Classifying Chest X-Ray Images Using CNN and Transformer Based Architectures

Burak Bozdağ
Computer and Informatics Engineering
Istanbul Technical University
İstanbul, Türkiye
bozdagb17@itu.edu.tr

***Abstract*— In the article titled "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale", it was emphasized that transformers applied directly to image patches and pre-trained on large datasets work really well on image classification [1]. In this article, it is aimed to examine that if classifying chest X-ray images as normal or infected using transformer-based architectures gives better results than CNN architectures. This have been achieved by using AlexNet, one of popular CNN architectures, and ViT (Vision Transformer) stated in the related article. Aside comparing models, fine-tuning models and hyperparameter optimization have also been performed.**

Keywords—transformer, convolution, deep-learning, x-ray, pneumonia, classifier, architecture, comparison

I. INTRODUCTION

As the influence of the artificial intelligence (AI) and machine learning (ML) sector increased, many researches about this area have been showed up and also there are many ongoing projects and researches ahead. By realizing that the computer can learn from data, new algorithms of machine learning topic are found and developed. The data itself is not limited to just a bunch of numbers but also covers a whole area of computer vision (CV) world. In this world, the data is made of images that can be either classified (supervised) or unclassified (unsupervised).

Analysis of the medical data has been a huge topic itself. Examining and analyzing X-ray images has an importance in the medical domain. There are various work areas such as diagnosing radiology results with respect to the X-ray images of patients. There are so many projects out there for diagnosing and classifying different types of diseases in the literature.

Inspired by the popularity of the medical domain, a new topic is proposed for comparing standard convolutional neural networks and vision transformers in this article. Fine-tuning, hyperparameter optimization, learning rate selection and regularization methods are also applied when setting up models.

In the rest of the article, there is given more detailed project explanations with the goals of project, impact of solution, state-of-the-art research articles and novel contributions. In the next section, scope of the project is included with detailed information about the project. At the end of the article, references are given.

II. PROJECT DESCRIPTION

In this project, chest X-ray images have been classified for the pneumonia disease as normal or infected. Aside classifying data, a comparison between different CNN and transformer models have been made in order to determine the best model for this scenario.

A. Goals of Project

There are 2 main purposes of this project.

- Classifying X-ray images for diseases
- Comparing CNN and transformer models

B. Impact of Solution

This article shows difference between CNN and transformer models in the domain of X-ray images. At the later parts of the article, results are given in a comparative manner.

C. State-of-the-Art

AlexNet [2] which published in 2012 is a state-of-the-art work that increased the popularity of convolutional neural network and deep learning models. This model's results are very good compared to existing machine learning and computer vision algorithms.

AlexNet includes 8 layers: 5 convolution and 3 fully connected layers. Comparing to the other algorithms, this model has more filter and convolution layers at each layer. There are various used functions such as max-pooling, dropout, augmentation, ReLU, SGD, etc. Rectified linear unit (ReLU) functions are placed after convolution layers.

The AlexNet structure is given below:

Identify applicable funding agency here. If none, delete this text box.

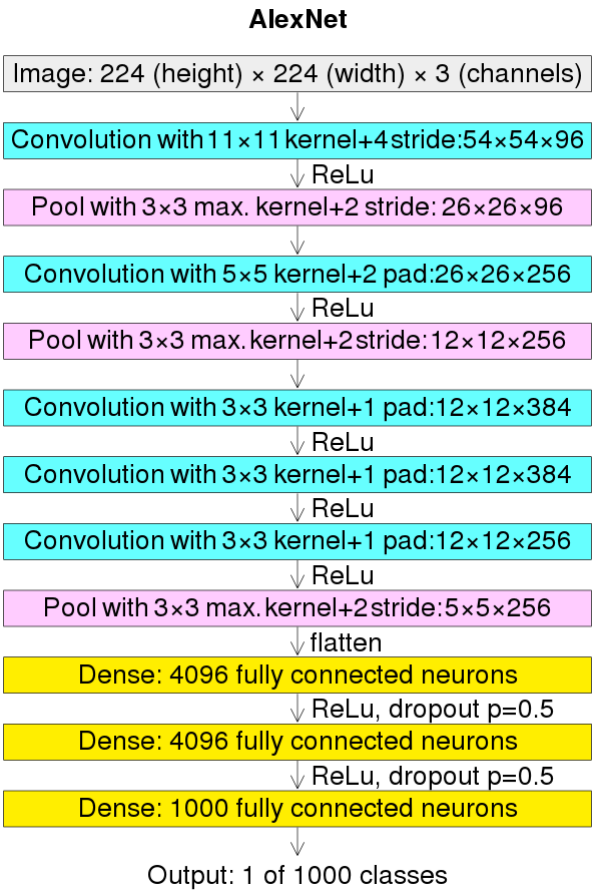

Fig. 1. AlexNet structure.

Even though transformer architectures are a standard for natural language processing (NLP) projects, its applicability to computer vision projects were limited. In computer vision, convolutional neural networks seem like the best way to solve classification problems.

In the related study [1], it is stated that the dependence to the CNN models is not necessary. Transformer structures can also be adapted to classification tasks and will give very good results. For example, some popular public datasets such as ImageNet, CIFAR-100, VTAB, etc. trained with vision transformers have given better results compared to the state-of-the-art CNN models.

The vision transformer structure is given below:

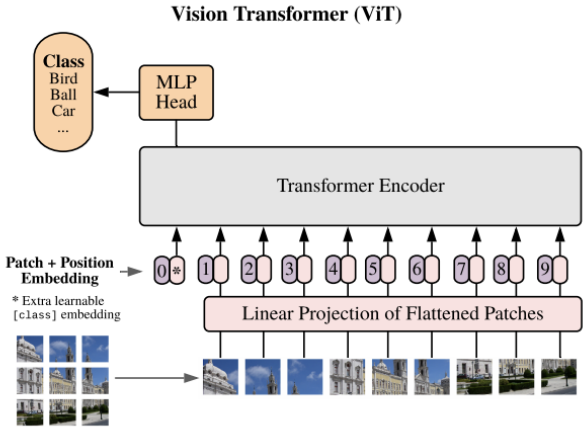

Fig. 2. Vision transformer (ViT).

Similar works about my study area include but not limited to:

- Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning: Demonstrating the general applicability of their AI system for diagnosis of pneumonia using chest X-ray images [5].
- Reliable Tuberculosis Detection using Chest X-ray with Deep Learning, Segmentation and Visualization: They have detected tuberculosis reliably from the chest X-ray images using image pre-processing, data augmentation, image segmentation, and deep-learning classification techniques [6].
- Exploring the Effect of Image Enhancement Techniques on COVID-19 Detection using Chest X-ray Images: A novel U-Net model was proposed and compared with the standard U-Net model for lung segmentation [7].

D. Novel Contributions

This article will lead us that if CNN models are enough and sufficient for detecting diseases from X-ray images or if transformer models can be also used for this purpose. This is achieved by comparing two models depending on different parameters.

III. Applied Processes and Methods

In this part, the information is given about the dataset, used technologies, applied processes and methods.

A. Dataset

The dataset used in this project is publicly available on Kaggle website named Chest X-Ray Images (Pneumonia) [4] from Paul Mooney.

Pneumonia dataset contains 5863 x-ray images in JPEG format which are categorized as pneumonia or normal.

The splitting of the dataset into training, validation and testing is done as shown in the table below:

TABLE I. DATASET SPLITTING

Pneumonia Dataset	Groups		
	Train	***Validation***	***Test***
5863	5216	16	624

B. Data Augmentation

In order to reduce overfitting and get a more generalized result after training, data augmentation methods can be applied at the pre-processing stage of machine learning tasks.

Data augmentation methods include but not limited to:

- Rescaling
- Zooming
- Rotation
- Horizontal-Vertical Flip

For this project, these data augmentation methods are applied in pre-training phase to the pneumonia dataset. These methods were applied only to the training set but not the test set because we do not want to alter the testing data for more realistic results.

Augmentation methods are applied by randomizing zooming, rotation and horizontal-vertical flip parameters in a determined interval. Rescaling is used for getting the data ready for being the input of models, so it is applied to all sets of the data (training, validation and testing groups).

Used parameters and intervals are given in the table below:

TABLE II. DATA AUGMENTATION PARAMETERS

Parameters	**Value**
Rescaling	1/255
Zooming	0.1
Rotation	0.2
Horizontal Flip	Yes
Vertical Flip	Yes

Some examples of augmented data samples are given below (rotation and flip):

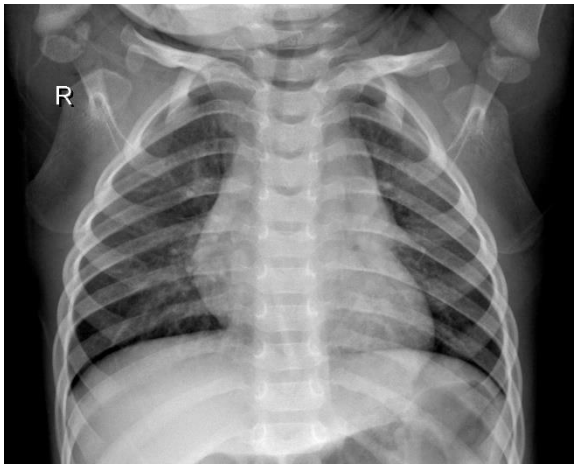

Fig. 3. Original chest x-ray sample.

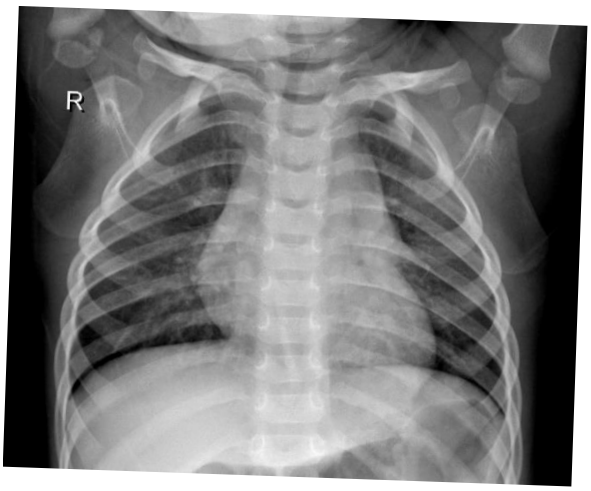

Fig. 4. Rotated chest x-ray sample.

Fig. 5. Flipped chest x-ray sample.

With these transformations, the model adapts to unexpected changes and the model gets more generalized for real-life scenarios.

C. Test Bench

The entire augmentation, training and testing process is done in a local computer with AMD Radeon RX 6600 XT GPU. TensorFlow and Keras API were used for implementation.

For GPU acceleration, DirectML plugin is used in order to speed up the training process [8].

Two different training runs are made for both AlexNet and vision transformer models. The training time for AlexNet and ViT were approximately 20 minutes and 1 hour respectively. The discussion about the training cost will be made in latter parts of the article.

D. Callbacks

During training; fine-tuning, hyperparameter optimization, monitoring progress and saving the best model can be made with the help of the Keras API. Callback methods provide these functionalities and makes the training stage more efficient.

In this project, used callback methods are as follows:

- Reducing learning rate: On each epoch, validation loss is being watched for improvements. If there is no more improvement in the validation loss value, the learning rate will be reduced for the next epochs of training.
- Early stopping: After learning rate reduction, this callback method gets active and starts to watch the validation loss value for each epoch. If the validation loss value does not improve anymore, the training will be stopped by this callback and the best model is selected.
- Model checkpoint: The best model is saved to a file for future use.

E. *Setting Up Models*

As a traditional solution of CNN (convolutional neural network), AlexNet architecture is selected and used in this comparison. AlexNet consists of a bunch of convolutional, max-pooling and dense layers sequentially.

For transformer model, the related ViT-B/16 (vision transformer) model is used. This model has been implemented in the Keras API as a library which is pre-trained with ImageNet 2012 dataset. ViT consists of a convolutional and reshape layer followed by 12 transformer encoders, normalization, lambda and dense layers.

As for model's arguments; optimizer and loss function are determined, and maximum number of epochs is limited to 50.

Adam is used as optimizer argument for both models. It is one of the popular optimizers that has been used in deep learning domain. Especially for the transformers, Adam optimizer is used more frequently because for the MLP (multi-layer perceptron) layer, this optimizer decreases the training cost much faster than other optimizers. In the following figure, the results for optimizers' training costs are given.

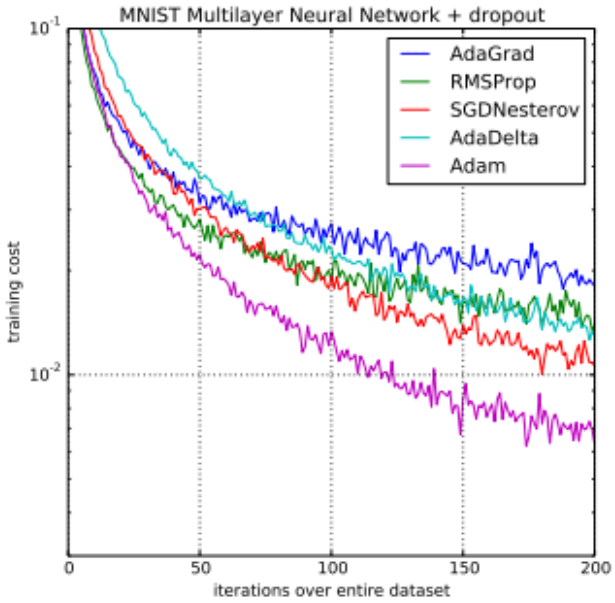

Fig. 6. *Training cost comparison of optimizers for MLP.*

As for loss function, binary cross-entropy is used because our domain consists of a single classification problem which includes two classes: normal or infected.

$$H_p(q) = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot log(p(y_i)) + (1 - y_i) \cdot log(1 - p(y_i))$$

Fig. 7. Binary cross-entropy / log loss.

F. *Evaluation and Results*

At evaluation stage, since the main goal is to compare two different architectures, models' optimizer and loss function remained same.

The initial value of the learning rate is set to 10^{-4} and decay is set to 10^{-6} for the Adam optimizer. Learning rate reduces to $2.5 * 10^{-5}$ when validation loss stops improving. Binary cross-entropy loss function is used for both models. The main metric to consider is determined as accuracy.

For the AlexNet model (CNN), accuracy and loss values with respect to the iterations are shown in graphs below.

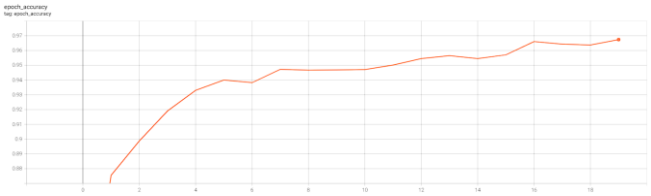

Fig. 8. AlexNet accuracy-epoch graph.

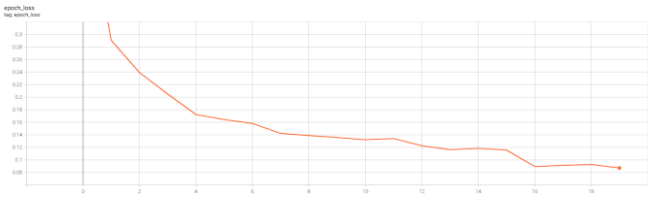

Fig. 9. AlexNet loss-epoch graph.

For the vision transformer model (ViT), accuracy and loss values with respect to the iterations are shown in graphs below.

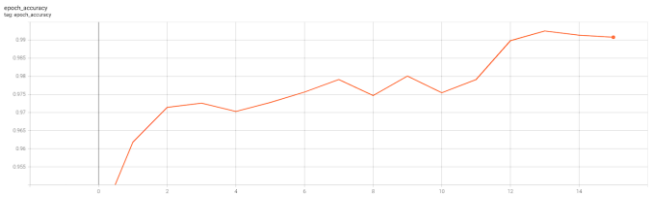

Fig. 10. Vision transformer accuracy-epoch graph.

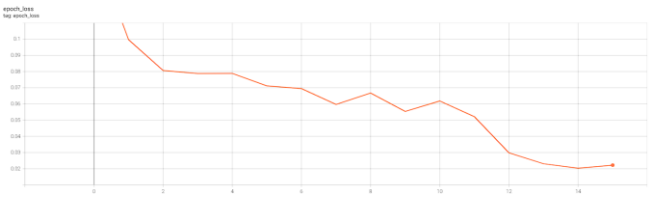

Fig. 11. Vision transformer loss-epoch graph.

The classification report of the AlexNet model is shown in the table below.

TABLE III. ALEXNET CLASSIFICATION REPORT

Labels	Precision	Recall	F1-Score
0 (normal)	0.92	0.71	0.80
1 (pneumonia)	0.85	0.96	0.90
Accuracy			0.87
Macro Avg.	0.88	0.84	0.85
Weighted Avg.	0.87	0.87	0.86

The classification report of the ViT model is shown in the table below.

TABLE IV. VISION TRANSFORMER CLASSIFICATION REPORT

Labels	Precision	Recall	F1-Score
0 (normal)	0.98	0.82	0.90
1 (pneumonia)	0.90	0.99	0.95
Accuracy			0.93
Macro Avg.	0.94	0.91	0.92
Weighted Avg.	0.93	0.93	0.93

IV. CONCLUSION

In this article, the main goal is to compare traditional CNN models to relatively new algorithm vision transformer. The comparison is made in the domain of chest x-ray binary classification problem.

For the CNN model, AlexNet is a solid preference because of its popularity in computer vision area. But for transformer models, they were been using in natural language processing tasks actively until the new research about vision transformers.

Comparing AlexNet and ViT, we can conclude that vision transformer performs better than AlexNet with the expense of the training cost. In the same test bench, the training of AlexNet took 20 minutes while the training of vision transformer took nearly 1 hour which is a big margin between each other.

REFERENCES

[1] A. Dosovitsky et al., "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale", 2021. Available: https://openreview.net/forum?id=YicbFdNTTy.

[2] Krizhevsky, Alex & Sutskever, Ilya & Hinton, Geoffrey. (2012). ImageNet Classification with Deep Convolutional Neural Networks. Neural Information Processing Systems. 25.10.1145/3065386.

[3] C. Szegedy et al., "Going Deeper with Convolutions", arXiv.org, 2014. [Online]. Available: https://arxiv.org/abs/1409.4842.

[4] "Chest X-Ray Images (Pneumonia)", Kaggle.com. [Online]. Available: https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia. [Accessed: 17- Nov- 2022].

[5] Kermany, Daniel S et al. "Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning." Cell vol. 172,5 (2018): 1122-1131.e9. doi:10.1016/j.cell.2018.02.010

[6] T. Rahman et al., "Reliable Tuberculosis Detection Using Chest X-Ray With Deep Learning, Segmentation and Visualization," in IEEE Access, vol. 8, pp. 191586-191601, 2020, doi: 10.1109/ACCESS.2020.3031384.

[7] Rahman, Tawsifur et al. "Exploring the effect of image enhancement techniques on COVID-19 detection using chest X-ray images." Computers in biology and medicine vol. 132 (2021): 104319. doi:10.1016/j.compbiomed.2021.104319.

[8] "Enable GPU Acceleration for TensorFlow 2 with tensorflow-directml-plugin," DirectML Plugin for TensorFlow 2 | Microsoft Learn, 2022. [Online]. Available: https://learn.microsoft.com/en-us/windows/ai/directml/gpu-tensorflow-plugin.