Quantitative comparison between Vision Transformers and Deep Convolutional Neural Networks for classification of femoral fractures (Atypical/Normal)

Chayan Shrang Raj

Linköping University, SE-58183, Linköping, Sweden

Abstract: Atypical femoral fractures are claimed to be an uncommon complication of long-term exposure of some drugs such as bisphosphonates. It is the elusive nature of these kinds of fracture that makes it hard to classify between normal and atypical femoral fractures. In this paper, we will take the help of some deep learning algorithms for image classification such as more conventional Convolutional Neural Networks and some recent Vision Transformer architectures to be able to better classify the two types of fractures in the thigh bone, called atypical femoral fractures (AFF) and normal femoral fractures (NFF). The X-Ray image dataset originates from 72 Swedish Hospitals with different imaging equipment. It is important to note that this dataset is highly imbalanced and required some preprocessing like resizing, rescaling, normalization, and data augmentation before being fed into the network architectures. Furthermore, we compared the two different computer vision algorithms based on some metric, sensitivity is a prioritized metric because of imbalanced nature of our image dataset.

Keywords: Atypical, Convolution, Vision, Neural Network, Transformers, image, fracture, femoral.

1 Introduction

There are good cues that artificial intelligence (AI) will transform medical diagnosis and offer improved healthcare at a reduced cost. Radiology and pathology are purported to be the first fields in medicine where AI shall expand its implementation horizon. It is interesting to understand that there are so many different medical setups in the world, and this creates hindrance to AI systems in healthcare which by purpose should predict in varied scenarios and with high confidence [1]. Radiography requires expert knowledge in interpreting radiographs of human anatomy for example, to diagnose fractures. Atypical femoral fracture is a serious but rare condition of femoral fractures resulting from long-term bisphosphonate therapy [2]. This gradual nature of this kind

of fracture makes it hard to discern between a normal and an atypical fracture and also makes the diagnostic accuracy of radiographs with conventional radiology reports a dire affair. In this paper, we will try to use and compare some of the deep convolutional neural networks with new state-of-the-art Vision transformers for binary image classification between normal femoral fracture and atypical femoral fracture.

2 Background and purpose

Bisphosphonate therapy has proven to be a crucial treatment for patients with osteoporosis and works by minimizing bone loss and reduces fracture risk by up to 50%. Consequently, it is also associated with amplifying risks of osteonecrosis (death of bone tissue because of less blood supply). To prevent irrecoverable damage to the bone, we need to discern these fractures in a pre-empt manner. Normally, diagnostic X-Rays are done to detect the fractures with the help of expert radiology doctors but in our case, atypical fractures are not easily discernable to the naked eye and hence amounts to more than 10% human error [3]. The amount of data we have is not enough to train our own CNN model from scratch as the data is highly imbalanced but also because the images are from different equipment, their orientation is also highly varied. It is for this purpose we are leveraging pre trained supervised deep learning algorithms for binary image classification (normal/atypical fracture) in a transfer learning style manner which are then fine-tuned on our dataset that contains around 4335 images from some 1200 patients.

3 Patients and methods

Dataset

For this project, we have 4335 femoral fracture radiographs (868 Atypical femoral fractures and 3467 normal femoral fractures) from 1215 patients that were used as the total dataset in this study. Since each patient has more than one radiographs for fractures, we have included all of them without any omission to balance the distribution of patient images in train/valid/test sets. A challenge is that images have different size and resolution as they originate from 72 Swedish hospitals with different imaging equipment. The scheme for splitting dataset has been chosen on a patient level where each patient has all the images in only one of train/valid/test split to decrease data leakage while training the network. We train our model on training data, validate the model using validation set and finally check the generalization accuracy on unseen test data.

Preprocessing Procedures

The dataset consists of thousands of images with varied dimensions ranging from 1000x1000 to 3000x3000 and grayscale color scheme, which is inconsistent with the image dimensions required to train convolutional neural networks. A vast majority of pre trained CNN models are trained on ImageNet dataset, hence grayscale images were converted to Red Green Blue (RGB) images with 3 channels (with identical duplication

for each channel) [4]. To make the images consistent with different neural network architectures, the images were all padded with black area to make them all square in dimensions and then downsampled to architecture specific sizes such as for ResNets, it was 224x224x3, for inceptionv3 network, it was 299x299x3. Original images were all 16 Bit grayscale images for which pixel values ranged from 0 to 60000, which is a lot of information for a network to understand and much of it goes forward as noise. To circumvent this, we converted all images to 8 Bit size and image intensity was normalized to have a mean of 0 and standard deviation of 1. Random augmentations were performed on training set to increases the robustness of the trained model such as rotations (20 degrees), brightness range (0.90 – 1.25), horizontal flip, vertical flip, width-shift range (0.1). Keep the class imbalance in mind, we have also used class weights for 'aff' and 'control' respectively while training the model.

Transfer Learning

Since the inception of well-defined machine learning models, it has opened an avenue for variety of industrial applications each with its own data methods. Transfer learning is a method where a machine learning model is reused for a downstream task (similar task but with some constraints). Many smaller tasks can get very high accuracy by using a pre-trained model on similar data and modifying the architecture for the small task [5]. Basically, pretrained models use features extracted from previous training on images and use that knowledge to understand objects off different and unique datasets, which in our case, are the radiographs of patients. This saves vast compute and time resources required to develop neural network models on these problems. We have only used supervised pre-trained deep neural networks trained primarily on ImageNet (currently the largest publicly available dataset for object recognition) and then we fine-tuned the model for our dataset.

Evaluation Methods

We followed a standard procedure for training, evaluating, and testing our model. We used train set consisting of around 60% of the whole dataset, keeping unique and all the radiographs of patients to prevent data leakage into testing data. Next, we used 20% of the remaining data for validation set and the rest of the data for testing data. Hence, we evaluated our model using validation data. We also calculated the diagnostic accuracy, sensitivity, precision, recall and specificity of each network which we ultimately used to present the area under the ROC (receiver operating characteristics) curves (ROCAUC).

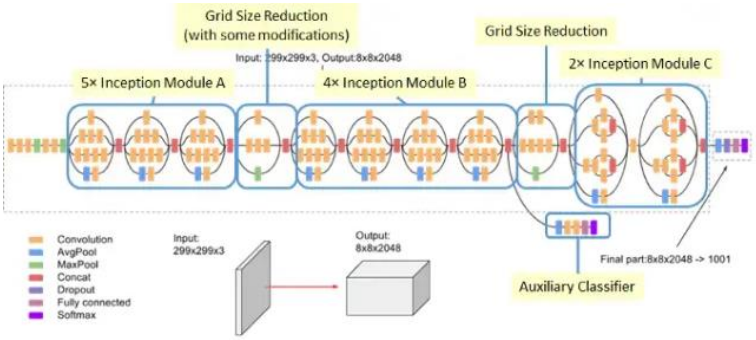
4 Network Architectures

Deep learning is a subset of broad machine-learning field which itself is a subset of an even broader Artificial Intelligence field [6]. For image classification task, we have been mostly using Convolutional Neural Networks, but since the development of state-

of-the-art Vision Transformers (ViT), we can see a shift in the usage and development of deep learning architectures for image analysis. In this paper, we will try to compare 4 conventional Deep Convolutional Networks with 4 different ViT architectures.

Inception-V3 Network

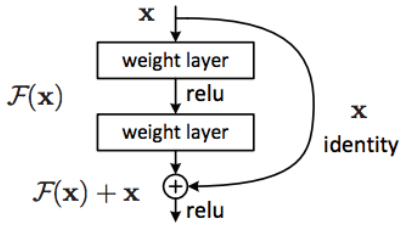
Inception-V3 is a convolutional neural network architecture from the inception family first introduced by google [7]. Version 3 makes several improvements including using label smoothing, factorized by even larger 7x7 convolutions which reduces the number of connections/parameters without decreasing the network efficiency, and taking advantage of an auxiliary classifier to propagate label information lower down the network (along with the use of batch normalization for layers in the sidehead) [8]. It has 42 layers but with computational cost of around 2.5 higher than that of GoogLeNet but much more efficient than VGGNet.



Source: Inception-v3 Architecture

ResNet-101

Training of 'plain' deep convolutional neural networks could be a difficult task because of exploding/vanishing gradient among many other issues. Residual networks solve these problems and allow for training of substantially much deeper networks for better performance [9]. An introduction of identity block allows the network to carry forward the residual information and skip the current weight layer. The identity mapping is multiplied by a linear projection to expand the channels of shortcut to match the residual.

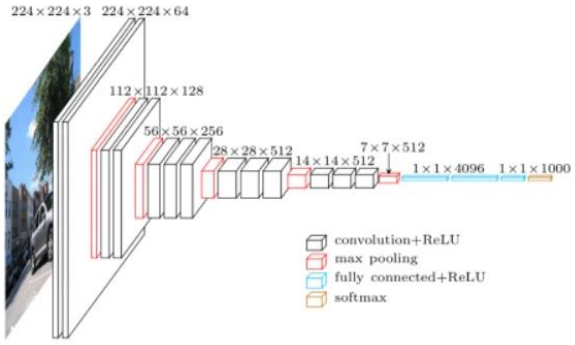


Source: Residual block

There are many variations of ResNets such as ResNet-18, ResNet-34, ResNet-50, ResNet-101, ResNet-110, ResNet-152, ResNet-164, ResNet-1202, etc. In our case, we have used ResNet 101 for binary image classification as it gave the highest accuracy among ResNet-18, ResNet-34, ResNet-50, ResNet-101, ResNet-110.

VGGNet-19

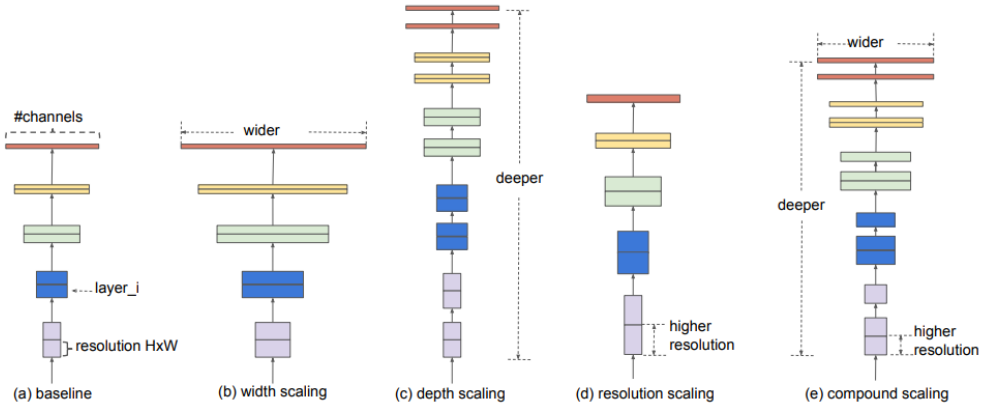
VGG (Visual Geometry Group) can be understood as a successor of the AlexNet but was created by a different group at Oxford [10]. We have used this architecture because it is one of the simplest and effective deep CNN architecture, which makes it easy to understand and implement using transfer learning. It has a fixed set of parameters such as kernel size (3x3), stride of 1-pixel, max pooling of 2x2 for all its convolution layers and fixed size of fully connected neural layers. This model is computationally expensive but was ranked 2nd place (after the inception network) for the image classification task and in 1st place for the localization task.



Source: VGG Neural Network Architecture

EfficientNet-B7 Network

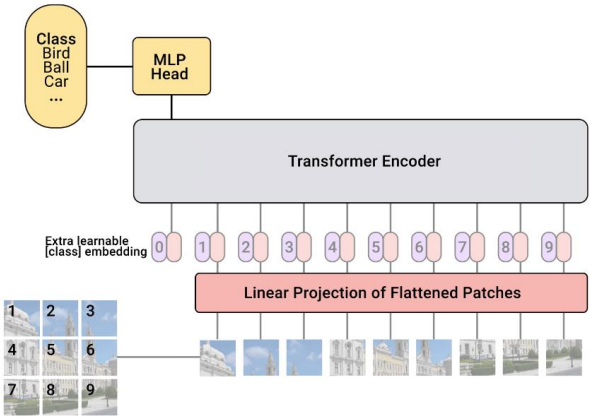
It has always been known that large scale convolutional neural networks are resource heavy because of a variety of hyperparameters involved in the training. Efficient Networks propose a new scaling method that uniformly scales all dimensions related to the network and input image such as depth/width/resolution using a trifling compounding coefficient [11]. Since, optimizing the layers and other different scaling options is primarily an experimental setup, there have been a lot of iterations of Efficient Networks on existing Resnets and MobileNets such as currently they have seven iterations of EfficientNet B0-B7. EfficientNet-B7 achieves state-of-the-art accuracy on ImageNet, while being 8.4x smaller and 6.1x faster on inference than the best existing ConvNets [12]. One of the main reasons for choosing this network was its ability to transfer well to different datasets.



Source: Model Scaling for EfficientNets

Vision Transformers (ViT)

ViTs are based on Transformers and attention mechanism [13] and currently the state-of-the-art neural networks for image classification. When pre-trained on a large dataset and transferred to different mid-sized or small image recognition benchmarks (ImageNet, CIFAR-100, VTAB, etc.), ViT outperforms state-of-the-art convolutional networks while requiring substantially fewer computational resources to train. Being, associated with attention mechanism, these models are incapable of addressing spatial images (permutation invariant) [14] and hence we always need to convert a spatial non-sequential signal to a sequence. Image patches are converted to sequence and are basically the sequence tokens (like words in NLP). Variations in architecture could be introduced by changing the number of encoder blocks (transformer blocks), number of multi-layer perceptron blocks at the end of the transformer block, by changing the embedding size (like NLP, the vector to store information about the sequence) or by changing the patch size.



Source: Vision Transformers model

For our use case, we have used 2 different settings for our model.

a) Patch size 16 without data augmentation, embedding size 768 (base model)
b) Patch size 16 with data augmentation, embedding size 768 (base model).

5 Results

When using standardized, pre-processed input data, the evaluated accuracies on the testing data are summarized in the table below:

Network	Accuracy	Precision	Recall	F1-Score	AUC
Inception-V3	71%	0.78	0.71	0.73	0.69
ResNet	55%	0.71	0.53	0.41	0.53
VGG19	71%	0.81	0.7	0.73	0.73
Efficient-Net B7	84%	0.85	0.84	0.85	0.86
ViT - 16 w/o Aug	88%	0.88	0.89	0.89	0.81
ViT - 16 w Aug	92%	0.92	0.92	0.91	0.84

6 Discussions

Our aim for this report was to compare different state-of-the-art CNN architectures with Vision Transformers (non-CNN architecture) and assess the effectiveness of ViT in classifying medical images. Though, it seems that the performance of ViT for this particular dataset has only been comparable to CNN architectures. We can see that ViT-16 with data augmentation achieved the highest accuracy among 6 models but EfficientNet-B7 achieved the maximum ROCAUC of 0.86. Future works may include fine tuning Vision Transformers for instance, using different embedding layer size, using class weights, or different patch size.

Github Link -

All the code has been provided in the below repository.

https://github.com/chayansraj/Computer-Vision-Research-Project

References

1. https://www.nature.com/articles/s41467-022-34945-8
2. https://pubs.rsna.org/doi/full/10.1148/ryai.210315
3. https://www.ccjm.org/content/ccjom/85/11/885.full.pdf
4. https://www.tandfonline.com/doi/full/10.1080/17453674.2018.1453714?src=recsys
5. https://machinelearningmastery.com/transfer-learning-for-deep-learning/
6. https://www.tandfonline.com/doi/full/10.1080/17453674.2019.1711323?src=recsys
7. https://arxiv.org/abs/1409.4842
8. https://arxiv.org/abs/1512.00567v3
9. https://arxiv.org/pdf/1512.03385
10. https://iq.opengenus.org/vgg19-architecture/
11. https://arxiv.org/pdf/1905.11946
12. https://paperswithcode.com/paper/efficientnet-rethinking-model-scaling-for
13. https://arxiv.org/abs/2010.11929
14. https://theaisummer.com/vision-transformer/