CS_Morgan at ImageCLEFmedical 2022 Caption Task: Deep Learning Based Multi-Label Classification and Transformers for Concept Detection & Caption Prediction

Md Mahmudur Rahman[a] and Oyebisi Layode[a]

[a] *Computer Science Department, Morgan State University, 1700 East Cold Spring Ln, Baltimore, MD, 21251, USA*

Abstract

This paper describes the participation of Morgan_CS in both Concept Detection and Caption Prediction tasks under the ImageCLEFmedical 2022 Caption task. The task required participants to automatically identifying the presence and location of relevant concepts and composing coherent captions for the entirety of an image in a large corpus which is a subset of the extended Radiology Objects in COntext (ROCO) dataset. Our implementation is motivated by using encoder-decoder based sequence-to-sequence model for caption and concept generation using both pre-trained Text and Vision Transformers (ViTs). In addition, the Concept Detection task is also considered as a multi concept labels classification problem where several deep learning architectures with "sigmoid" activation are used to enable multi-label classification with Keras. We have successfully submitted eight runs for the Concept Detection task and four runs for the Caption Prediction task. For the Concept Detection Task, our best model achieved an F1 score of 0.3519 and for the Caption Prediction Task, our best model achieved a BLEU Score of 0.2549 while using a fusion of Transformers.

Keywords 1

Medical imaging, Image annotation, Deep learning, Caption prediction, Concept detection, Transformer, Multi-label classification.

1. Introduction

Automatic caption and concept generation of biomedical images is a challenging AI problem which requires both techniques from Computer Vision to interpret the concepts of the image and understanding the relationship among concepts and techniques from natural language processing (NLP) to generate the textual description. During the past decade, immense progress has been made to automatically generate clinical reports and captions from medical images using Deep Learning to assist clinicians in accurate decision-making and speed up the diagnosis process [1-3]. Most of the existing literature in medical domain are based on a Convolutional Neural Network (CNN) - Recurrent Neural Network (RNN) framework where a pre-trained CNN is used to encode the images and a RNN is used to either encode the text sequence generated so far, and/or generate the next word in the sequence [1].

Recently, transformer based pretrained language models have demonstrated state-of-the-art performance on several Natural Language Processing (NLP) tasks, for example, Bidirectional Encoder Representation from Transformers (BERT)-like architectures [4] has been proven quite effective in NLP and successfully adapted in Computer Vision field [5]. Capitalizing on these advances in NLP and

[1] CLEF 2022 – Conference and Labs of the Evaluation Forum, September 5–8, 2022, Bologna, Italy
EMAIL: md.rahman@morgan.edu (A. 1); oylay1@morgan.edu (A. 2)
ORCID: 0000-0003-0405-9088 (A. 1); 0000-0002-6924-0390 (A. 2)

Computer Vision, the medical imaging field has also witnessed growing interest for Transformers in segmentation, detection, classification, reconstruction, synthesis, registration, clinical report generation, and other tasks [6]. Recent works have shown that these Transformer modules can fully replace the standard convolutions in deep neural networks by operating on a sequence of image patches and capturing the global context of an input image, giving rise to ViTs [7], which push the state-of-the-art in numerous computer vision tasks. Core components behind the success of ViTs that are self-attention which determines the relative importance of a single token (patch embedding) with respect to all other tokens in the sequence and multi-head self-attention consists of multiple self-attention blocks (heads) concatenated together channel-wise to model complex dependencies between different elements in the input sequence.

Inspired from this transition and motivated by the successful adaptation of Transformers in medical domain, we investigated encoder-decoder based Text and ViTs for sequence-to-sequence model generation in both the Concept Detection and Caption Prediction tasks based on our participation in the ImageCLEFmedical 2022 Caption task [8] under the ImageCLEF 2022 evaluation campaign [9]. These tasks requires participants to automatically identifying the presence and location of relevant concepts and composing coherent captions for the entirety of an image in a large corpus which is a subset of the extended ROCO dataset [10]. Especially, we investigated a seq2seq model for medical image captioning that employs both pre-trained CNN (e.g., CheXNet [11]) and ViT as the encoder and the pre-trained BERT as the decoder. In addition, the concept detection task is also considered as a multi-label concept classification problem where scikit-learn based MultiLabelBinarizer class is used with several CNN architectures with "sigmoid" activation at the end of the network architecture to enable us to perform multi-label classification with Keras [12].

2. Methodologies

In this section, we describe the encoder-decoder based transformer architecture for sequence-to-sequence modeling, the multi-label classification based on using different CNN models approach for concept detection and training and fusion strategies for the submission.

2.1. Transformer-based Encoder-Decoder Models

The captioning model utilized in this task follows the modular transformer architecture consists of the following three models.

2.1.1. Encoder

The goal of the ‘transformer’ encoder used in this study was to transform visual features obtained from a pretrained CheXNet model into an attended projection representative of features of the image that are important towards the correct sequence generation. The CheXNet model consists of 5 conv blocks of multiple convolutional, ReLU activation, batch normalization, concatenation, and average pooling layers [11]. To obtain an attended projection of the image fed into the pretrained CheXNet model, features are fetched from the last layer of the 5^{th} conv block of the pretrained model. The features are reshaped from a [7 X 7 X 1024] dimension to give 49 different 1024 shaped embeddings at [49 X 1024] for each image. The 1024 dim embeddings are down sampled to [49 X 512] by passing it through a fully connected layer with 512 neurons. The features are further fed into a multi-head attention layer that learns a matrix [512 X 512] and attention weight [49 X 49].

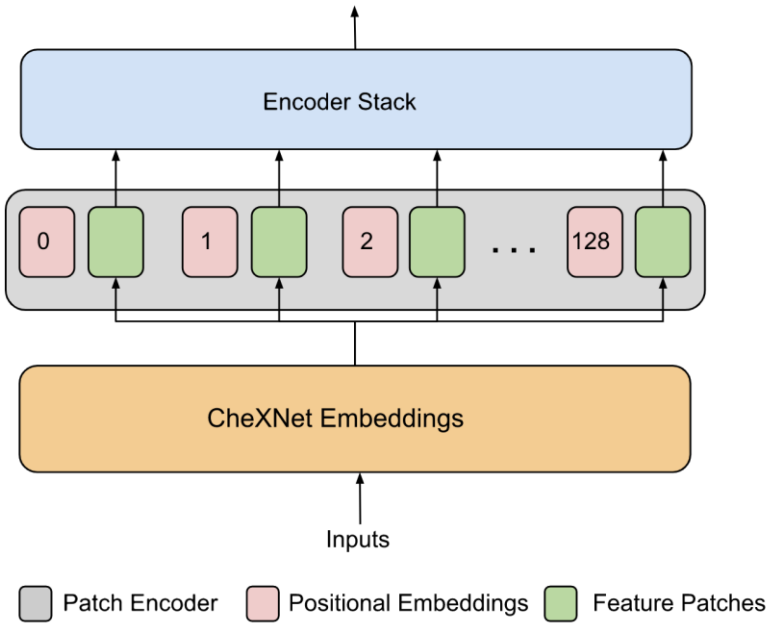

Figure 1: Encoder block for the vision transformer model 1 (ViT1).

2.1.2. Vision Transformer 1 (ViT1)

The concept of ViT is an extension of the original concept of Transformers, the latter of which is described earlier in this article as text transformer [7]. It is only the application of Transformer in the image domain with slight modification in the implementation to handle the different data modality. More specifically, a ViT uses different methods for tokenization and embedding. However, the generic architecture remains the same.

A modified vision transformer architecture (ViT1) is utilized in this study where the operation of this architecture involves a patch encoder that transforms pretrained CheXNet embeddings into patch embeddings that also carry information about the position of the feature patches. The steps involved in the patch encoder are detailed below (Fig. 1):

1. Split features into feature patches.
2. Obtain higher dimensional embeddings from each feature patch.
3. Add positional embeddings to keep the feature order information.

The [49 X 1024] features obtained from the CheXNet pretrained model [11] are flattened into a 50176 dim feature representation of the image. The patch encoder splits this feature representation into [128 X 392] feature patches. The goal of this architecture was to apply the knowledge already gained from pretrained models in Vision Transformers. The final output of the patch encoder is a combination of the [128 X 392] feature projections and the [1 X 392] dim positional embeddings. The output of the patch encoder is fed into a ViT encoder which includes a multi-head attention layer, a skip connection and an intermediate dense layer that projects the visual feature representation into the specified dimension size.

2.1.3. Vision Transformer 2 (ViT2)

The set-up of the ViT2 encoder (Fig. 2) consists of a patch encoder that receives an input of 224 X 224 images and produces a dense projection of the patches and a positional embedding representative of each patch's locality. 16 X 16 X 3 overlapping patches are obtained from the image and are flattened into a 768-dimensional array and fed into a 512- neuron fully connected layer which results in a 512-dimensional array. The patch projections and positional embeddings are fed into the encoder stack which is a layer of 8 encoders consisting of a multi-head attention layer that provides the attended representation of the features, a skip connection and an intermediate dense layer that projects the visual feature representation into the specified dimension size.

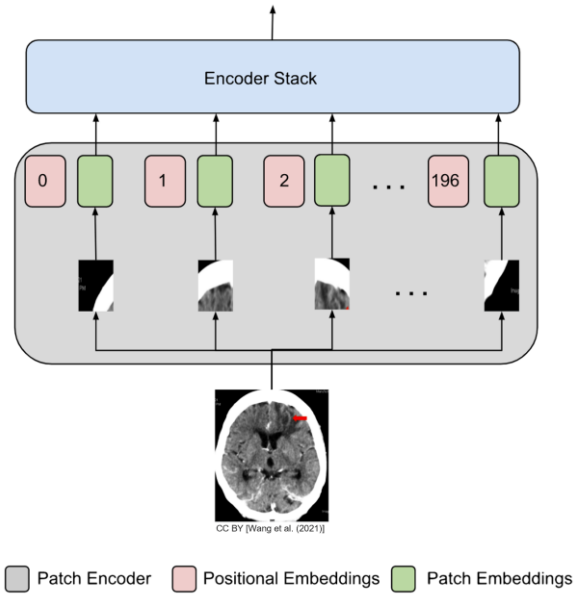

Figure 2: Encoder block for the vision transformer model 2 (ViT2).

2.1.4. Decoder

The textual decoder generates captions based on the visual representation provided by the encoder. The decoder feeds the image encodings alongside the positional encodings of the words in the generated captions into multiple attention heads that compute a scaled dot product of the query sequence to all other sequence entries. The decoder predicts auto-regressively the future track positions. The decoder query is compared against the encoder keys, value and against the previous decoder prediction.

The input of the decoder is an array consisting of the encoder output, a tokenized text sequence and the positional embeddings of the tokenized text sequence. The decoder consists of two masked multi-head attention layers, feed forward neural modules and a classification head. The output of the decoder layer is a matrix [maximum sequence length -1 X vocabulary size], the output of the decoder is right shifted by ensuring the prediction for an input sequence length 1 is 29 for an example maximum sequence length 30. This enables a concatenation of the previous input with the classification at position 0 of the generated output. This concatenation can be used as the text token input for the next iteration of the prediction. This prediction loop continues until an end flag is predicted at position 0 of the generated output.

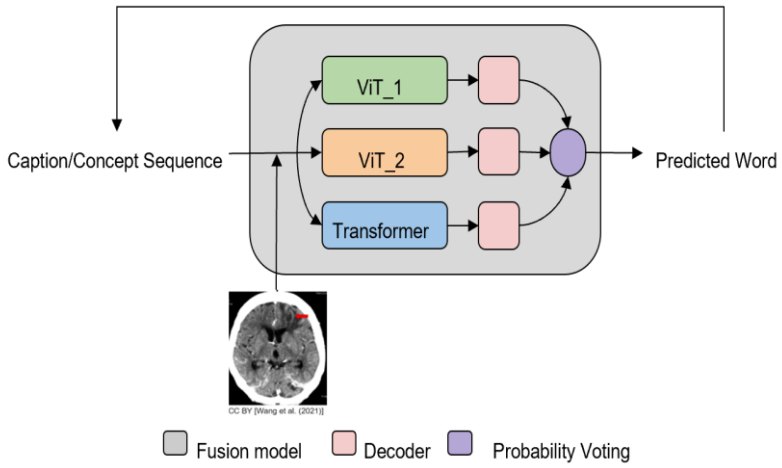

Figure 3: Process diagram of the fusion model

2.1.5. Fusion of Transformers

In the bid to utilize the varying information learned by all models, the trained models were combined in a late fusion. The fusion predictor working principle involved a sequence generation based on the most likely word from the words generated by the models. At each iteration of the sequence generation, the three models were fed with the same image and the attended embeddings or encodings were obtained as the results for each of the encoder models, the encodings and the text embeddings were passed into separate decoders to generate three possible words that could fit the current iteration of the sequence generation. The most likely word is selected from the generated words and appended to the generated sequence for the next iteration. In the prediction of the next word in the caption sequence, all models were fed with the same images and same texts. However, only the predictions with maximum probability were accepted by the fused model (Fig. 3). This results in the models working together to predict a single caption sequence.

2.1.6. Training

The training set of the ImageCLEFmedical 2022 Caption task consists of 83,275 radiology images where image has associated ground truth caption and concepts. The validation and test sets comprise of 7,645, and 7,601 radiology images, respectively [7]. All models were trained with a SGD optimizer, a sparse categorical entropy loss, a learning rate set at 0.001 and batch size of 32. The encoder block of the transformer model was trained with a 512 embedding dense projection dimension and a feed forward dimension of 512. The encoder was also trained on 256 attention heads. The vision transformer_1 (ViT1) encoder was also trained with a 512 embedding dense projection set, a patch encoding dimension of 512 and 256 multi-attention heads. The ***vision transformer_2 (ViT2)*** had similar configurations as the vision transformer with a patch creator replacing the patch encoder. The decoder was trained on a vocabulary size of 20,000 while the maximum sequence length for the model was 30. All the text sequences exceeding a length of 30 were cut off at the 30th word due to the fact that most of the words relevant to the image appeared in the first 30 words of the caption sequence. The decoder was trained with 256 attention heads while the word embedding dimension was set as 512. All the models were trained for about 50 epochs and early stopping was applied to prevent overfitting.

2.2. Multi-label Concept Classification

The concept detection task also considered as a multi-label classification problem which involves predicting zero or more concept labels (multiple mutually non-exclusive classes) to each image instance. We focused on flat (non-hierarchical) multi-label classification methods using three different CNN models: modified version of popular VGGNet [14] and GoogLeNet [15] and original AlexNet [16] architecture with following configuration:

- Number of nodes in the output layer matches the number of concept labels in the training set images.
- We divided the detection task into a series of multiple binary classification problems by keeping activation function of the classification(output) layer in our model to "sigmoid".

The first CNN architecture used is a simplified version of the original VGGNet architecture [14], which consists of two sets of CONV => RELU => CONV => RELU => POOL layers, followed by a set of FC => RELU => FC => SOFTMAX layers. The first two CONV layers learn 32 filters, each of size 3 X 3. The second two CONV layers learn 64 filters, again, each of size 3 X 3. The POOL layers perform max pooling over a 2 X 2 window with a 1 X 1 stride. We also inserted batch normalization layers after the activations along with dropout layers (DO) after the POOL and FC layers [12].

Also, a smaller version of original GoogLeNet [15] architecture is used using the Miniception module which consists of building blocks including a convolution module, Inception module, and Downsample module [17]. These modules are put together to form the overall architecture. The Mininception module performed two sets of convolutions –a 1 X 1 CONV and a 3 X 3 CONV. These two convolutions are performed in parallel, and the resulting features concatenated across the channel dimension. Next comes the downsample module, which is responsible for reducing the spatial dimensions of an input volume. The first Inception module learns 32 filters for both the 1 X 1 and 3 X 3 CONV layers. When concatenated, this module outputs a volume with K = 32 + 32 = 64 filters. The second Inception module learns 32, 1 X 1 filters and 48, 3 X 3 filters with output volume size is K = 32 + 48 = 80 after concatenation. The down sample module reduces our input volume sizes but keeps the same number of filters learned at 80. The four Inception modules are stacked on top of each other before applying a down sample, allowing GoogLeNet to learn deeper, richer features.

The original AlexNet [16] model was also used for experimentation where the first block applies 96, 11 X 11 kernels with a stride of 4 X 4, followed by a RELU activation and max pooling with a pool size of 3 X 3 and strides of 2 X 2, resulting in an output volume of size 55 X 55. A second CONV => RELU => POOL layer is then applied using 256, 5 X 5 filters with 1 X 1 strides. After applying max pooling again with a pool size of 3 X 3 and strides of 2 X 2 we are left with a 13 X 13 volume. Next, we apply (CONV => RELU) * 3 => POOL. The first two CONV layers learn 384, 3 X 3 filters while the final CONV learns 256, 3 X 3 filters. After another max pooling operation, we reach our two FC layers, each with 4096 nodes and ReLU activations in between. The final layer in the network is our "sigmoid" classifier for multi-label classification.

2.2.1 Concept Model Training

The Concept Unique Identifiers (CUI) as concept labels of each training images are extracted from a csv file and added in a list. More than 8K concept labels are found in the training set that could be predicted, and more than one can be true: hence the multi-label nature of the problem. In our implementation, all images are resized to 96 X 96 for space and computational efficiency and scaled the raw pixel intensities to the range [0, 1] and stored as NumPy arrays. A data list contains images stored as NumPy arrays and converted concept labels list (as a list of lists) to a NumPy array as well. After that, labels are binarized for multi-class classification by utilizing the scikit-learn library's MultiLabelBinarizer class, which actually transforms the CUI labels into a vector that encodes which concepts are present in the image [12]. The high volume of classification (CUI) labels (>8K) and

imbalance in the label frequency results in a huge bias towards the multi-label classification problem. Hence, data augmentation (scaling, rotation, flipping, etc.) is also applied while training as we have only a handful of images per concept class. All the above models are built by initializing the Adam optimizer [13] and compiled using ***binary cross-entropy*** rather than categorical cross-entropy to treat each output label as an independent Bernoulli distribution where the labels are not disjoint. After training is complete, the models and label binarizers are saved to disk (cloud storage) and loaded later during prediction in the test set.

High memory and computing systems with four NVIDIA® T4 GPU drivers were procured from the cloud and trained in parallel using the TensorFlow 2.8 mirrored strategy. Training was also performed on the VertexAI workbench of the Google Cloud platform.

3. Run Descriptions and Results

We have submitted in total twelve (12) runs in this year's participation of the ImageCLEFmedical 2022 Caption task, where 8 runs were submitted for the Concept Detection task and the remaining 4 runs were submitted for the caption prediction task as shown in Table 1 and 2. For transformer-based approaches, the run descriptions are very similar for both Concept Detection and Caption Prediction. For Concept Detection, first four runs in Table 1 are based on the multi-label classification approach as described in Section 2.2 and the other four runs are transformer based as described in Section 2.1. In Table 1, a Secondary F1 score is also included in addition to the F1 score, which was calculated using a subset of manually validated concepts (anatomy and image modality) only. In Table 2, additional metrics, such as ROUGE, METEOR, CIDEr, and BERTScore are included in addition to the BLEU score used for ranking.

As it is observed in Table 1 and 2, both the best accuracies for Concept Detection (F1 score: 0.3519) and Caption Prediction (BLEU Score: 0.2549) are achieved while fusing the outputs of both Text and Visual Transformers. In addition, we achieved a comparable accuracy (F1 score: 0.3165) for Concept Detection using multi-label classification scheme while using a smaller version of original VGGNet architecture and considering all the concept output probabilities with a threshold of 2% only. The best scores are in boldface for each metric in Table 1 and Table 2. Overall, we ranked 9th out of 11 groups for the Concept Detection task and ranked 8th out of 10 groups for the Caption Prediction task.

3.1. Description of the Runs

The submitted runs for both the Concept Detection and Caption Prediction tasks are described as follows:

1. minigooglenet_prob2_100: This run utilized a multi-label classification model by using scikit-learn MultiLabelBinarizer class with a smaller version of the original GoogLeNet architecture using the Miniception module as described in Section 2.2. For training the following parameters are used: EPOCHS = 25, initial learning rate (INIT_LR) = 1e-3, batch size (BS) = 32 and IMAGE_DIMS = (96, 96, 3). For this run, we only considered concept labels with a probability threshold of 2% for each test image.

2. alexnet_prob2_100: This run utilized a multi-label classification model by using scikit-learn MultiLabelBinarizer class with the original version of the AlexNet architecture. We considered the following hyper parameters for training: EPOCHS = 100, INIT_LR = 1e-3, BS = 32, IMAGE_DIMS = (96, 96, 3), For this run, we only consider concept labels with a probability threshold of 2% for each test image.

3. vggnet_top20: This run utilized a multi-label classification model by using scikit-learn MultiLabelBinarizer class with a mini version of the original VGGNet architecture as described in Section 2.2. To initialize the number of epochs to train for, initial learning rate, batch size, and image dimensions, the following hyper parameters are used: EPOCHS = 200, INIT_LR =

1e-3 (the default value for the Adam optimizer), BS = 32 and IMAGE_DIMS = (96, 96, 3). For this run, we only consider the top 20 concept labels based on probability scores for each test image.

4. vggnet_prob2_100: This run is almost similar to the previous run. Here, we only considered concept labels with a probability threshold of 2% for each test image.

5. vit_1_20000_concepts: This run involved the concept sequence generation using the model obtained from training the ViT1 architecture in the training dataset. The model was trained for 50 epochs with early stopping applied based on the validation loss obtained from the validation training dataset.

6. vit_2_20000_concepts: Utilizing the ViT2 architecture, this run involved training a model on the imageCLEF'2022 training dataset for 50 epochs. The model training was stopped early at the 44th epoch after a patience of 10 was applied over the validation loss obtained from the validation dataset. The model also learned a vocabulary of 8000 CUI and a maximum length of the sequence was 30 CUI. The text training input was cut off after 28 words and flags "start" and "end" were appended to the text sequence.

7. t4mr_20000_concepts: In this run, encoder was used to transform visual features obtained from a pretrained CheXNet model as described earlier. The architecture was trained for 50 epochs and was stopped early after the 30th epoch.

8. fusion_concepts: This run involved the fusion model that combines the ViT1, ViT2 and Transformer models by obtaining the probabilities of the next CUI predictions and choosing the concept with the highest probability to generate the concept sequence. This run utilized an image dimension equal to (224,224,3) and initial seed text "start" fed into the model runs for ViT1, ViT2 and Transformer architecture.

9. vit_1_20000_caption: This run involved the sequence generation of caption using the model obtained from training the ViT1 architecture in the training dataset. The model was trained for 50 epochs with early stopping applied at the 32nd epoch based on the validation loss obtained from the validation training dataset.

10. vit_2_20000_caption: Utilizing the ViT2 architecture, this run involved training a model on the imageCLEF'2022 training dataset for 50 epochs. The model training was stopped early at the 44th epoch after a patience of 10 was applied over the validation loss obtained from the validation dataset.

11. t4mr_20000_caption: This run involved training the Transformer architecture described earlier where visual features were obtained from a pretrained CheXNet model. The architecture was trained for 50 epochs and was stopped early after the 32nd epoch. In all these runs for Caption Prediction, the models learned a vocabulary of 20,000 words and a maximum length of the sequence was 30 words. The text training input was cut off after 28 words and flags "start" and "end" were appended to the text sequence.

12. fusion_caption: This run involved the fusion model that combines the ViT1, ViT2 and Transformer models by obtaining the probabilities of the next word predictions and choosing the word with the highest probability to generate the caption sequence.

Table 1

Results for the Runs submitted to the Concept Detection task

Submission Run	Run Name	F1 Score	Secondary F1
181984	minigooglenet_prob2_100	0.1757	0.1426
181959	alexnet_prob2_100	0.2323	0.3046
181952	vggnet_top20	0.1648	0.1214
181957	vggnet_prob2_100	0.3165	0.3757
182091	vit_1_20000_concepts	0.3340	0.6064
182111	vit_2_20000_concepts	0.3307	0.3307
181964	t4mr_20000_concepts	0.3257	0.5848
182150	fusion_concepts	**0.3519**	**0.6280**

Table 2
Results for the Runs submitted to the Caption Prediction task

Submission Run	Run Name	BLEU	ROUGE	METEOR	CIDEr	BERTScore
182092	vit_1_20000_caption	0.2501	0.1411	0.0549	0.1382	0.5814
182115	vit_2_20000_caption	0.2404	0.1338	0.0516	0.1299	0.5745
181962	t4mr_20000_caption	0.2459	**0.2459**	0.0489	0.1143	0.5721
182238	fusion_caption	**0.2549**	0.1440	**0.0559**	**0.1481**	**0.5834**

4. Conclusion

This article describes the strategies and results based on the participation of the Morgan_CS group in the ImageCLEFmedical 2022 Caption task. We submitted runs for both Concept Detection and Caption Prediction tasks under this benchmark evaluation. Our best results were achieved while using transformer-based approaches, especially when using fusion of different Transformers. Hence, in future we plan explore more domain specific Transformers based approaches with better fusion mechanism, which has already witnessed growing interest in medical field to capture global context compared to CNNs with local receptive fields.

Acknowledgements

This work is supported by an NSF Grant (Award ID. 2131207), entitled "CISE-MSI: DP: IIS: III: Deep Learning Based Automated Concept and Caption Generation of Medical Images Towards Developing an Effective Decision Support System (DSS)" under the CISE MSI Research Expansion program.

References

[1] H. Ayesha, S. Iqbal, M. Tariq, M. Abrar, M. Sanaullah, I. Abbas, A. Rehman, M. F. K. Niazi, S. Hussain, Automatic medical image interpretation: State of the art and future directions, Pattern Recognition, 114:107856 (2021). URL: https://doi.org/10.1016/j.patcog.2021.107856. doi: 10.1016/j.patcog.2021.107856.

[2] J. Pavlopoulos, V. Kougia, I. Androutsopoulos, A Survey on Biomedical Image Captioning, in: Proceedings of the Second Workshop on Shortcomings in Vision and Language, pp. 26–36. Association for Computational Linguistics, Minneapolis, Minnesota, 2019. URL: https://aclanthology.org/W19-1803. doi:10.18653/v1/W19-1803.

[3] M. M. A. Monshi, J. Poon, V. Chung, Deep learning in generating radiology reports: A survey. Artificial Intelligence in Medicine, 106:101878, (2020). URL: https://doi.org/10.1016/j.artmed.2020.101878. doi: 10.1016/j.artmed.2020.101878.

[4] J. Devlin, M.W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, NAACL-HLT (1) (2019). pp. 4171-4186. doi: 10.18653/v1/n19-1423

[5] F. Shamshad, S. Khan, S. W. Zamir, M. H. Khan, M. Hayat, F. S. Khan, H. Fu, Transformers in Medical Imaging: A Survey, arXiv:2201.09873, (2022). URL: https://doi.org/10.48550/arXiv.2201.09873.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need. in: Proceedings of Advances in neural information processing systems, pp. 5998–6008, 2017. arXiv:1706.03762. URL: https://doi.org/10.48550/arXiv.1706.03762.

[7] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, in: Proceedings of ICLR, (2021). arXiv:2010.11929v2 [cs.CV]. URL: https://doi.org/10.48550/arXiv.2010.11929

[8] J. Rückert, A. B. Abacha, A. G. S, de Herrera, L. Bloch, R. Brüngel, A. Idrissi-Yaghir, H. Schäfer, H. Müller and C. M. Friedrich, Overview of ImageCLEFmedical 2022 – Caption Prediction and Concept Detection, in Experimental IR Meets Multilinguality, Multimodality, and Interaction, in: Proceedings of CEUR Workshop (CEUR-WS.org), Bologna, Italy, September 5-8, 2022.

[9] B. Ionescu, H. Müller, R. Péteri, J. Rückert, A. B. Abacha, A. G. S. de Herrera, C. M. Friedrich,L. Bloch, R. Brüngel, A. Idrissi-Yaghir, H. Schäfer, S. Kozlovski, Y. D. Cid, V. Kovalev, L.D. Ștefan, M. G. Constantin, M. Dogariu, A. Popescu, J. Deshayes-Chossart, H. Schindler, J. Chamberlain, A. Campello, A. Clark, Overview of the ImageCLEF 2022: Multimedia Retrieval in Medical, social media and Nature Applications, in Experimental IR Meets Multilinguality, Multimodality, and Interaction. in: Proceedings of the 13th International Conference of the CLEF Association (CLEF 2022), Springer Lecture Notes in Computer Science LNCS, Bologna, Italy, September 5-8, 2022.

[10] O. Pelka, S. Koitka, J. Rückert, F. Nensa, C. M. Friedrich, Radiology Objects in COntext (ROCO): A Multimodal Image Dataset, Intravascular Imaging and Computer Assisted Stenting and Large-Scale Annotation of Biomedical Data and Expert Label Synthesis, , in: Proceedings of the 7th Joint International Workshop, CVII-STENT (2018) and Third International Workshop, LABELS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain. doi: 10.1007/978-3-030-01364-6_20.

[11] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Ding, A. Bagul, C. Langlotz, K. Shpanskaya, M. P. Lungren, A. Y. Ng, CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning. arXiv:1711.05225 [cs.CV], URL: https://doi.org/10.48550/arXiv.1711.05225

[12] Multi-label classification with Keras, URL: https://pyimagesearch.com/2018/05/07/multi-label-classification-with-keras/

[13] D. P. Kingma, L. J. Ba, Adam: A Method for Stochastic Optimization. CoRR abs/1412.6980 (2015): n. pag.

[14] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition. ICLR (2015). arXiv 1409.1556.

[15] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1-9, (2015). URL: https://doi.org/10.48550/arXiv.1409.4842

[16] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks. Communications of the ACM, 60:6 (2017), pp. 84-90. URL: https://doi.org/10.1145/3065386

[17] Z. Chiyuan, B. Samy, H. Moritz, R. Benjamin, V. Oriol, Understanding Deep Learning (Still) Requires Rethinking Generalization, Communications of the ACM, 64:3 (2021), pp. 107-115. URL: https://doi.org/10.1145/3446776