electronics

MDPI

Article

Deep Learning and Cloud-Based Computation for Cervical Spine Fracture Detection System

Paweł Chłąd * and Marek R. Ogiela *

Cryptography and Cognitive Informatics Laboratory, AGH University of Science and Technology, 30 Mickiewicza Ave., 30-059 Krakow, Poland

* Correspondence: pchlad5@gmail.com (P.C.); mogiela@agh.edu.pl (M.R.O.)

Abstract: Modern machine learning models, such as vision transformers (ViT), have been shown to outperform convolutional neural networks (CNNs) while using fewer computational resources. Although computed tomography (CT) is now the standard for imaging diagnosis of adult spine fractures, analyzing CT scans by hand is both time consuming and error prone. Deep learning (DL) techniques can offer more effective methods for detecting fractures, and with the increasing availability of ubiquitous cloud resources, implementing such systems worldwide is becoming more feasible. This study aims to evaluate the effectiveness of ViT for detecting cervical spine fractures. Data gathered during the research indicates that ViT models are suitable for large-scale automatic detection system implementation. The model achieved 98% accuracy and was easy to train while also being easily explainable.

Keywords: machine-learning; vision transformers; computer vision; medical; cloud; explainable AI

Citation: Chłąd, P.; Ogiela, M.R. Deep Learning and Cloud-Based Computation for Cervical Spine Fracture Detection System. *Electronics* **2023**, *12*, 2056. https://doi.org/10.3390/electronics12092056

Academic Editors: Francisco Luna-Perejón, Lourdes Miró Amarante and Francisco Gómez-Rodríguez

Received: 25 March 2023
Revised: 19 April 2023
Accepted: 24 April 2023
Published: 29 April 2023

1. Introduction

Every year, around 8 million spine fractures occur globally, with the cervical spine being one of the most commonly affected areas [1]. Spine damage often results in spinal cord injury, and there are approximately 250,000 to 500,000 such injuries annually, which can result in death or lifetime disability [2].

The incidence rate of spinal fractures is higher in the elderly population due to degenerative disease and osteoporosis [3]. The diagnosis of such fractures is typically conducted using 3D CT scans, and these require educated and specialized medical personnel, often making it difficult to obtain a timely diagnosis. Quick detection of fractures is crucial for preventing neurological degradation and disability after trauma [4].

With the rise of computer vision models, deep learning, and ubiquitous medical data, it has become feasible to implement systems that can augment the work of already overloaded medical personnel [5]. Quicker diagnosis could prevent lifelong disabilities and even death in some cases. The 3D space of CT scans is also difficult to navigate for human beings, as doctors must sift through hundreds of 2D slices of the scans, which might take substantial amount of time, whereas machine learning models can go through thousands of images in seconds [6].

Current cloud technologies allow for the development of tools for the medical industry that can automatically scale, improve existing models, and build medical datasets [7,8]. The use of cloud could also allow for the exploitation of much bigger models and would not burden hospitals and clinics with additional infrastructure dedicated to running machine learning models. Note that a careful balance must be struck between model size, available bandwidth, latency, and data availability.

The goal of such a tool is not to replace medical personnel, but instead to provide additional information and insights to professionals. The model must provide a reason behind the given result, and the vision transformer architecture is easily explainable due

to the notion of "attention", which can be used to create heatmaps that show areas of interest. By combining the heatmap with the CT scan, personnel will be able to quickly identify damage and confirm the model's result. In case of a mistake by the model, it can be corrected by experts themselves, by marking an image as "damaged".

This paper focuses on the evaluation of vision transformer architecture for damage detection in vertebral CT scans and proposes a cloud-based system for automatic inference and training of such models. The goal of this work was to achieve accuracy similar to state-of-the-art methods [9].

2. Materials and Methods

2.1. Vision Transformer

Vision transformer is one of the newest architectures in the field of computer vision and machine learning [10]. ViT works on the basis of a mechanism called "attention", which allows the model to focus on specific parts of the input sequence and ignore those that are less significant. This differs from older architectures, as attention is calculated per input rather than being statically encoded in the model itself. The computational complexity of attention calculation is quadratic ($O(n^2)$, as we need to compute attention for each pair). For most computer vision applications, the base unit of analysis is a pixel, but since ViT scales quadratically, a simple 1024×1024 image will result in over a trillion computations. This prohibitive scaling led to the use of an analysis space reduction technique which splits the image into equally sized patches that are later used as an input to the architecture.

Patches are $W \times H \times C$ tensor, which is later flattened and turned into vectors $x_1, \ldots, x_n$ of $W \cdot H \cdot C$ size, which then, through the use of a learnable dense layer, are encoded into a representation fit for the transformer-encoder layer. Every encoding dense layer has the same fixed dimensionality D, which is given as a hyperparameter of the model. The output of this projection is called patch embedding, which will be denoted as a sequence of vectors $p_1, \ldots, p_n$, of D dimension. Often, a class token is added and encoded as an additional sequence vector in order to mark a class of passed image to aid in training, but not always. After patch vectors are projected onto latent space dimensionality, positional encoding vectors $e_1, \ldots, e_n$ are added to each of $p_1, \ldots, p_n$; this is done in order to preserve the positional information from each of the patches after flattening and projection. The positional encoding varies from architecture to architecture; for example, a simple ViT architecture (which is used later as a baseline for this paper) uses 2d sinusoidal positional embedding.

ViT is based on transformer architecture shown in Figure 1 and is typically used for natural language processing tasks. The transformer typically consists of two parts: encoder and decoder (Figure 2). The encoder's task is to map the input sequence to a sequence of real number representations [11]. The decoder's task is to use continuous sequences to create an output sequence. Note that the tokens generated by decoders are later used as an input for the decoder to use to generate later tokens; this is done in order to achieve auto-regressive characteristics. The difference between the ViT and the original transformer architecture is that the ViT does not have a decoder layer; instead, the output of the encoder block is routed to the MLP head.

One encoder block consists of 4 parts, multi-head attention, add and norm, a feed-forward layer, and another add and norm layer. All layers have the same dimensionality D, and the output of each layer is then equal to $LayerNorm(x + Sublayer(x))$, where the sublayer is either multi-head attention (MHA) or a feed-forward layer.

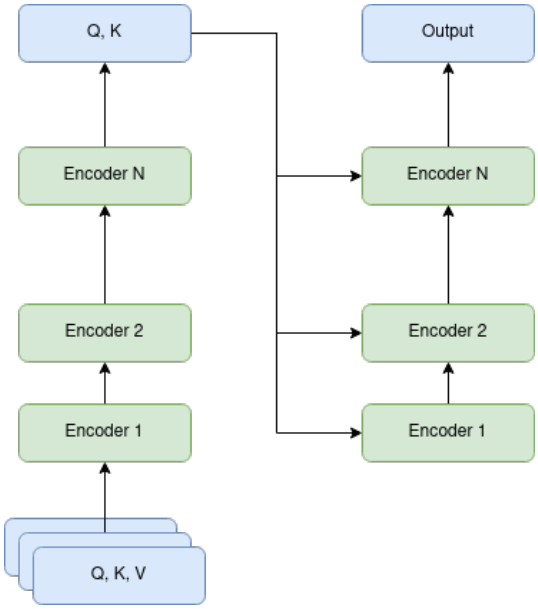

Figure 1. Transformer general architecture.

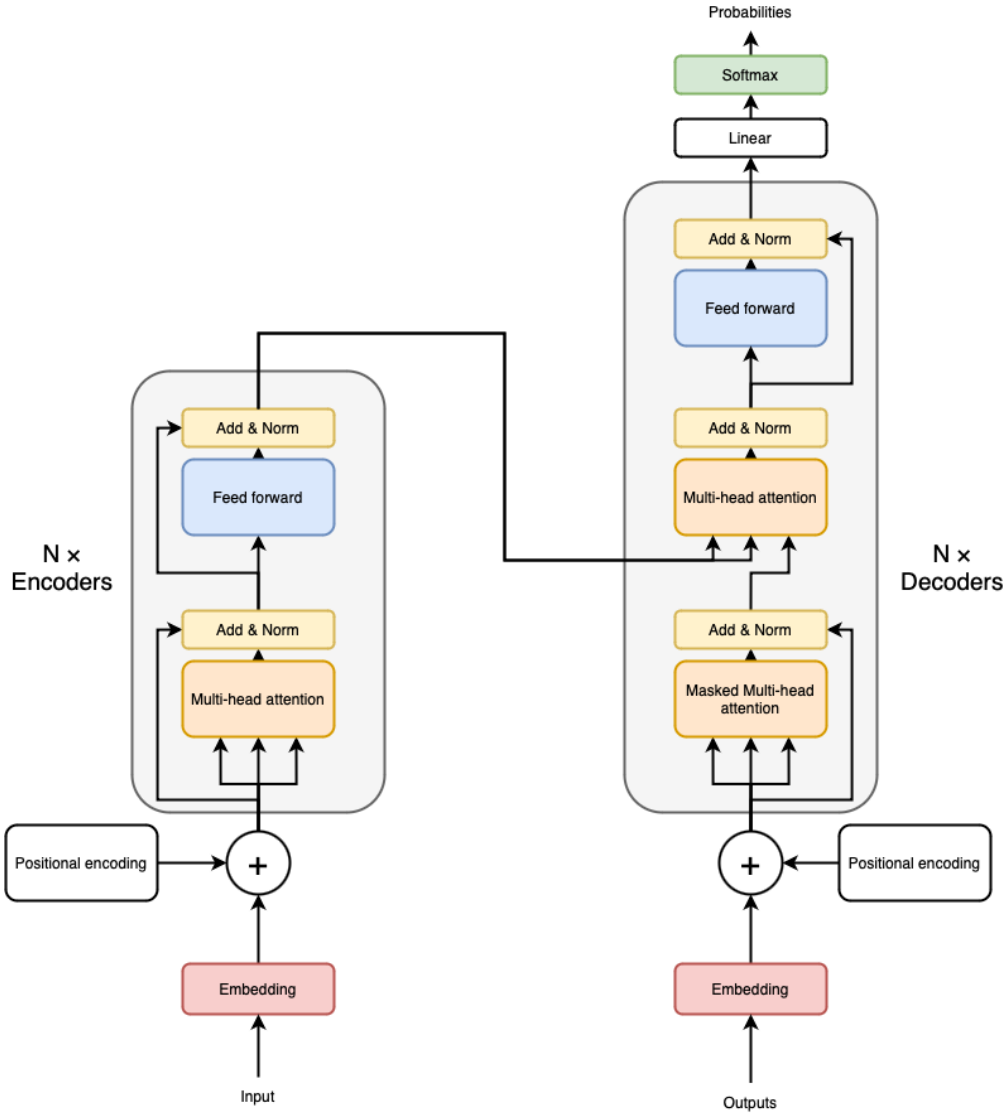

Figure 2. Transformer encoder, decoder internals.

Multi-head attention (Figure 3) is responsible for calculating the scaled dot product attention function (Equation (1)), which is a measure of the importance of a given part of an input sequence.

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (1)$$

The Q, K, V inputs vectors, in the case of ViT, are the same vector and are encoded by a learnable linear transformation layer. First the dot product attention is calculated, then it is divided by a scaling factor. This is done to produce bigger gradients in the softmax function, as, for large values of D, the dot products are large in magnitude.

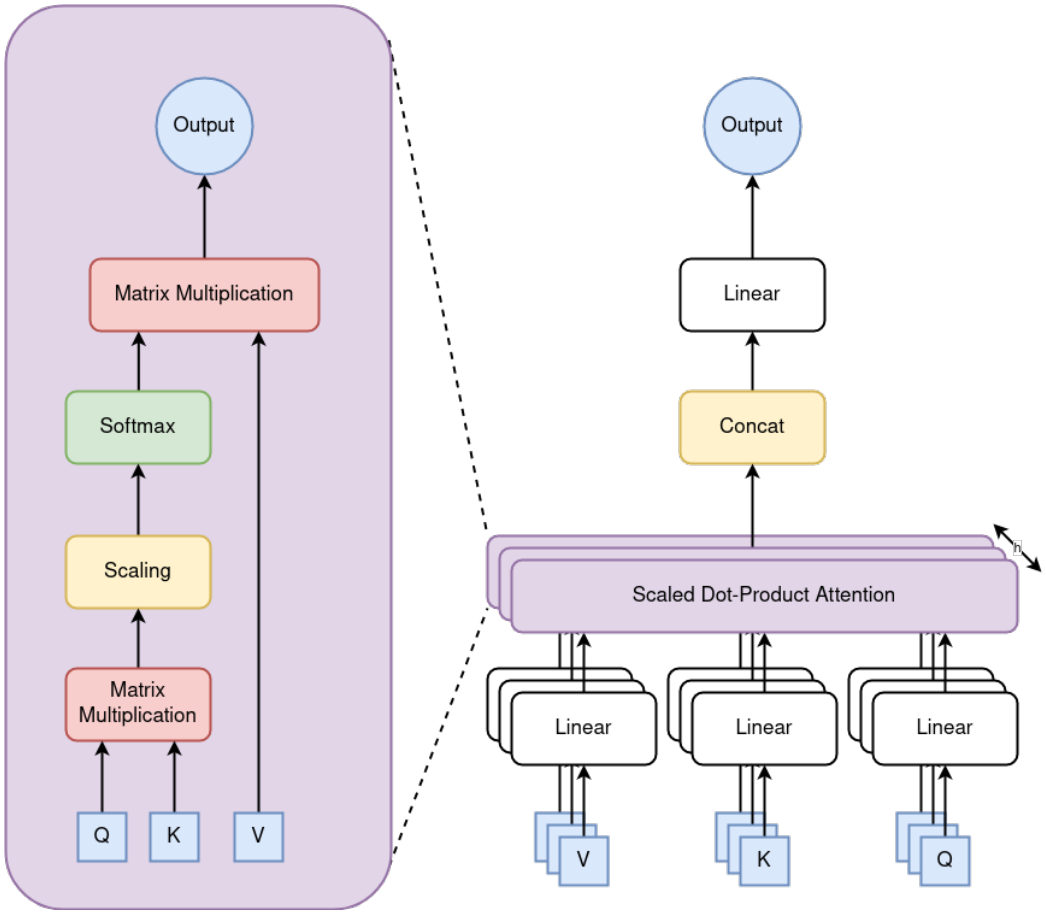

Figure 3. Multi-head attention internal workings.

The "multi-head" attention signifies that h of such layers are used in parallel to each other. Each such layer has a different linear transformation and thus represents a different representation subspace, which allows the model to focus on different parts of the input sequence or, in the case of ViT, different parts of an image. All of this helps the model to generalize better. The outputs of each layer are then later concatenated and transformed linearly in order to preserve the earlier given dimensionality D. The mechanism is similar to that of combining different convolutional filters in CNNs.

After passing through the MHA layer, the input is added to the output, normalized, and passed to the feed-forward layer in order to project the input sequence to latent space.

The parallel nature of both transformer and vision transformer architecture enables efficient evaluation on hardware accelerators, which previously was not really possible with recurrent models. The use of dynamically calculated attention and the lack of convolutions results in lesser inductive bias in comparison to CNNs, all the spatial relationships must be learned in the process of training—the only "local" layers are MLPs—while attention works globally, which allows the model to find non-trivial, non-local relationships.

Putting all the details together, the final Vision Transformer architecture emerges, as can be seen in Figure 4. An input image is divided into patches, which are later flattened. The resulting vectors are projected into latent space, and positional encoding is added. Then, the output goes into Encoder blocks, and the MLP head produces the final prediction result.

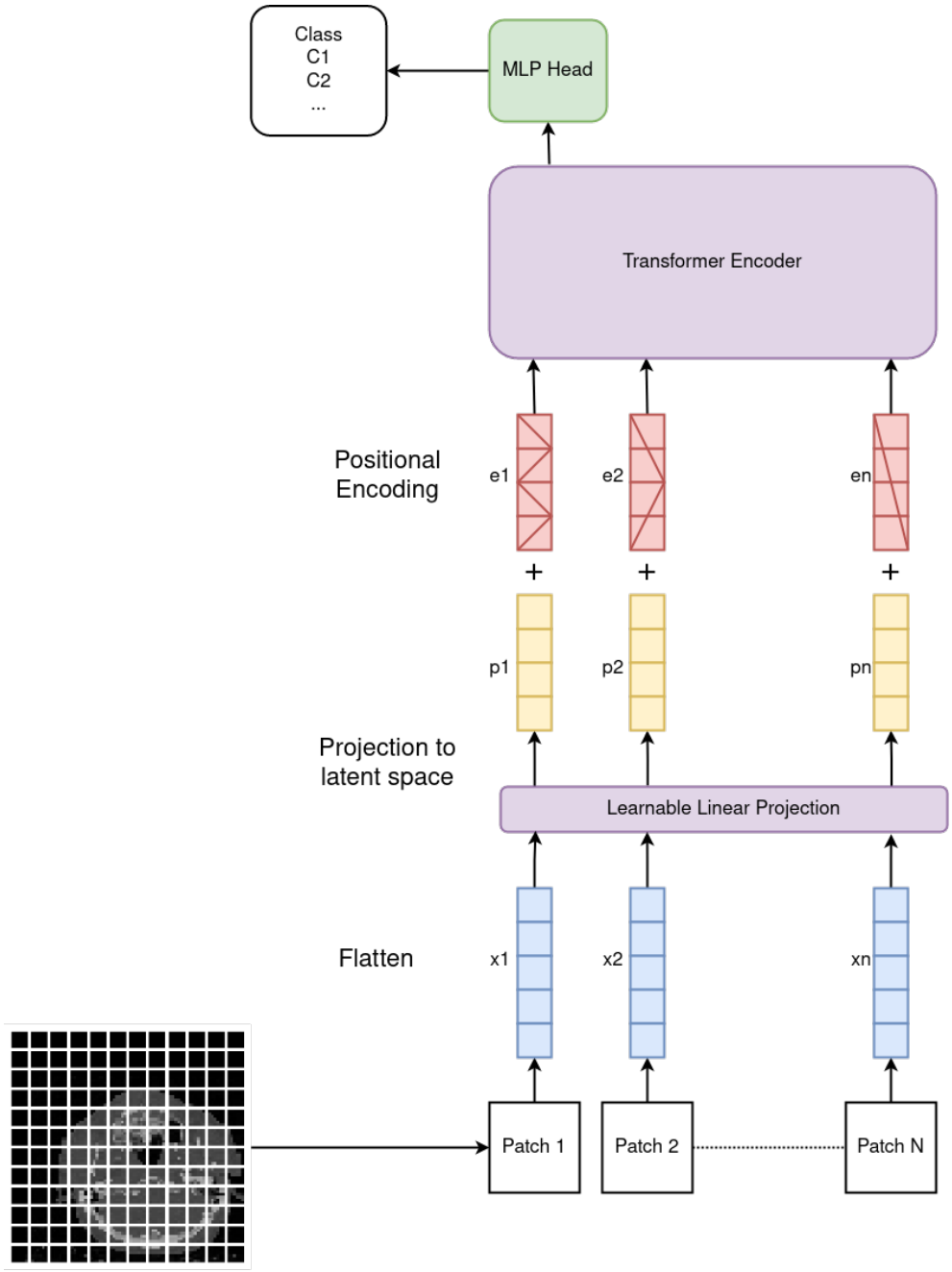

Figure 4. Vision Transformer architecture.

2.2. Attention Masks

When dealing with medical datasets and making medical decisions that can potentially be life-saving or, if erroneous, can cause death, it is crucial to establish the reasoning behind our decisions. This is even more important when ML models provide additional feedback and data to medical personnel, as it aids in explaining the model's decision-making process. The attention of vision transformers can be visualized using a technique called "attention rollout" [12]. Originally implemented as a technique for text-based transformers, it can be easily adapted for ViT.

In order to implement attention rollout for vision transformer, a process must be undertaken which involves iterating through each layer and recursively multiplying attention matrices with rollout matrices. This is illustrated in Equation (2).

$$R(l_i) = \begin{cases} A(l_i)R(l_{i-1}) & i > 0 \\ A(l_i) & i = 0 \end{cases} \quad (2)$$

However, in order to obtain meaningful results, it is important to consider the fusion of attention heads. This step ensures that we obtain a complete and accurate picture of attention in the network. One approach that we have used is the max fusion with discard approach. This involves taking attentions over all of the heads in a given layer, choosing the maximum values, and then discarding 90% of the lowest attention scores. This approach reduces the noise caused by unimportant attention patches, which is particularly relevant when dealing with data in a fast-paced medical environment.

It is worth noting that this approach contrasts with the mean fusion approach used in the original attention rollout paper [12], and while both approaches have their advantages, we have found that the max fusion with discard approach is particularly effective at reducing noise and obtaining accurate results.

2.3. Dataset

The ViT models were trained using the RSNA 2022 Cervical Spine Fracture Detection Challenge dataset [13]. This dataset consists of 2019 3D CT scans of the cervical spine area. Each CT scan is made up of roughly 100 to 800 slices of varying thickness and is labeled with a binary vector of length 7. The dataset also includes a set of 81 3D segmentations that provide pixel-perfect descriptions of the vertebrae. These segmentations were used to train the YOLOv5 object detection model to locate vertebrae on the CT scan slices. Additionally, the dataset includes bounding boxes that indicate areas of interest, such as damaged areas. However, most of the slices do not have such annotations, and, as a result, it is impossible to determine whether a given slice shows damaged vertebrae. In this work, we will focus on predicting a damaged/non-damaged label for a given CT scan slice.

Given ViT only works on individual slices of the scan, it was necessary to prepare labels for the model, since using labels for a whole CT scan would be misleading (an image can only contain 1 vertebrae, so the model should only mark that one visible vertebrae as damaged/non-damaged). In order to obtain positive cases, we used bounding box annotations provided with the dataset which marked the locations of damage in a vertebrae. Each slice with an annotation was marked as damaged, while slices without bounding boxes were marked as non-damaged. This resulted in a severe imbalance of classes, with only 7217 images being marked as damaged. In order to fix this imbalance, we applied undersampling to the whole dataset, which resulted in a 50:50 split between the classes while using all of the positive slices. The negative slices were chosen by randomly sampling from the negative pool of slices, then the pool of all slices was split in a 70:30 train-to-test ratio.

Before training, each slice undergoes a windowing processes in which densities similar to the spine bone density are enhanced using the parameters $W = 1800$, $L = 400$ [14]. During this process, each slice image is resized to a 224×224 size and then split into 16×16 slices (or 32×32 slices if such is required by the given architecture). A number of lightly applied augments were implemented, and these are shown in Table 1. The augmentation procedure helps with increasing image count, and since vision transformers are lacking due to some internal locality biases, a significant amount of data is needed. After all augmentations, the images were normalized, and the results are shown in Figure 5.

Table 1. Augmentations and their settings used in the training process.

Transform	Settings	Probability
RandomRotation	10 deg.	1
RandomTranslation	10% width, 10% height	1
RandomHorizontalFlip	Not applicable	0.5
RandomVerticalFlip	Not applicable	0.5
RandomGaussianBlur	kernel size = (5, 9), sigma = (0.1, 5)	0.5

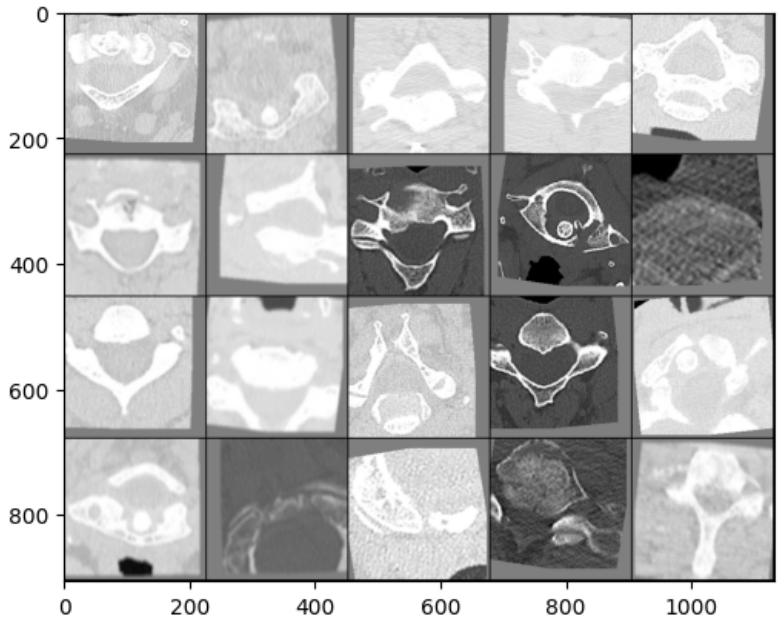

Figure 5. 20 sample images taken from dataset and augmented (without slicing into patches).

2.4. YOLOv5—Object Detection Model

In medical imaging, the ability to accurately identify and diagnose spinal injuries is critical. However, analyzing and interpreting the vast amount of information contained in CT scan slices can be a daunting task, especially when dealing with complex abnormalities like osteoporosis. In order to address this challenge, we implemented an object detection model to locate and extract the relevant information needed to identify damaged vertebrae.

To accomplish this, we selected YOLOv5 (You Only Look Once) [15], an object detection model (which can be seen in Figure A1), to identify vertebral regions in CT scan slices. By detecting the vertebrae and then extracting them from the image, we were able to focus our attention on the specific area of interest, which significantly reduced the amount of irrelevant information. Using this approach, we obtained bounding boxes on the detected vertebrae, which were then padded to preserve the aspect ratio of the region of interest. Finally, the area marked by the final bounding box was cut out and saved for later processing.

One of the biggest challenges we faced when implementing this approach was the lack of data for object detection. To overcome this, we leveraged the provided vertebrae segmentations to create a set of bounding box descriptions. To create a description for a given CT scan slice, we looked for the minimum X and Y coordinate of the non-zero pixels and similarly the maximum X and Y coordinate. Using all four anchor points, we then created a bounding box description. In total, we created 18,796 bounding boxes, which were then used for training.

We have chosen yolov5m model variant for this task, as it was a reasonable middle ground between accuracy and required compute. The model was trained for 100 epochs, after which have achieved 0.98 MAP score.

Overall, our approach using YOLOv5 object detection model has proven to be highly effective in reducing the amount of irrelevant information in CT scan slices and improving the accuracy of identifying damaged vertebrae via ViT model. By using bounding box descriptions, we were able to extract the specific region of interest, which allowed for more efficient and accurate processing of the data.

2.5. Data Pipeline

Putting it all together, a clear picture of the data pipeline emerges. First, one must take a CT scan and convert it to series of slices with proper windowing. WW: 1800, WL: 400, or WW: 2000; WL: 500 windowing will work the best, as they focus on the bone tissue density [16]. After conversion into slices, the object detection model will then crop the slice image to the target vertebrae in order to decrease the amount of unnecessary information coming to the classification model. Finally, cropped images are sent to the vision transformer model and classification probabilities are produced, as is shown in Figure 6.

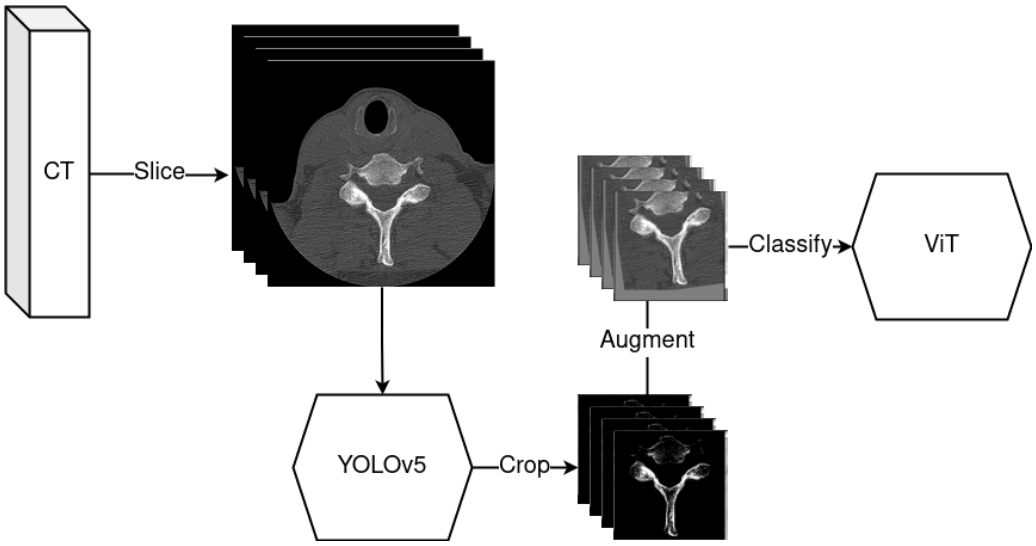

Figure 6. Data pipeline required for model to work.

This entire process is well suited to use in a cloud environment, and the resulting model of a self-learning fracture detection system could be devised as shown in Figure 7.

Hospitals can improve radiologists' ability to detect damaged vertebrae by providing them with a system that offers immediate analysis and highlights areas with a high probability of damage in specific slices. A probability map can be generated for the entire CT scan, allowing medical personnel to focus on the areas with a higher probability of damage.

CT scans can be automatically uploaded to cloud object storage, which stores raw scans for inference and later training purposes. The newly arrived scans are then sent to the data pipeline for inference. The inference results are stored in the analytics database and sent to medical personnel for use and review. Physicians can accept or disagree with the results and provide expert knowledge to the system by marking slices where the damage is visible or not. The results of the review process are then sent to the analytics database to be corrected.

After the system has been used for some time, a larger dataset will exist in the analytics database. This dataset can be used by the ML-Ops team to further improve and optimize the model. The results from the training are also sent to the analytics database for review by the ML-Ops team.

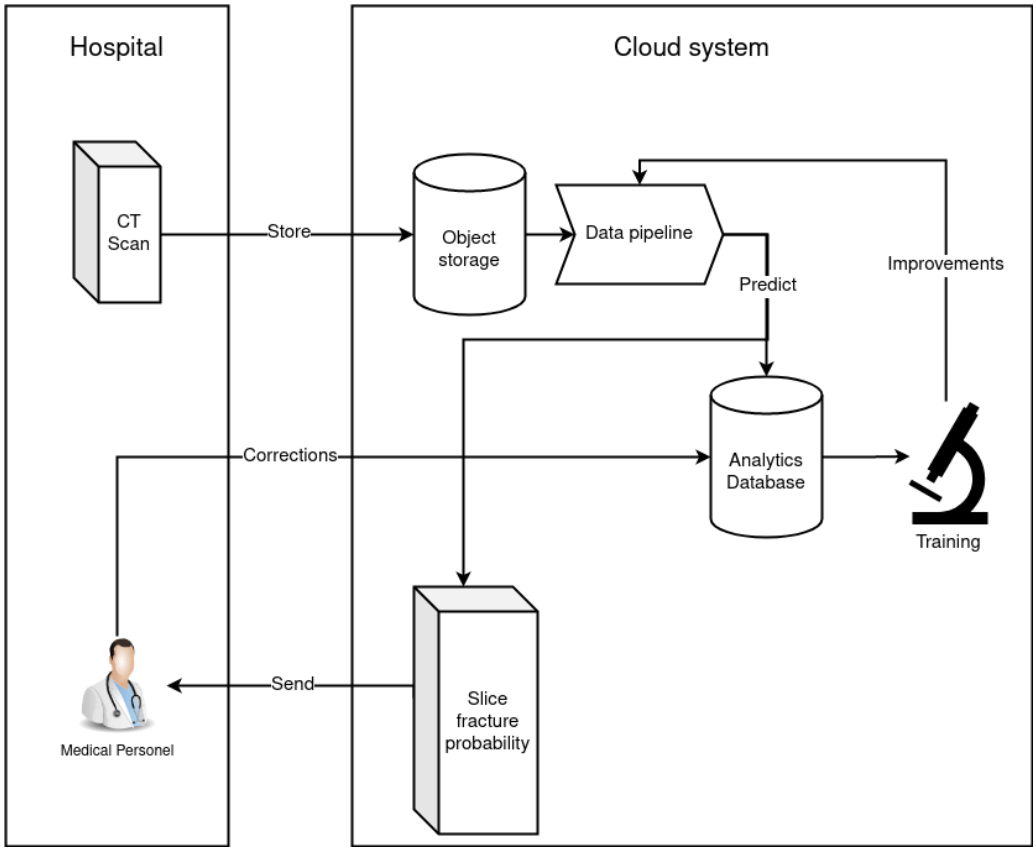

Figure 7. Proposed cloud system for processing medical data.

It should be noted that such tools are already available in various cloud environments, such as Azure or AWS (Amazon Web Services), making the implementation of this system a manageable task.

2.6. Model Training

The main goal of training was to attain the highest accuracy possible, and the task performed by the models is to detect whether or not the shown CT scan slice contains a fracture, making this task a binary classification problem.

The models were trained with the AdamW optimizer [17]. Each of the models was trained with a variable learning rate starting at $LR = 0.0002$. Due to the small size of the dataset, each model was trained for 100 epochs or 200 epochs (in the case of DeIT). This lengthy training process was necessary to ensure that the augmentation procedure was applied as diversely as possible. This was important because it helped to enhance the robustness of the models and make them more effective. After the training process, the model with the best validation accuracy was chosen for the results. The exact model parameters are shown in Table 2.

Table 2. Used Model architectures.

Model	Patch Size	Latent Space Dim	Encoder Blocks	MLP Heads	Parameters Total
ViT-B32	32×32	768	12	12	87,466,819
ViT-B16	16×16	768	12	12	87,466,819
DeIT-T16	16×16	192	12	3	5,524,802

3. Results

3.1. Model Training

While conducting research, many different architectures, sets of hyperparameters, and dataset splits were tested. Since the dataset was (for practical purposes) ideally balanced, the only metric that mattered was accuracy. Many of the tested models did not yield satisfactory results or did not converge at all.

Table 3 shows the model architecture used in a run and whether or not the aforementioned augmentation procedure was used. The training process results are shown in Figures A2–A5.

Table 3. Model results.

Model	Augmentation	Accuracy
ViT-B16	No	85%
ViT-B16	Yes	91%
ViT-B32	Yes	92%
DeIT-T16 (200 Epochs)	Yes	98%

Interestingly, the smallest DeiT-T16 model happens to be the most accurate one, but only after 200 epochs [18]. The other models trained faster but acquired lower accuracy. We attempted to train a vision transformer from scratch, but the performance of pre-trained architectures was significantly higher. Therefore, we have decided not to include the results.

During our research, we found that the augmentation procedure was a crucial part of our training process. We experimented with various augmentation techniques, including automated augmentation procedures like AugMix [19] and RandAug [20], as well as simple affine transformations, while most of the augmentations, with the exception of Gaussian blur, were a simple set of translations and rotations, we discovered that the automated augmentation procedures performed worse than our simple augmentation procedure. We believe that this is because automated augmentation procedures often manipulated the color in images that were supposed to be grayscale, resulting in outputs that could not be seen in any test case.

In contrast, the simple affine transformations aided the vision transformer in developing locality, which was necessary due to a lack of inductive biases. We hypothesize that these transformations helped the model learn to recognize patterns in the data and develop a better understanding of the underlying structure of the images. By contrast, the automated augmentation procedures resulted in outputs that were not representative of the underlying data, which could have led to overfitting and poor performance on test data. The influence of the augmentation procedure can be seen in the loss and test accuracy charts in Figures A2 and A3.

3.2. Attention Masks

We obtained attention masks for models using the aforementioned max fusion with discard technique. The masks shown in Figure 8 focus on the bone parts of the image, confirming that the vision transformer correctly identified the most important parts of the image. However, the masks also reveal that the vision transformer produces more noise in darker images, indicating that there is still room for improvement.

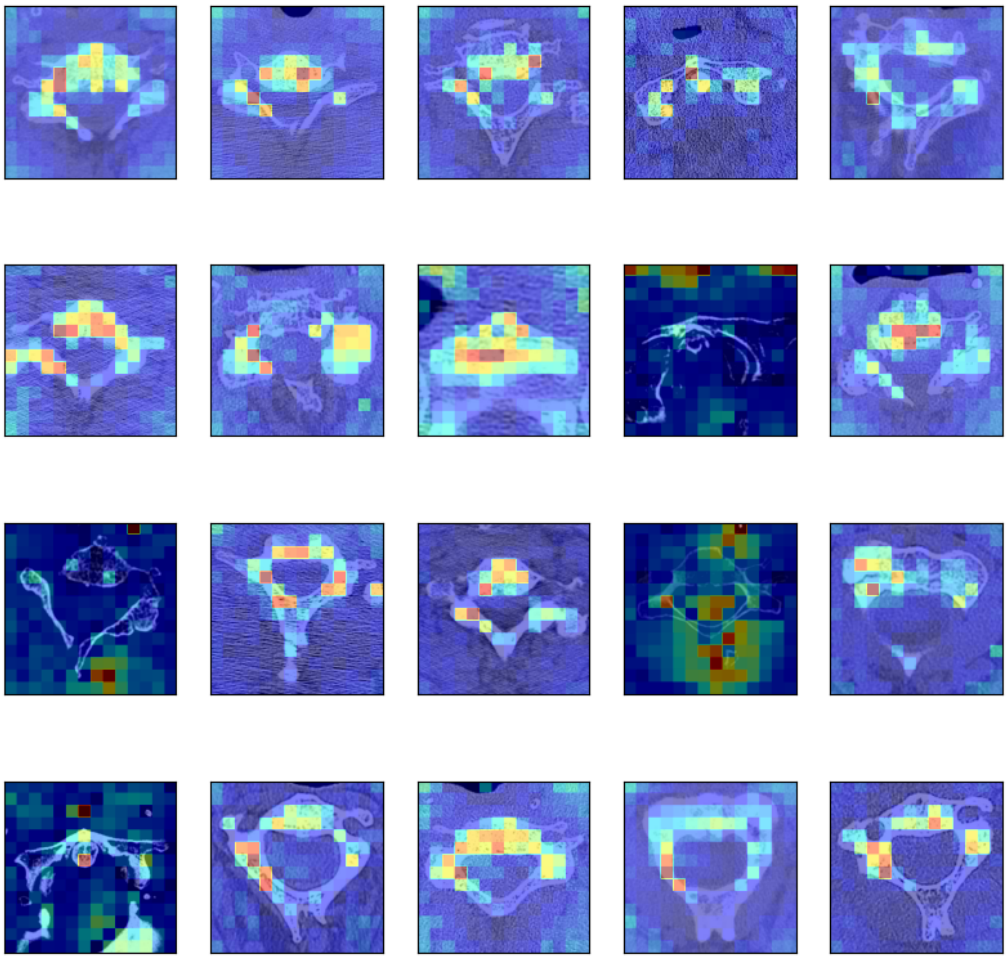

Figure 8. DeiT-T16 Attention maps for 20 selected images from the test set.

4. Discussion

All of the pre-trained models achieved competitive levels of accuracy (90–98%) [21,22] (92%) [23] for detecting vertebrae damage based on CT scan slices. The performance was similar to the performance of CNNs and ViTs shown in other studies focusing on medical aspect [24]. This is a significant achievement in the medical field as it allows for more accurate diagnoses and better treatment for cervical spine fracture detection. In addition to this, they provide attention heatmaps that explain the models' decisions, making vision transformers suitable for classification and detection tasks in medical settings [25,26]. These heatmaps can help doctors and medical professionals better understand the reasoning behind the model's decisions, allowing them to make more informed decisions.

The parallel nature of ViTs allows for easy scalability in both training and production environments, including cloud environments. This means that not only can the models be trained more quickly, but they can also be deployed more easily and efficiently. We have also shown that augmentation procedures have a very high impact on model performance. By using these procedures in combination with ViT, the accuracy is comparable to state-of-the-art solutions. This may be due to the lack of inductive biases exhibited by CNNs, while ViT can learn the implied localities of a given dataset, making it easier to learn the underlying structures of images. With the ability to use augmentation procedures, the models can be made even more accurate and effective.

The high performance of even the smallest models shows that it is viable and cost-effective to implement cloud solutions at scale. This means that more medical facilities can benefit from these models without having to invest in expensive hardware. The actual production-grade implementation needs to be conducted carefully, taking into

consideration the risks associated with the usage of AI and cloud technologies in the sensitive field of medicine. As our proposed system requires internet access to function properly, the data must be appropriately secured (encrypted) and preferably anonymized due to privacy concerns, which is especially important when dealing with medical data as it contains sensitive information [27,28]. It is also essential for the system to be reliable, as the increasing usage of AI technologies in medicine increases the risk of failure, which could endanger people's lives. Cloud-integrated AI seems to be an effective method of decreasing such failures, as these systems have been proven to be resilient. However, the data cannot be fully private, as they need to be decrypted for processing. For full data privacy, fully homomorphic encryption or AI models that can operate on encrypted data would be needed.

There is an issue with the adoption of such technologies in medical environments [29]. As medical personnel are often overworked, additional training for complex tool use can be challenging to execute. In order to improve the adoption of AI in medicine, tools need to be designed in a way that does not require intensive training. Ideally, the tools would be integrated with medical equipment in order to provide AI-enhanced results immediately. Another important aspect is to focus on the observability of results, which can indicate the technological impact on patient outcomes. Positive results can be a strong persuasion point for the use of such technologies. Additionally, cost is a significant factor that management personnel are likely to consider, which, in the case of cloud-based systems, is reduced due to the reduction in manpower and infrastructure needs [30].

There is also a philosophical aspect to the risk involved in integrating AI systems into medicine: the question of responsibility for medical errors. Who is responsible for incorrect predictions made by AI algorithms? As these algorithms become increasingly more efficient over time, whose fault will it be if someone dies due to a medical mistake? We leave this question to the readers to consider.

5. Conclusions

We conclude that vision transformers are a viable alternative to CNN architectures with comparable performance, given the augmentation procedures. They can be used to build advanced cloud-based medical systems which can further improve the working speed of medical personnel and patient outcomes in cases of vertebral fractures.

Author Contributions: P.C.: conceptualization, methodology, software, validation, formal analysis, investigation, writing—original draft preparation, and visualization. M.R.O.: conceptualization, methodology, validation, writing—review and editing, and supervision. All authors have read and agreed to the published version of the manuscript.

Funding: AGH University of Science and Technology: This work was partially supported by the funds of the Polish Ministry of Education and Science assigned to AGH University of Science and Technology.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset used in this paper is publicly available at [13]. The rest of the data are available from the authors upon request.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

MDPI	Multidisciplinary Digital Publishing Institute
DOAJ	Directory of open access journals
ViT	Vision Transformer
CNN	Convolutional Neural Network

CT	Computer Tomography
DL	Deep Learning
ML	Machine Learning
AR	Attention Rollout
MLP	Multi Layer Perceptron
MHA	Multi Head Attention
AWS	Amazon Web Services

Appendix A

Appendix A.1

Figure A1 shows the yolov5 architecture. It is composed of multiple convolutional layers that transform the input image into feature maps. Thanks to concatenation of outputs of intermediate layers, the model is able to differentiate between features on multiple scales. The C3 (Concentrated-Comprehensive Convolution) modules allow for image segmentation which are layer used by the model to differentiate between individual object on an image [31]. The SPPF layer is mathematically equivalent to the SPP layer [32], but are more optimized.

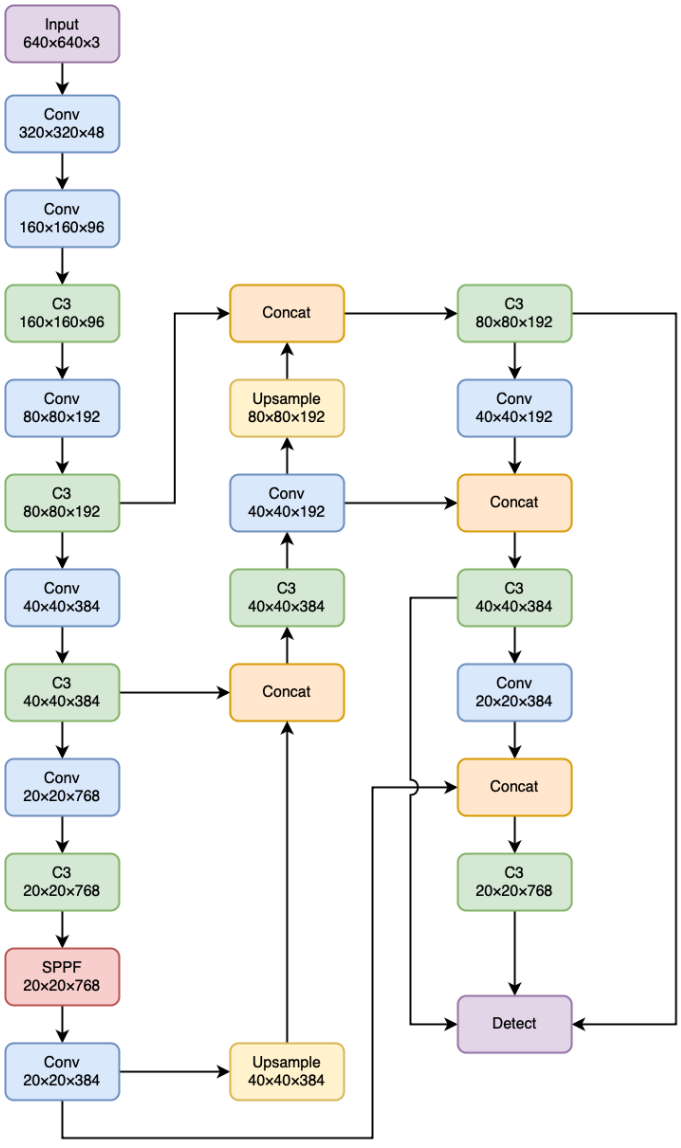

Figure A1. Yolov5m architecture.

Figures A2–A5 show the training process of selected model architectures. The particularly interesting is comparison of Figures A2 and A3 as one can see the difference that the augmentation procedure does to the training process. In a model without augmentation one can see a high degree of overfitting.

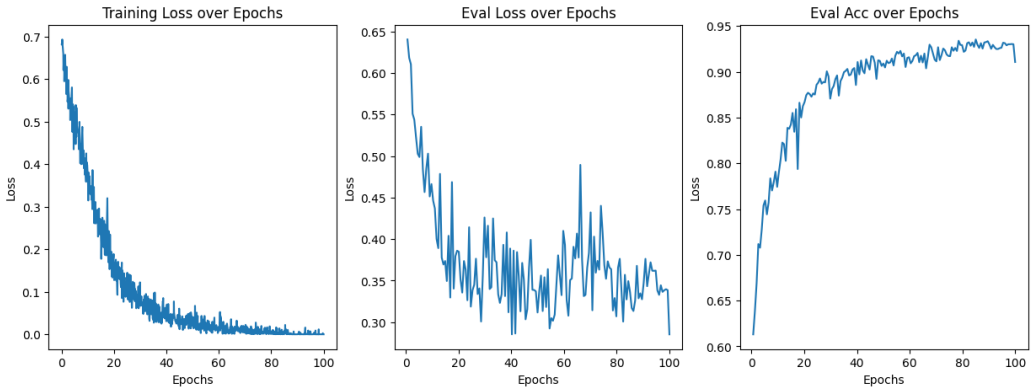

Figure A2. Training statistics of ViT-B16.

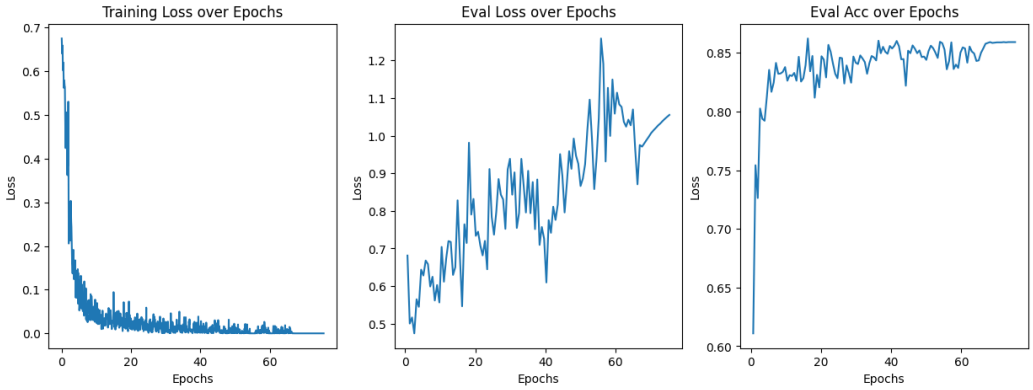

Figure A3. Training statistics of ViT-B16 without augmentation procedure.

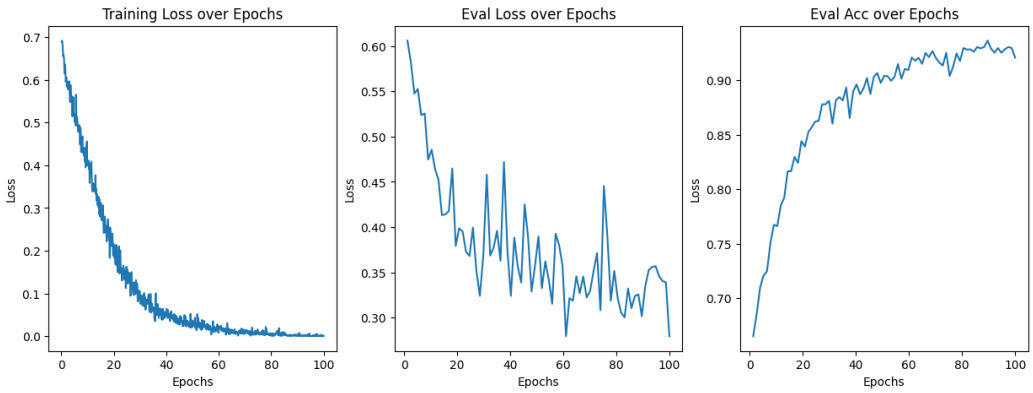

Figure A4. Training statistics of ViT-B32.

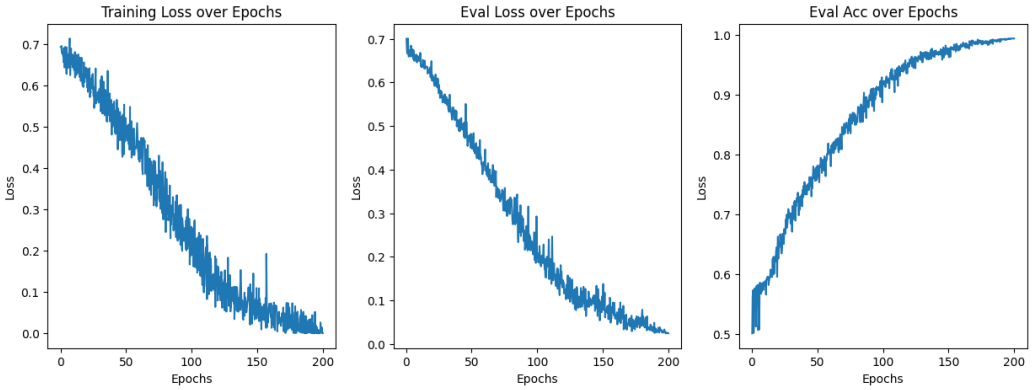

Figure A5. Training statistics of DeIT-T16.

References

1. Dong, Y.; Peng, R.; Kang, H.; Song, K.; Guo, Q.; Zhao, H.; Zhu, M.; Zhang, Y.; Guan, H.; Li, F. Global incidence, prevalence, and disability of vertebral fractures: A systematic analysis of the global burden of disease study 2019. *Spine J.* **2022**, *22*, 857–868. https://doi.org/10.1016/j.spinee.2021.12.007.
2. World Health Organization. Spinal Cord Injury. 2013. Available online: https://www.who.int/news-room/fact-sheets/detail/spinal-cord-injury (accessed on 19 March 2023).
3. Ismael Aguirre, M.F.; Tsirikos, A.I.; Clarke, A. Spinal injuries in the elderly population. *Orthop. Trauma* **2020**, *34*, 272–277. https://doi.org/10.1016/j.mporth.2020.06.004.
4. Fehlings, M.G.; Perrin, R.G. The Timing of Surgical Intervention in the Treatment of Spinal Cord Injury: A Systematic Review of Recent Clinical Evidence. *Spine* **2006**, *31*, S28–S35.
5. Meena, T.; Roy, S. Bone Fracture Detection Using Deep Supervised Learning from Radiological Images: A Paradigm Shift. *Diagnostics* **2022**, *12*, 2420. https://doi.org/10.3390/diagnostics12102420.
6. Perotte, R.; Lewin, G.O.; Tambe, U.; Galorenzo, J.B.; Vawdrey, D.K.; Akala, O.O.; Makkar, J.S.; Lin, D.J.; Mainieri, L.; Chang, B.C. Improving Emergency Department Flow: Reducing Turnaround Time for Emergent CT Scans. *AMIA Annu. Symp. Proc.* **2018**, *2018*, 897–906.
7. Amazon Corporation. *AWS SageMaker*; Amazon Corporation: Seattle, WA, USA, 2023.
8. Microsoft Corporation. *Train and Deploy Machine Learning Anywhere*; Microsoft Corporation: Redmond, WA, USA, 2022.
9. Farhadi, F.; Barnes, M.R.; Sugito, H.R.; Sin, J.M.; Henderson, E.R.; Levy, J.J. Applications of artificial intelligence in orthopaedic surgery. *Front. Med. Technol.* **2022**, *4*, 995526. https://doi.org/10.3389/fmedt.2022.995526.
10. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image is Worth 16 × 16 Words: Transformers for Image Recognition at Scale. *arXiv* **2021**, arXiv:2010.11929.
11. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. *arXiv* **2017**, arXiv:1706.03762.
12. Abnar, S.; Zuidema, W. Quantifying Attention Flow in Transformers. *arXiv* **2020**, arXiv:2005.00928.
13. Flanders, A.; Carr, C.; Colak, E.; FelipeKitamura; Lin, H.M.; JeffRudie; Mongan, J.; Andriole, K.; Prevedello, L.; Riopel, M.; et al. RSNA 2022 Cervical Spine Fracture Detection. 2022. Available online: https://www.kaggle.com/competitions/rsna-2022-cervical-spine-fracture-detection/overview (accessed on 5 March 2023).
14. Murphy, A. Windowing (CT). Reference Article, Radiopaedia.org, 2017. Available online: https://radiopaedia.org/articles/52108 (accessed on 5 March 2023). https://doi.org/10.53347/rID-52108.
15. Jocher, G.; Chaurasia, A.; Stoken, A.; Borovec, J.; NanoCode012; Kwon, Y.; Michael, K.; TaoXie; Fang, J.; Imyhxy; et al. ultralytics/yolov5: v7.0 - YOLOv5 SOTA Realtime Instance Segmentation. 2022. Available online: https://zenodo.org/record/7347926#.ZEE-y85ByUk (accessed on 5 March 2023). https://doi.org/10.5281/zenodo.7347926.
16. Bogdan Pruszyński, A.C. *Radiologia Diagnostyka obrazowa RTG TK USG i MR*; PZWL Wydawnictwo Lekarskie: Warszawa, Poland, 2014; p. 54.
17. Loshchilov, I.; Hutter, F. Decoupled Weight Decay Regularization. *arXiv* **2019**, arXiv:1711.05101.
18. Hugo, T.; Matthieu, C.; Matthijs, D.; Francisco, M.; Alexandre, S.; Herve, J. Training data-efficient image transformers & distillation through attention. In Proceedings of the International Conference on Machine Learning, Virtual, 18–24 July 2021; Volume 139, pp. 10347–10357.
19. Hendrycks, D.; Mu, N.; Cubuk, E.D.; Zoph, B.; Gilmer, J.; Lakshminarayanan, B. AugMix: A Simple Data Processing Method to Improve Robustness and Uncertainty. *arXiv* **2019**. https://doi.org/10.48550/ARXIV.1912.02781.
20. Cubuk, E.D.; Zoph, B.; Shlens, J.; Le, Q.V. RandAugment: Practical automated data augmentation with a reduced search space. *arXiv* **2019**. https://doi.org/10.48550/ARXIV.1909.13719.
21. Burns, J.E.; Yao, J.; Summers, R.M. Vertebral Body Compression Fractures and Bone Density: Automated Detection and Classification on CT Images. *Radiology* **2017**, *284*, 788–797. https://doi.org/10.1148/radiol.2017162100.

22. Wang, G.; Wu, Y.; Sun, Q.; Yang, B.; Zheng, Z. SID2T: A Self-attention Model for Spinal Injury Differential Diagnosis. In Proceedings of the Intelligent Computing Theories and Application, 18th International Conference, ICIC 2022, Xi'an, China, 7–11 August 2022; Huang, D.S., Jo, K.H., Jing, J., Premaratne, P., Bevilacqua, V., Hussain, A., Eds.; Springer International Publishing: Cham, Switzerland, 2022; pp. 650–662.
23. Small, J.E.; Osler, P.; Paul, A.B.; Kunst, M. CT Cervical Spine Fracture Detection Using a Convolutional Neural Network. *AJNR Am. J. Neuroradiol.* **2021**, *42*, 1341–1347. https://doi.org/10.3174/ajnr.A7094.
24. Nafisah, S.I.; Muhammad, G.; Hossain, M.S.; AlQahtani, S.A. A Comparative Evaluation between Convolutional Neural Networks and Vision Transformers for COVID-19 Detection. *Mathematics* **2023**, *11*, 1489. https://doi.org/10.3390/math11061489.
25. He, K.; Gan, C.; Li, Z.; Rekik, I.; Yin, Z.; Ji, W.; Gao, Y.; Wang, Q.; Zhang, J.; Shen, D. Transformers in medical image analysis. *Intell. Med.* **2023**, *3*, 59–78. https://doi.org/10.1016/j.imed.2022.07.002.
26. Li, J.; Chen, J.; Tang, Y.; Wang, C.; Landman, B.A.; Zhou, S.K. Transforming medical imaging with Transformers? A comparative review of key properties, current progresses, and future perspectives. *Med. Image Anal.* **2023**, *85*, 102762. https://doi.org/10.1016/j.media.2023.102762.
27. Radanliev, P.; De Roure, D. Disease X vaccine production and supply chains: Risk assessing healthcare systems operating with artificial intelligence and industry 4.0. *Health Technol.* **2023**, *13*, 11–15. https://doi.org/10.1007/s12553-022-00722-2.
28. Inukollu, V.N.; Arsi, S.; Ravuri, S.R. Security issues associated with big data in cloud computing. *Int. J. Netw. Secur. Its Appl.* **2014**, *6*, 45.
29. Safi, S.; Thiessen, T.; Schmailzl, K.J. Acceptance and Resistance of New Digital Technologies in Medicine: Qualitative Study. *JMIR Res. Protoc.* **2018**, *7*, e11072. https://doi.org/10.2196/11072.
30. John, N.; Shenoy, S. Health cloud—Healthcare as a service(HaaS). In Proceedings of the 2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Delhi, India, 24–27 September 2014; pp. 1963–1966. https://doi.org/10.1109/ICACCI.2014.6968627.
31. Park, H.; Yoo, Y.; Seo, G.; Han, D.; Yun, S.; Kwak, N. C3: Concentrated-Comprehensive Convolution and its application to semantic segmentation. *arXiv* **2019**, arXiv:1812.04920.
32. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. In *Computer Vision—ECCV 2014*; Springer International Publishing: Cham, Switzerland, 2014; pp. 346–361. https://doi.org/10.1007/978-3-319-10578-9_23.