowest layers.

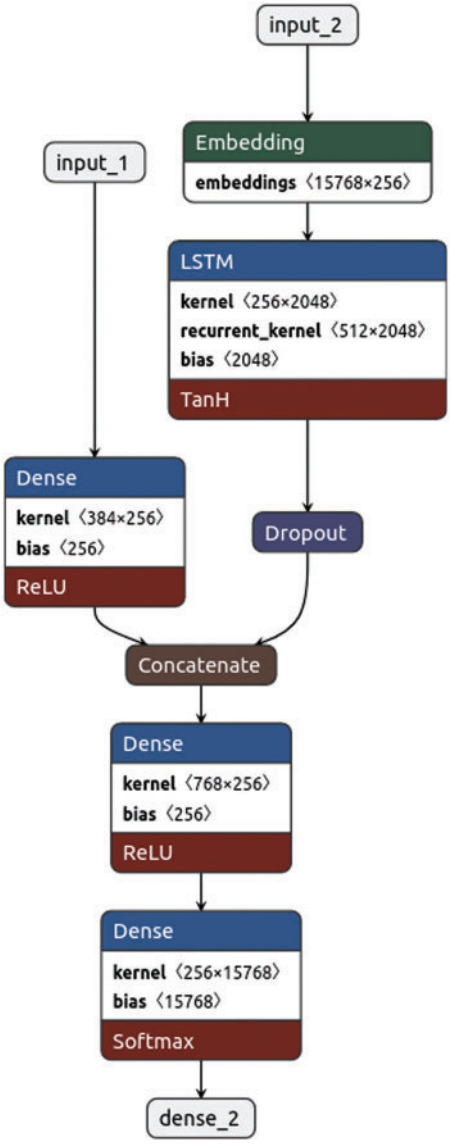

Figure 5: Plot of the caption generation deep learning model for DINO (vits8 and vits16)

2- PVT model in different two versions PVT_v1 and PVT_v2:

a-PVT_v1 was introduced in [14] to overcome the difficulty of porting transformer to various dense prediction tasks and it is unlike convolutional network, to control the scale of feature maps, PVT_v1 uses a progressive shrinking strategy by patch embedding layers instead of use different convolutional strides but PVT_v1 achieved a little improvement.

b- PVT_v2 [15] proposed to solve the problem with PVT_v1 in fixed-size of PVT_v1in position encoding that make a problem with processing the images with a flexible way, it reduces the complexity that come with PVT_v1 and it fix losing of local continuity of the image to a certain extent due to using non-overlapping patches that exists in PVT_v1. We compared also this model with its two different models (PVTv2-b5 and PVTv2-b2-Linear).

3- XCIT model proposed a solution for global interactions among all tokens by add a modification on the operations of self-attention which is a transposed on self-attention. This transposition occur in the interactions on features channels instead of tokens which make model has the flexibility and reduce the computational complexity as it has linear complexity and also achieve good results on images that have high resolutions.

4- SWIN model used shift-window in splitting the image instead of the traditional splitting that was sliding-window. This new manner effect on the performance as sliding-window can lack the connections between windows but with shift-window it use non-overlapping manner.

For evaluation purposes, we compare the results generated by the tested attention-based transformer models with other non-transformer models including ResNet50 and VGG16 which are types of convolutional network. In total, the tested vision models in our experiments are 14 different models. We compared and evaluated using each one of these different vision models as features extractor from images in the proposed image captioning model in several ways as explained next in this section.

The input images to the vision encoder are resized to square shaped images with different resolutions in each experiment as required by each of the used vision model. The size used in each of the tested model is shown in Tab. 1.

Table 1: Shape of the image for different models

Vision model	Resolution of the input image
ResNet50 [28]	2048 × 2048
VGG16 [29]	4096 × 4096
ViT_large_patch32_224 [10]	1024 × 1024
PVTv1-Small [14]	512 × 512
PVTv1-Large [14]	512 × 512
PVTv2-b5 [15]	512 × 512
PVTv2-b2-Linear [15]	512 × 512
DINO-ResNet50 [16]	2048 × 2048
DINO-ViTs8 [16]	384 × 384
DINO-ViTs16 [16]	384 × 384
DINO-ViTb8 [16]	768 × 768
DINO-XCIT-m24_p8 [16]	512 × 512
XCIT-Large_24_p16_224_dist [17]	768 × 768
SWIN-Large_patch4_window7_224 [18]	1536 × 1536

In our proposed evaluation process for evaluation the use of different attention-based vision transformers for image captioning, we considered some criteria. In the following we define each of these criteria and present the evaluation metric(s) used for evaluation it:

A. Efficiency of Image Captioning

This criterion aims to measure how efficient is the model in producing the captions for the input image. For this, criterion, we used a set of common metrics including BLEU (bilingual evaluation understudy)-1,2,3,4, METEOR (Metric for Evaluation of Translation with Explicit ORdering), ROUGE (Recall-Oriented Understudy for Gisting Evaluation), CIDEr (Consensus-based Image Description Evaluation) and SPICE (Semantic Propositional Image Caption Evaluation) scores, which are denoted as B1,2,3,4, M, R, C and S, respectively. BLEU scores [30] are used in text translation for evaluating translated text against one or more reference translations. We compared each generated caption against all of the reference captions for the image and considered very popular for captioning tasks. BLEU scores for 1, 2, 3 and 4 are calculated for cumulative n-grams. SPICE metric [31] is a more meaningful evaluation also for the semantics of generated captions. ROUGE [32] is used for text summary originally. METEOR [32] is another metric for the evaluation of machine translation output slightly younger than BLEU. CIDER and SPICE are specially designed for image captioning, CIDER [33] uses TF-IDF weighting term, it works with calculate the frequency of a word in a certain corpus which indicates for the character or semantic meaning of the word but SPICE uses word tuples to calculate the intersection of the candidate and ground truth captions.

The evaluation results of the image captioning model using these metrics for all the tested vision models is shown in Tab. 2 where the scores of our tested models with DINO, PVT_v1 and PVT_v2 indicates that using it for extracting the features of the images, improves the performance of the captioning model as the scores of the three backbones of DINO with LSTM is better than using VGG16 and ResNet50-LSTM model which are types of CNN models. ViTb/8 backbone and LSTM, with smaller patches of ViTs enhancing the quality of the generated features. XCIT also achieves better results than the previous models due to *adding transposed attention to deal with feature dimensions instead of token dimensions*. But the most effective model is SWIN-transformer which makes the result better than all the other models due to changing the manner of splitting the patched using window-shifting instead of windows-sliding that also reduce the computational complexity and time rather than other models.

Table 2: Image captioning efficiency measurements comparisons on MSCOCO. All models are fine-tuned with self-critical training

Model-name	B1	B2	B3	B4	M	R	C	S
VGG-16	0.489	0.295	0.174	0.102	0.160	0.338	0.470	0.123
ResNet50	0.505	0.307	0.182	0.109	0.166	0.348	0.504	0.127
ViT	0.508	0.310	0.183	0.107	0.171	0.353	0.525	0.135
PVT_v1_Small	0.511	0.314	0.188	0.110	0.173	0.357	0.541	0.138
PVT_v1_Large	0.516	0.318	0.190	0.113	0.175	0.357	0.548	0.140
PVT_v2_b5	0.525	0.326	0.194	0.114	0.180	0.366	0.573	0.143
PVT_v2_b2_Linear	0.519	0.320	0.192	0.113	0.178	0.363	0.564	0.143
DINO-ResNet50	0.522	0.324	0.197	0.117	0.176	0.363	0.548	0.138

(Continued)

Table 2: Continued

Model-name	B1	B2	B3	B4	M	R	C	S
DINO-ViTs/16	0.520	0.322	0.193	0.115	0.175	0.361	0.543	0.137
DINO-ViTs/8	0.523	0.329	0.199	0.118	0.180	0.366	0.576	0.142
DINO-ViTb/8	0.526	0.329	0.199	0.118	0.180	0.368	0.573	0.144
DINO-xcit_medium_24_p/8	0.524	0.329	0.199	0.119	0.179	0.366	0.568	0.142
Xcit	0.530	0.327	0.195	0.114	0.182	0.371	0.582	0.147
Swin-Transformer	**0.554**	**0.354**	**0.216**	**0.129**	**0.192**	**0.388**	**0.641**	**0.158**

Fig. 6 show the comparison of BLEU scores for using the tested models and Fig. 7 shows the comparison between METEOR, ROUGE, Cider and SPICE for the same models.

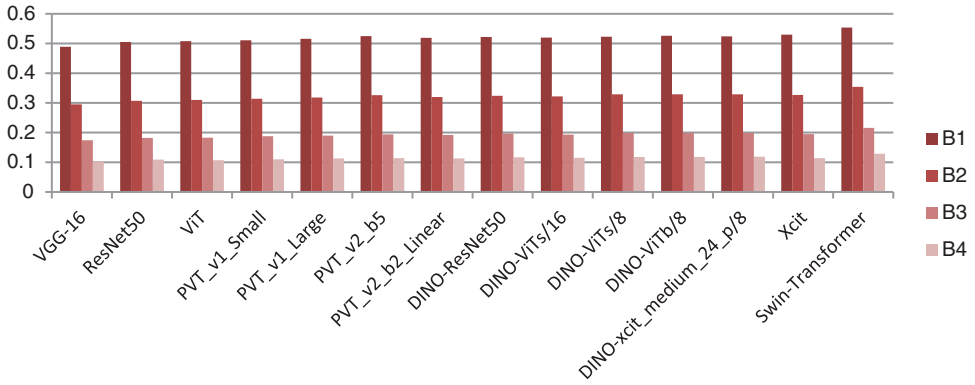

Figure 6: Comparison between bleu scores for 11 models

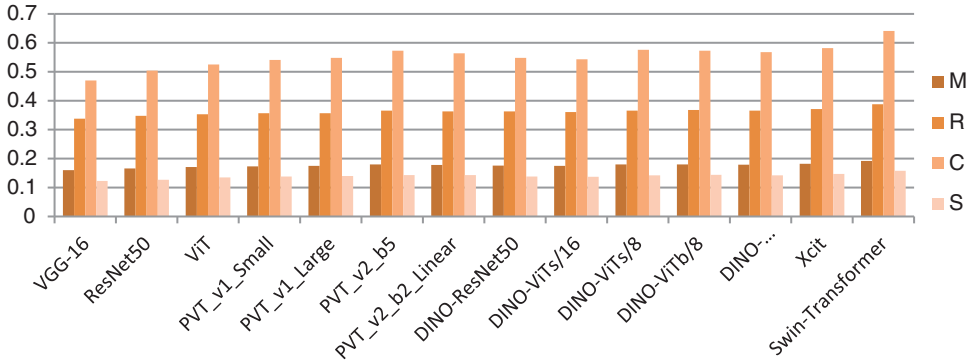

Figure 7: Comparison between METEOR, ROUGE, cider and SPICE for 11 models

As shown in Fig. 7, the performance of the five backbones of DINO is better than using VGG16 and ResNet50-LSTM models and little better than PVT_v1 and PVT_v2. XCIT in better than all above models but again SWIN-transformer is the best model among all the others.

B. Space Requirements

This criterion is particularly important in case of using the image captioning model on memory-constrained devices. In our evaluation, we used to metrics for this criterion: the number of parameters and the size of the trained model. Finding the number of parameters in its structure of the model is a common metric to evaluate the space requirements of the model as the smaller this number the less the size of the model. In case of the proposed captioning process, the language decoder is fixed in all the experiment and only the vision model is changing. Tab. 3 the number of parameters for both the vision model and the corresponding image captioning model for all the 14 tested models are shown, where Dino with ViTs16 and ViTs8 is have shown the least number of the parameters compared with the others, while VGG has the is the largest number.

Table 3: Number of parameters for each model used in the captioning model

Vision model used	No of params in the vision model	No of params for captioning model
ResNet50	25.6	10,385,304
VGG16	138	10,909,592
ViT	307	10,123,160
PVTv1-Small	24.5	9,992,088
PVTv1-Large	61.5	9,992,088
PVTv2-b5	82	9,992,088
PVTv2-b2-Linear	22.6	9,992,088
DINO-ResNet50	23	10,385,304
DINO-ViTs8	**21**	**9,959,320**
DINO-ViTs16	**21**	**9,959,320**
DINO-ViTb8	85	9,992,088
DINO-XCIT-m24_p8	84	9,992,088
XCIT	189	10,057,624
SWIN	197	10,254,232

We also measured the size of each trained image captioning model for the 14 tested vision models. As shown in Tab. 4 the image captioning model using DINO transformer with the ViTs8 and ViTs16 have the smallest size, while the largest model is the one using VGG16 model, while SWIN-based model, the most efficient in image captioning, is about 3% more in size. These results agree with the results obtained by using the number of parameters metric.

Table 4: Sizes of the tested of image captioning models

Model	Model size(MB)
ResNet50	124.7
VGG16	131
ViT	121.5

(Continued)

Table 4: Continued

Model	Model size(MB)
PVTv1-Small	120
PVTv1-Large	120
PVTv2-b5	120
PVTv2-b2-Linear	120
DINO-ResNet50	124.7
DINO-ViTs8	**119.6**
DINO-ViTs16	**119.6**
DINO-ViTb8	120.7
DINO-XCIT-m24_p8	120
XCIT	120.7
SWIN	123.1

C. Time Evaluation

For each of the 14 tested vision model, we compared the time taken for training for each model to get the best epoch for captioning. As shown in Fig. 8, PVTv2-b5 and PVTv2-b2- was the fastest in training as they took the least training time (7.4 h), and PVTv1-Large was the slowest, as it finished training in 17.5 h while SWIN, which is the most efficient in producing caption, has taken 10 h.

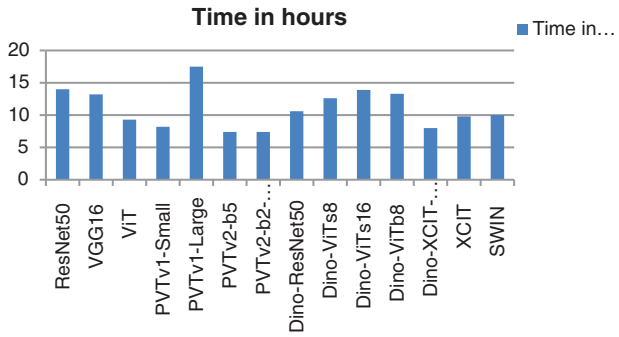

Figure 8: Training times of the tested image captioning models

D. Performance Evaluation

The performance evaluation is an important criterion as it reflects how fast the image captioning model in generating captions of an input image is. We used the Flops metric to evaluate the performance for the 14 tested image captioning models, as FLOPs are used to describe how many operations are required to run a single instance of a given model. The more the FLOPs the more time model will take for inference, i.e., the better models have a smaller number of FLOPS. Tab. 5 shows number of FLOPS of each of the 14 tested image captioning models. The worst model VGG16 was the worst, while DINO-ViTs8 and DINO-ViTs16 were the fastest models in generating the captions while SWIN model was a bit slower (2.4% slower).

Table 5: Number of FLOPS for the tested image captioning models

Captioning model using	FLOPs
ResNet50	12.46 m
VGG16	12.99 m
ViT	12.20 m
PVTv1-Small	12.07 m
PVTv1-Large	12.07 m
PVTv2-b5	12.07 m
PVTv2-b2-Linear	12.07 m
DINO-ResNet50	12.46 m
DINO-ViTs8	**12.04 m**
DINO-ViTs16	**12.04 m**
DINO-ViTb8	12.14 m
DINO-XCIT-m24_p8	12.07 m
XCIT	12.14 m
SWIN	12.33 m

Different samples of the image captioning produced by the tested models are shown in Fig. 9, on the left are the given images, on the right are the corresponding captions. The captions in each box are from the same model sample. We show the captions from all the tested models. Most of the captioning sentences are more accurate than using the ResNet50-LSTM and VGG16 models, especially using SWIN-transformer.

Image Caption with LSTM model

ResNet50 : two birds are sitting on the branch of tree
VGG16 : two birds are standing in the grass with an open sky
ViT : two birds are standing in the middle of the dirt
PVT_v1-Small : two birds are sitting on the branch of the tree
PVT_v1-Large : bird perched on branch in the woods
PVT_v2-b5 : bird perched on branch of tree
PVT_v2-b2-Linear : bird is perched on branch with its wings
DINO-ResNet50 : group of birds are sitting on branch
DINO-ViTs8 : two birds sitting on top of the top
DINO-ViTs16 : the bird is standing in the tree
DINO-ViTb8 : bird sitting on branch in the background
DINO-XCIT-m24_p8 : two birds are perched on tree branch
XCIT: bird is standing on the branch of tree
SWIN: flock of birds are standing on branch

Figure 9: (Continued)

ResNet50 : zebra standing in the grass with trees in the background
VGG16 : two giraffes standing in the grass in the wild
ViT : zebra standing in the grass with trees in the background
PVT_v1-Small : zebra standing in the grass with trees in the background
PVT_v1-Large : zebra standing in the grass with trees in the background
PVT_v2-b5 : zebra standing in the grass next to tree
PVT_v2-b2-Linear : zebra standing in the grass with trees in the background
DINO-ResNet50 : zebra standing in the grass with its head in the background
DINO-ViTs8 : zebra standing in the grass with trees in the background
DINO-ViTs16 : zebra is standing in the grass with its head
DINO-ViTb8 : zebra standing in the grass next to tree
DINO-XCIT-m24_p8 : zebra standing in the grass with some trees
XCIT: zebra standing in the grass with some grass in the background
SWIN: zebra standing in the grass with trees in the background

ResNet50 : man is playing frisbee in field
VGG16 : two men playing frisbee in field
ViT : group of people playing frisbee in park
PVT_v1-Small :man in the air with frisbee in the air
PVT_v1-Large : young boy is playing baseball in the park
PVT_v2-b5 : young boy playing frisbee in park
PVT_v2-b2-Linear : man holding frisbee in the air
DINO-ResNet50 : man in white shirt and white shirt playing frisbee
DINO-ViTs8 : two people are playing frisbee on the grass
DINO-ViTs16 : man in red shirt and white shirt playing with Frisbee
DINO-ViTb8 : young boy in the grass playing frisbee
DINO-XCIT-m24_p8 : man in white shirt playing Frisbee
XCIT: young boy is playing frisbee in the grass
SWIN: two boys playing frisbee in the grass

Figure 9: Samples for comparison produced by the tested image captioning models

6 Conclusion

In this paper, we focused on increasing image captioning performance by extracting the features from images using an attentions-based vision transformer model acting as an encoder to extract the features; followed by an LSTM-based decoder acting as a language model to build the corresponding caption. For finding the best vision transformer encoder for this model, we tested 4 relatively a new transformer: self-distillation transformer model called DINO with five different backbones, PVT transformer with two versions using (two different methods for both), the third is XCIT transformer and the last transformer is SWIN model. In our experiments, we built different image captioning models for using each of these transformers with different configurations/versions and make a comparison between them and other image captioning models that use non-transformer-based convolutional networks which are ResNet50 and VGG16. It was found that an image captioning model using SWIN transformer as a vision transformer is significantly efficient in generating the image captions over all the other models based on BLEU, METEOR, ROUGE, CIDEr, and SPICE metrics. This efficiency is due to the manner of SWIN transformer in splitting the image using shifted-window instead traditional manner. In terms of the performance in producing the captions, the DINO -based image captioning model is the fastest as its structure has less number of FLOPS. On the other hand, PVT-based image captioning model is the fastest in training in time, especially by using PVT_v2_b5. Finally, image captioning model based on XCIT transformer shows very good results in terms of its efficiency in generating the captions as it comes after the SWIN model, while it has a slightly smaller size, and it shows a slightly faster performance and takes less time for training.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] J. Zhang, Z. Wang, Y. Zheng and G. Zhang, "Design of network cascade structure for image super-resolution," *Journal of New Media*, vol. 3, no. 1, pp. 29–39, 2021.

[2] X. R. Zhang, J. Zhou, W. Sun and S. K. Jha, "A lightweight CNN based on transfer learning for COVID-19 diagnosis," *Computers, Materials & Continua*, vol. 72, no. 1, pp. 1123–1137, 2022.

[3] L. Liu, J. Liu and J. Han, "Multi-head or single-head? An empirical comparison for transformer training," 2021. [Online]. Available: https://arxiv.org/abs/2106.09650.

[4] Y. Chu, X. Yue, L. Yu, M. Sergei *et al.,* "Automatic image captioning based on ResNet50 and LSTM with soft attention," *Wireless Communications and Mobile Computing*, vol. 2020, pp. 1–7, 2020.

[5] S. Khan, M. Naseer, M. Hayat, S. Waqas *et al.,* "Transformers in vision: A survey," 2021. [Online]. Available: https://arxiv.org/abs/2101.01169.

[6] C. Lala, P. Madhyastha, J. K. Wang and L. Specia, "Unraveling the contribution of image captioning and neural machine translation for multimodal machine translation," *The Prague Bulletin of Mathematical Linguistics, De Gruyter Open*, vol. 108, no. 1, pp. 197–208, 2017.

[7] L. Weng, "Attention? Attention. lil'log," 2018. [Online]. Available: https://lilianweng.github.io/lil-log/2018/06/24/attention-attention.html.

[8] S. Chaudhari, V. Mithal, G. Polatkan and R. Ramanath, "An attentive survey of attention models," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 12, no. 5, pp. 1–32, 2021.

[9] K. Xu, J. Ba, R. Kiros, K. Cho *et al.,* "Show, attend and tell: Neural image caption generation with visual attention," in *Int. Conf. on Machine Learning, PMLR*, pp. 2048–2057, 2015.

[10] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn *et al.,* "An image is worth 16x16 words: Transformers for image recognition at scale," 2010. [Online]. Available: https://arxiv.org/abs/2010.11929.

[11] S. Ren, K. He, R. Girshick and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[12] A. Kirillov, R. Girshick, K. He and P. Dollár, "Panoptic feature pyramid networks," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, pp. 6399–6408, 2019.

[13] K. He, G. Gkioxari, P. Dollár and R. Girshick, "Mask R-CNN," in *Proc. of the IEEE Int. Conf. on Computer Vision*, pp. 2961–2969, 2017.

[14] W. Wang, E. Xie, X. Li, D. Fan *et al.,* "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," in *Proc. of the IEEE/CVF Int. Conf. on Computer Vision*, pp. 568–578, 2021.

[15] Z. Liu, J. Ning, Y. Cao, Y. Wei *et al.,* "Video swin transformer," 2021. [Online]. Available: https://arxiv.org/abs/2106.13230.

[16] M. Caron, H. Touvron, I. Misra, H. Jégou *et al.,* "Emerging properties in self-supervised vision transformers," in *Proc. of the IEEE/CVF Int. Conf. on Computer Vision*, pp. 9650–9660, 2021.

[17] A. Ali, H. Touvron, M. Caron, P. Bojanowski *et al.,* "Xcit: Cross-covariance image transformers," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[18] Z. Liu, Y. Lin, Y. Cao, H. Hu *et al.,* "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. of the IEEE/CVF Int. Conf. on Computer Vision*, pp. 10012–10022, 2021.

[19] J. Lu, C. Xiong, D. Parikh and R. Socher, "Knowing when to look: Adaptive attention via a visual sentinel for image captioning," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 375–383, 2017.

[20] L. Huang, W. Wang, J. Chen and X. Wei, "Attention on attention for image captioning," in *Proc. of the IEEE/CVF Int. Conf. on Computer Vision*, pp. 4634–4643, 2019.

[21] S. He, H. R. Tavakoli, A. Borji and N. Pugeault, "Human attention in image captioning: Dataset and analysis," in *Proc. of the IEEE/CVF Int. Conf. on Computer Vision*, pp. 8529–8538, 2019.

[22] J. Wang, W. Wang, L. Wang, Z. Wang *et al.,* "Learning visual relationship and context-aware attention for image captioning," *Pattern Recognition*, vol. 98, no. 4, pp. 107075, 2020.
[23] R. Biswas, M. Barz and D. Sonntag, "Towards explanatory interactive image captioning using top-down and bottom-up features, beam search and re-ranking," *KI-Künstliche Intelligenz*, vol. 34, no. 4, pp. 571–584, 2020.
[24] W. Liu, S. Chen, L. Guo, X. Zhu *et al.,* "Cptr: Full transformer network for image captioning," 2021. [Online]. Available: https://arxiv.org/abs/2101.10804.
[25] B. Tarján, G. Szaszák, T. Fegyó and P. Mihajlik, "Investigation on N-gram approximated RNNLMs for recognition of morphologically rich speech," in *Int. Conf. on Statistical Language and Speech Processing*, Springer, Cham, pp. 223–234, 2019.
[26] M. Caron, I. Misra, J. Mairal, P. Goyal *et al.,* "Unsupervised learning of visual features by contrasting cluster assignments," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9912–9924, 2020.
[27] Common objects in Context, Retrieved from. https://cocodataset.org/.
[28] Netron: A visualizer for neural network, deep learning and machine learning models, Retrieved from https://netron.app/.
[29] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.
[30] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014. [Online]. Available: https://arxiv.org/abs/1409.1556.
[31] A. B. Sai, A. K. Mohankumar and M. M. Khapra, "A survey of evaluation metrics used for NLG systems," 2020. [Online]. Available: https://arxiv.org/abs/2008.12009.
[32] P. Anderson, B. Fernando, M. Johnson and S. Gould, "Spice: Semantic propositional image caption evaluation," in *European Conf. on Computer Vision*, Springer, Cham, pp. 382–398, 2016.
[33] R. Vedantam, C. L. Zitnick and D. Parikh, "Cider: Consensus-based image description evaluation," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 4566–4575, 2015.