Advanced Topics in Online Privacy and Cyber-security (67515) - Final Project

Noam Ben Sason - 318505260

1 Project Idea

Today, machine learning (and particularly deep learning) techniques are very popular, since they achieve excellent results in many tasks. These techniques often include a model that learns a large data set, which may be crowd-sourced and contain sensitive information. In order to learn meaningful information from the data, the ML models maintains a set of weights (parameters), that are updated during the training of the ML model (more on the training process later).

One would want to avoid an attack such as a reconstruction attack, in which an attacker would be able to reconstruct data examples or details about it from the model's weights, gradients or outputs.

In particular, privacy is very important in the medical domain. One would not want an attacker to be able to infer details about the patients that their data wad included in a medical dataset that was used to train a model. Medical confidentiality is very important, and when broken can have consequences in the personal, health, and legal aspects.

I chose to use a differential private mechanism on deep learning models to achieve this goal. I chose the paper "Deep Learning with Differential Privacy", that present the DP-SGD algorithm, an algorithm, to be used during the training of a deep neural network model, in order to achieve privacy given a budget of (ϵ, δ).

In this project I will train, evaluate and compare between 2 model types, based on 2 different architectures - CNN (convolution based) and ViT (transformer based). In particular, I am interested in exploring the affect of adding DP to both of them, and to check if defining a privacy budget "hurts" one architecture more than the other, and how different architecture influence the budget we can define without hurting the utility (for example - do we have to define a smaller budget for one architecture to keep up with the utility of the other architecture?).

For this project I chose a data set of Chest X-Ray images, that was made for a classification task on different diseases (Pneumonia, Covid-19,Tuberculosis, or no disease). The full dataset can be found here.

For the implementation of the privacy mechanism I will be using the 'Opacus' library, a differential privacy library that works well with the library 'pytorch', a library for deep models and training.

From the pre-research I did, this is a new and (I believe) interesting experiment. I was not able to find recorders on training and evaluating a transformer-based model with a DP-SGD mechanism on this sensitive medical chest x-ray data, and was not able to find a comparison of this experiment to the more commonly-used CNN model in this setting.

2 Differential Privacy

Differential privacy (DP) is a strong standard for privacy guarantees for algorithms on aggregate databases. The purpose of DP is to give the ability to release statistical information about datasets while protecting the privacy of individual data subjects.

I will be using the following definition:

A randomized mechanism $M : D \to R$ with domain D and range R satisfies (ϵ, δ)-differential privacy if for any two adjacent inputs $d, d' \in D$ and for any subset of outputs $S \subseteq R$ it holds that

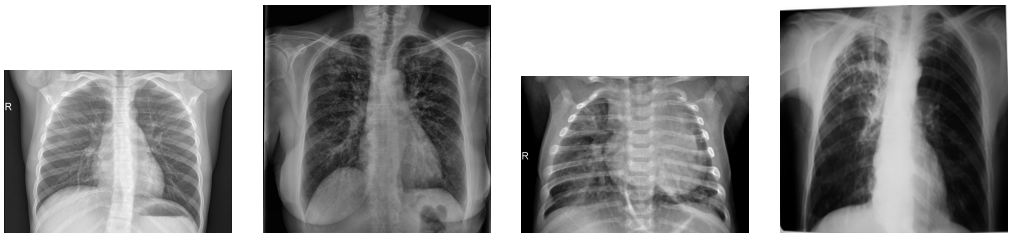
$$Pr[M(d) \in S] \leq e^{\epsilon} \cdot Pr[M(d') \in S] + \delta$$

The ϵ is the upper bound for our privacy loss. The δ is the failure probability.

More on the differential privacy mechanism used in this project will be detailed later (DP-SGD).

3 Chest X-ray Data

I wanted to use a medical data for this project, and came across in my searches in the chest X-ray dataset. The dataset contains 7135 x-ray photos, taken from patients with different conditions, like Pneumonia, Covid-19,Tuberculosis, and also from patients that are not sick. All those conditions are labeled for classification. The training set consists of 6326 examples, and the test set consists of 773 examples.



Healthy	Covid-19	Pneumonia	Tuberculosis

Table 1: Chest X-rays of patients with different conditions.

This data is sensitive in so many ways. First, there are many details about a patients that can be inferred from the images, such as gender, age, and race.

For example, it is possible to differentiate between a male and female chest in x-ray. Source below ('radiology key'). The major difference between male and female chest x-rays is caused by differences in the amount of breast tissue, and in female X-rays, the nipple shadows can also be seen:

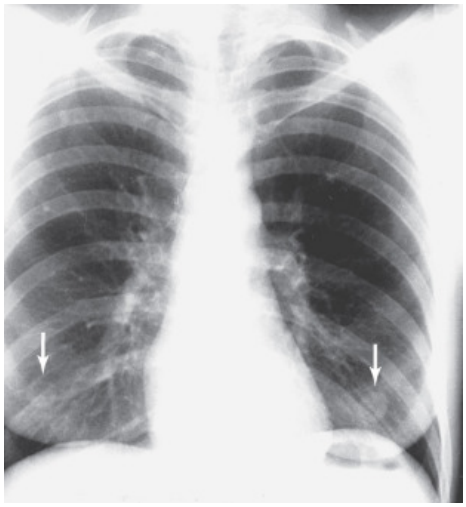
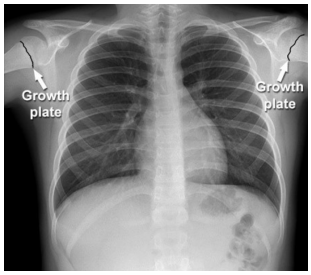
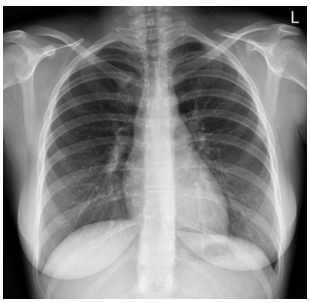


Figure 1: A female chest X-ray. The nipple shadows are pointer with arrows.

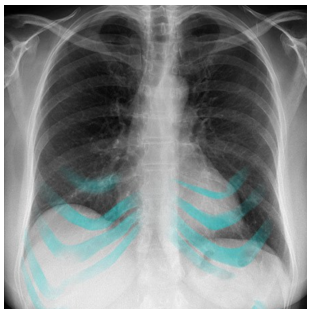
Another example for a detail like this is age - it can be inferred from the chest x rays. Young children have growth plates, and their rib cage is smaller, compared to older people with bigger rib cages. Source below (from 'radiology masterclass').



11 yeas old



15 years old

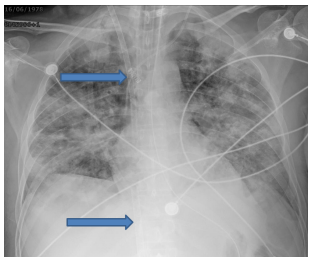


77 years old

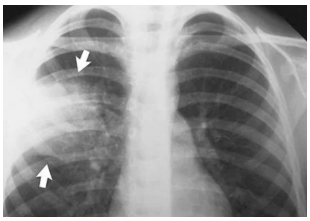
Table 2: Chest X-rays of patients in different ages

Another detail about the patients that can be inferred from the photos is of course the condition or disease the patient has.

Here are some examples for the image-features that can indicate that the patient have a specific disease:



Covid-19



Pneumonia

Tuberculosis

Table 3: Chest X-rays of patients with different diseases

About table 3:

1. Covid-19: The cloudy texture indicates covid-19.
2. Pneumonia: This chest X-ray shows an area of lung inflammation indicating the presence of pneumonia.
3. Tuberculosis: Chest x-ray showing discrete round nodule(s) with round edges, after secondary tuberculosis.

So we can see there are many sensitive detail in this kind of data. Once someone gets those details about a patients, he could cross information between datasets and infer a specific patients have a specific condition.

All those graphic features can be identify by a neural network, and could be represented in its weights. In later sections I will explain how such features could be represented in the model weights, and how we can prevent a potential attacker to take advantage of this fact.

This section gives us a strong motivation for this project - to find a good way to preform tasks like classification, with a controlled privacy loss.

4 Deep Learning Optimization

In order to understand the DP-SGD algorithm, we first need to understand the optimization algorithm of a neural network. There are multiple versions of such optimizers, but in this project I will be discussing only SGD (the basic algorithm) and Adam (a version of SGD).

Stochastic Gradient Descent (SGD) is an iterative optimization algorithm commonly used in deep learning. It updates model parameters (weights) by computing gradients on batches of the training data. The gradients guide the model on how to update parameters to minimize the loss function, pointing in the direction of steepest descent. By following the negative gradient of the loss function, the algorithm adjust the parameters iteratively to converge towards a solution.

The Adaptive Moment Estimation (Adam) optimizer is very similar to SGD, with two additions: It adapt the learning rate mid training based on the size of the gradient, and also "remembers" past gradients by maintaining a linear combination of them with the new gradients.

In the following sections I will describe the DP-SGD algorithm, which is an additional mechanism that can be added to either SGD or Adam.

5 DP-SGD Algorithm

The DP-SGD is a differential privacy mechanism that can be added to the SGD/ADAM optimizers.

The DP-SGD algorithm presented in the paper:

Algorithm 1 Differentially private SGD

Require: Examples $\{x_1, \ldots, x_N\}$, loss function $\mathcal{L}(\theta) = \frac{1}{N}\sum_i \mathcal{L}(\theta, x_i)$. Parameters: learning rate η_t, noise scale σ, group size L, gradient norm bound C.

Initialize θ_0 randomly
for $t \in [T]$ **do**
    Take a random sample L_t with sampling probability L/N
    Compute gradient
For each $i \in L_t$, compute $g_t(x_i) \leftarrow \nabla_\theta \mathcal{L}(\theta_t, x_i)$
    Clip gradient
$\bar{g}_t(x_i) \leftarrow g_t(x_i) / \max\left(1, \frac{\|g_t(x_i)\|_2}{C}\right)$
    Add noise
$\tilde{g}_t \leftarrow \frac{1}{L}\left(\sum_i \bar{g}_t(x_i) + \mathcal{N}(0, \sigma^2 C^2 I)\right)$
    Descent
$\theta_{t+1} \leftarrow \theta_t - \eta_t \tilde{g}_t$
end for
Output θ_T and compute the overall privacy cost (ϵ, δ) using a privacy accounting method.

As one can see, there are 3 changes between the regular SGD and the DP-SGD:

1. Clipping the gradient's norm using a threshold C. This clipping ensures that the gradient's norm is at most C. The purpose of this clipping is to limits the meaningful information the model learns from the data in each step.

2. Adding Gaussian noise to the gradients to force randomness of the learning.

3. Sampling random examples to the current batch instead of just dividing the data in advance

Note that those 3 changes can also be applied on the Adam optimizer.

The paper also present a accounting method to track the loss of privacy in training time.

6 Neural Network Models

6.1 CNN model

A Convolutional Neural Network (CNN) is a type of deep learning model particularly well-suited for processing grid-like data, such as images. It uses kernels (filters) are small, learnable matrices that slide over the input data (e.g. an image), performing a convolution operation. As they move across the data, they

detect specific features such as edges, textures, or patterns by computing dot products between the kernel values and the input data. During training, the model adjusts these kernels to detect increasingly complex and relevant features at different layers. This helps the CNN model capture spatial hierarchies of features, from simple patterns in early layers to complex objects in deeper layers (link to paper in bibliography).

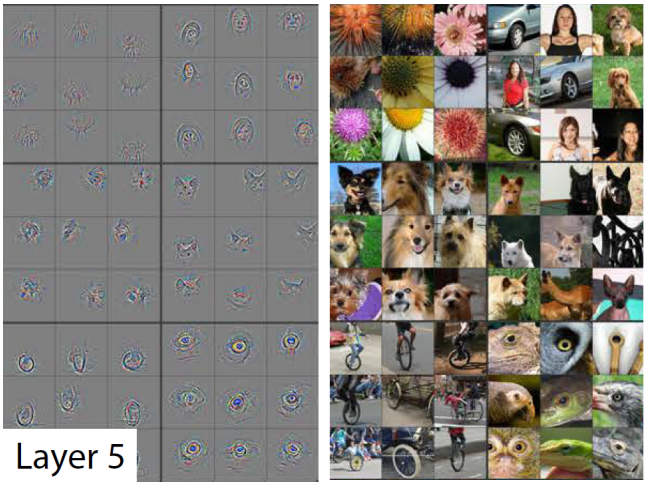


Figure 2: Visualization of the different patterns learned by a CNN model. From "Visualizing and Understanding Convolutional Networks" by Matthew D. Zeiler and Rob Fergus (2013)" (link to paper in bibliography)

A CNN model can identify in the chest X-ray images features that are indicating a certain disease, and also features like gender, age, race and more. This can be seen in the data section.

6.2 ViT model

The Vision Transformer (ViT) is a transformer-based model for vision tasks. The model was proposed in "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale" in 2021. Until this paper, the transformer architecture considered useful for mainly NLP task. This is the first paper that successfully trains a Transformer encoder on images data, attaining very good results compared to familiar convolutions architectures.

ViT attains excellent results compared to state-of-the-art convolutional networks while requiring substantially fewer computational resources to train.

ViT uses a mechanism called self-attention. The mechanism is used for modeling the relationships between different patches of an image. In self-attention,

each patch interacts with all other patches to capture global dependencies. The attention mechanism allows the model to weigh the importance of other patches relative to the current patch.

In this project I used this implementation of ViT: `https://github.com/tintn/vision-transformer-from-scratch/tree/main`

7 Attacker Model

We assume a potential adversary might have:

1. Access to the trained model (interaction with the model via inputs and outputs). This access is assumed in attacks such as 'model-inversion attack'.

2. Full knowledge of the training mechanism and access to the model's parameters.

As I mentioned earlier, a sensitive information about identified pattern in the data can we stored in some way in the model's weights. Because of that, there is a concern that the attacker might infer a specific photo was in the dataset, or a specific sensitive information found in a specific photo, that can be capture by the models (disease, gender, ethnicity - all that can be infer from looking at a chest x-ray of a patient) and be saved in some form in the model's weights. After inferring this information, the attacker might cross information with other databases (based of gender, age, etc) and infer some patient have a specific disease.

8 State of the Art

Before starting this project I did some research on what experiments have already been done on the topic. I was not able to find experiment like what I wanted to do - training and evaluating a transformer-based model with a DP-SGD mechanism on this chest x-ray data. I also did not find a comparison of an experiment like this to a CNN based model in this setting.

What I did found were different components of this experiment, but not together:

1. Detecting Covid-19 (only Covid-19, no other diseases) in chest X-ray images using differential privacy and . This was not done with a transformer-based model like ViT. `https://www.frontiersin.org/journals/medicine/articles/10.3389/fmed.2024.1409314/full`.

2. Using ViT to classify chest x-rays (without any privacy mechanism) - `https://arxiv.org/pdf/2304.11529`.

9 Hyper-parameters selection for Training

There are a few hyper-parameters for the training of the models that needs to be set:

1. Batch size: the number of examples that we give to the model at the same time at training time.
2. Test batch size: the number of examples that we give to the model at the same time at evaluation time.
3. Learning rate (LR): the size of the steps we take each time we update the model's weights.
4. number of epochs: the number of times the model receives the entire dataset.
5. optimizer: the optimization algorithm to update the model's weights.

For the search in the hyper parameter space, I use the tool 'Weights and Biases' (wandb). The goal was to find hyper-parameters that optimize the accuracy of the classification of the model. The wandb tool is a logging tool, a hyper-parameters search tool, and can also be used for plot generation.

I tried many sets of hyper-parameters for each model, and chose the best one for each of them. The chosen sets are detailed below.

I chose those hyper parameters using the two models (CNN, ViT) without the differential privacy mechanism. Later, I am going to compare the two models without the DP mechanism to the same models with the DP. For this comparison to make sense I made sure all the same model types have the same hyper-parameters set.

9.1 CNN hyper parameters

Hyper Parameter	**Value**
BATCH SIZE	64
LR	0.001
NUM EPOCHS	10
OPTIMIZER	Adam
TEST BATCH SIZE	128

Table 4: Hyper Parameters for the CNN model

9.2 ViT hyper parameters

For the ViT model there are some more hyper-parameters needed to be set. Those are mostly architecture hyper parameter.

Hyper Parameter	Value
BATCH SIZE	32
hidden dropout prob	0
hidden size	128
initializer range	0.02
intermediate size	512
LR	0.00001
num attention heads	6
NUM EPOCHS	10
num hidden layers	4
OPTIMIZER	Adam
patch size	4
qkv bias	true
TEST BATCH SIZE	128

Table 5: Hyper Parameters for the ViT model

10 Evaluation

10.1 Method

For evaluation I used the test set of the chest X-ray dataset, that consists of 773 examples.

I chose to evaluate the classification task with the 'accuracy' metric: If x is the number of examples that were labeled correctly by the model, and n is the total number of examples, then the accuracy if $\frac{x}{n}$. I also evaluated the test loss, using the 'Cross entropy' loss, even though I think this is a less informative metric for the project's task.

10.2 Result for base model (no DP)

I evaluated the 2 models with the sets of training-hyper-parameters described above, and got the following results:

	Test Accuracy	Test Loss (Cross Entropy)
CNN	0.801	0.01
ViT	0.748	0.005

Table 6: Results for the 2 compared models without the differential privacy mechanism.

One can see that the accuracy of the CNN model is higher that the ViT's accuracy. I will discuss this in the 'Discussion' section

10.3 Exploring the effect of the privacy hyper-parameters

In this section I will try to analyze the effect that each on the privacy hyper-parameter have on the performance of the models. For each model and for each hyper parameter, I will fix all the other hyper parameters and analyze the impact of the specific hyper parameter.

I will be exploring the effect of 3 hyper parameters:

1. epsilon (ϵ): Upper bound for out privacy loss
2. delta (δ): Probability of information being leaked.
3. max grad norm (C): The maximum norm of the per-sample gradients. Any gradient with norm higher than this will be clipped to this value.

Note that given a budget, the Opacus's privacy engine will calculate automatically an appropriate noise multiplier (σ) to ensure privacy budget of (ϵ,δ) at the end of epochs.

10.3.1 Privacy hyper-parameters in CNN

I trained the CNN model with the DP-SGD mechanism. I wanted to explore the effect different privacy parameter have on this model (and later on - on the ViT model).

I set for each of the 3 hyper-parameters a set of values to explore:

1. ϵ: [0.25,0.5,1,3,8,12,20]
2. δ: [0.001,0.0001,0.00001,0.000001]
3. C: [0.25,0.5,1,2,3]

For each set of hyper-parameters, I run an experiments of training and evaluation, and used the wandb tool to log it. I created a graph using wandb that shows the different effect each hyper parameter has on the test accuracy:

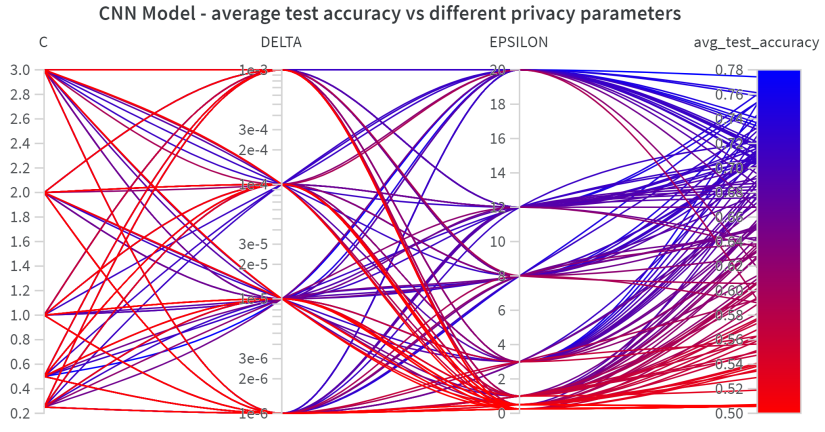


Figure 3: The effect each of the privacy hyper parameters has on the test accuracy using the CNN model. In the plot each line is a run with different combination of the hyper parameter. The bluer the run, the higher the accuracy. The redder the run, the lower the accuracy.

Even without fixing any values yet, we can see a clear pattern - the lower the epsilon (0.25,0.5,1) the lower the test accuracy (redder), and the higher the epsilon (8,12,20) - the higher the accuracy (bluer).

Because it is hard to see the affect of the other hyper-parameters, I will also do the value-fixing I described above:

- Effect of ϵ (fixing $\delta = 0.000001$, C $= 1$):

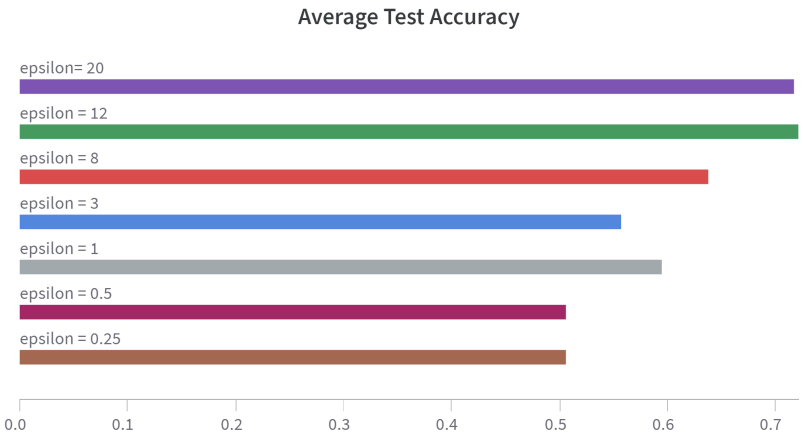


Figure 4: The effect of the hyper-parameter ϵ on the test accuracy.

We can see here what we already saw in the last graph - the epsilon has a big effect on the accuracy, and it makes a lot of sense. The epsilon is the threshold of the privacy loss, it determine the scale of privacy that will be kept. A big epsilon means the privacy loss will potentially be bigger, so there will be less restrictions and more potential for larger accuracy. A small epsilon means bigger constraints on the privacy loss and less potential for a high accuracy.

- Effect of δ (fixing $\epsilon = 3$, C $= 1$):

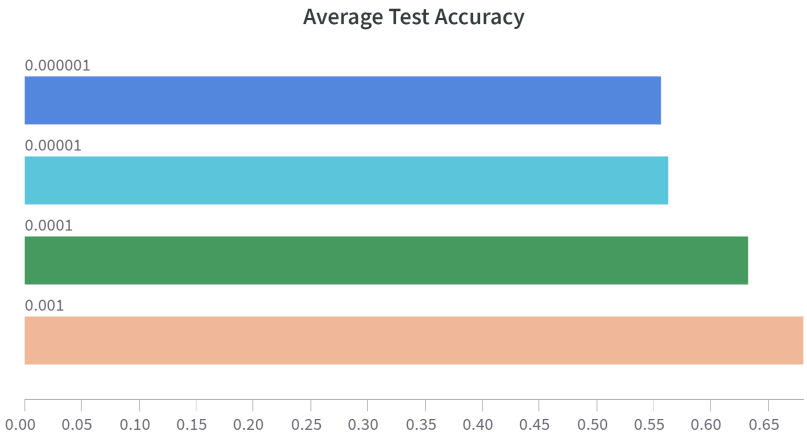


Figure 5: The effect of the hyper-parameter δ on the test accuracy.

In this plot we can see that a higher δ is correlated with higher accuracy, as we would expect. The δ is the probability of information being leaked, and when the δ is tighter the privacy loss is smaller.

- Effect of C (fixing ϵ =12, δ =0.000001):

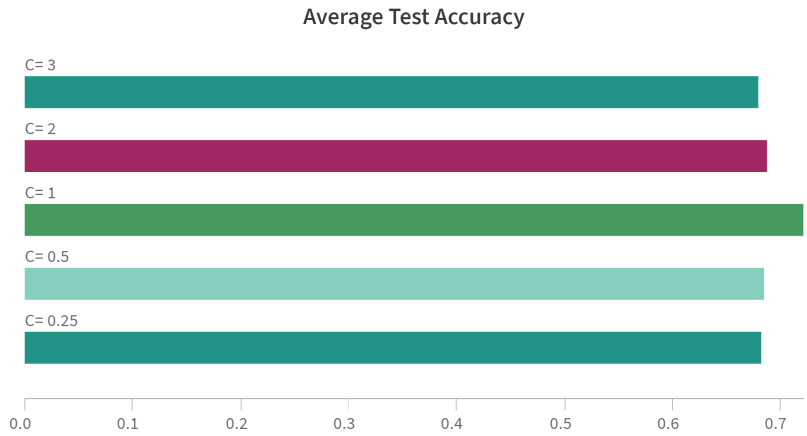


Figure 6: The effect of the hyper-parameter C on the test accuracy.

It seems like the 'max grad norm' has the least affect on the accuracy - the

test accuracies do not have big scale-differences, and there is no monotone pattern. It seems that C=1 is the best value from all the experiments, a classic norm. For this model and data, it seems there is not a clear candidate for a optimal C, and the size of the clipping is not the most significant thing in the DP-SGD mechanism in this case.

10.3.2 Privacy hyper-parameters in ViT

I also wanted to explore the effect different privacy parameter have on the ViT model, in order the later on compare those effects and result to the CNN model. I used the same values for each of the hyper parameter to explore as the CNN's, run all the runs with all the different combination, and created a similar plot as before:

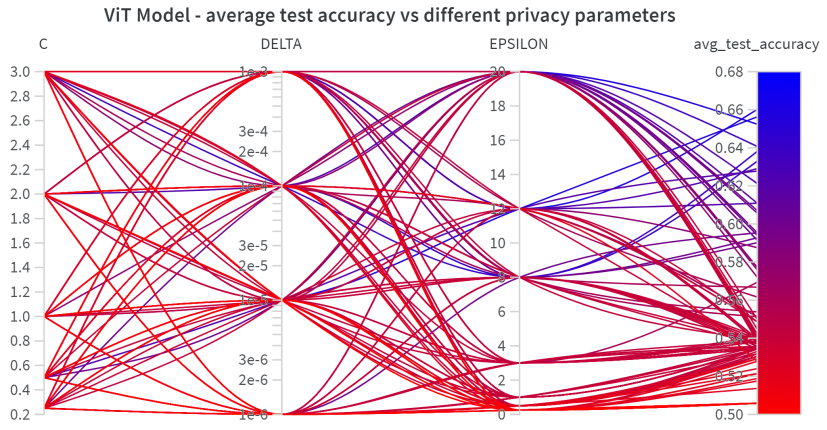


Figure 7: The effect each of the privacy hyper parameters has on the test accuracy using the ViT model. In the plot each line is a run with different combination of the hyper parameter. The bluer the run, the higher the accuracy. The redder the run, the lower the accuracy.

One can immediately see that the runs of the ViT model are a lot more redder, and there are fewer blue runs. It seems that forcing a privacy budget disturbs the ViT model more that the CNN model. I will discuss this in the 'Discussion' section. It can also be infer from this plot that the epsilon is probably the most important privacy hyper-parameter, and has the most effect on the accuracy.

- Effect of ϵ (fixing $\delta = 0.000001$, C = 1):

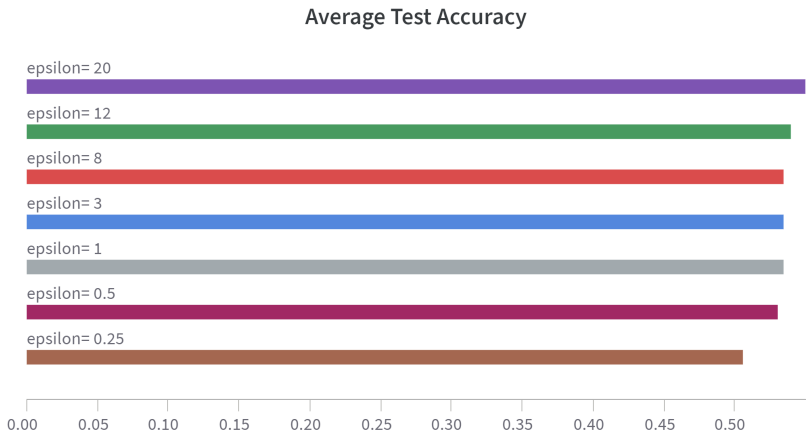


Figure 8: The effect of the hyper-parameter epsilon on the test accuracy.

We can see here the same effect of epsilon - the smaller the epsilon, the stronger the privacy, the lower the accuracy. We can see that the effect of epsilon is less significant in the ViT's case, the differences between the accuracies is smaller, and the accuracies themselves are lower the the CNN's.

- Effect of δ (fixing $\epsilon = 20$, C $= 0.25$):

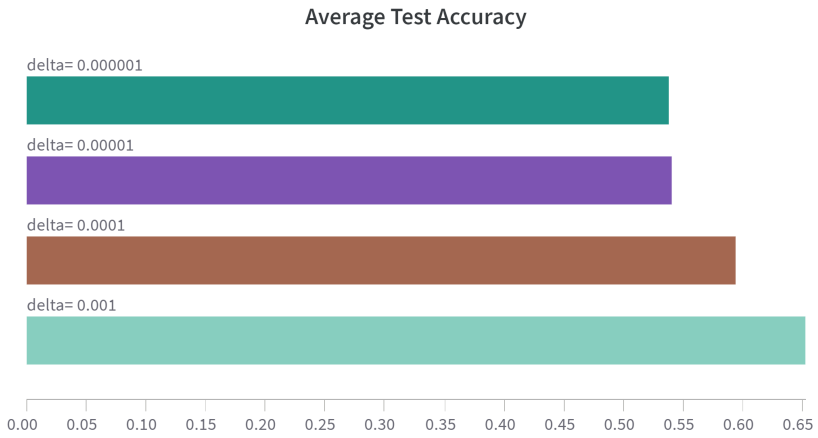


Figure 9: The effect of the hyper-parameter delta on the test accuracy.

We can see here too the same effect of delta - the smaller the delta, the smaller probability of privacy leakage, the lower the accuracy.

- Effect of C (fixing $\epsilon = 20, \delta = 0.000001$)

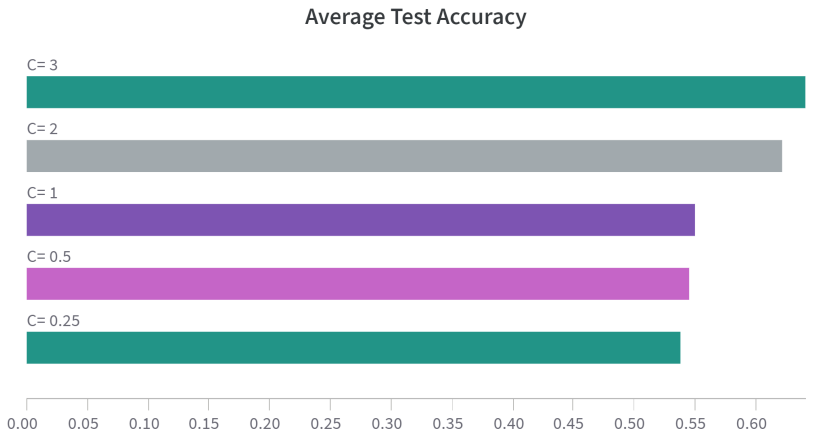


Figure 10: The effect of the hyper-parameter C on the test accuracy.

In the ViT's case, we can see the 'max grad norm' is a lot more significant

than the CNN's. We can see that the more we clip the gradient's norm, the lower the accuracy. It seems that the gradients calculated in the ViT's model on this x-ray data hold meaningful information, and when clipping them this information loss affects the accuracy in a significant way.

10.4 Final Results

	Epsilon	**Delta**	**C**	**Test Accuracy**
CNN (no DP)	-	-	-	0.801
CNN low DP	20	0.001	2	0.761
CNN mid DP	8	0.00001	1	0.678
CNN high DP	0.25	0.000001	0.25	0.505
ViT (no DP)	-	-	-	0.748
ViT low DP	12	0.001	1	0.66
ViT mid DP	8	0.0001	0.25	0.567
ViT high DP	0.25	0.000001	0.25	0.505

Table 7: Final results from chosen different experiments, to be discussed in the next section. 'low DP' means less privacy, 'high DP' means more privacy.

	Epsilon	**Delta**	**C**	**Test Accuracy**
CNN + DP	3	0.000001	0.5	0.66
ViT + DP	12	0.001	1	0.66
CNN + DP	1	0.000001	2	0.56
ViT + DP	12	0.001	2	0.56
CNN + DP	1	0.00001	1	0.518
ViT + DP	0.25	0.001	0.5	0.518

Table 8: Chosen final results, grouped and compared by accuracy. The question explored is "To achieve the same accuracy, what is the difference between the budgets we need to give to the 2 different models?"

11 Discussion

11.1 Discussion: Models without DP

First, the baseline (without DP) results for the CNN (accuracy 0.801) are a little bit better than the ViT's (accuracy 0.748). This could be explained by that the CNN model is good at identifying local spatial features in the images, while the ViT - that is transformer based - trying to "look at the big picture" and consider the full context of the image. ViT treat the image as a sequence of

patches and rely on learning global patterns. It seems that for disease classification in the X-ray data, it works better to identify specific patterns and textures than look at global patterns. The CNN model is more suited for medical images like X-rays, where the edges and little details are important.

11.2 Discussion: Adding privacy

Before we compare the 2 models with the privacy mechanism, we can note that for both of then we can notice a clear (and expected) pattern - as we force more privacy (smaller budget) the accuracy of the model will be lower. This was expected, and make a lot of sense. This can be seen the figures 4, 5,8 and 9.

For example, we can see that the CNN model got a accuracy of 0.717 with $\epsilon = 20$, and accuracy of 0.505 with $\epsilon = 0.25$ (when δ and C are fixed).

Another example - we can see that the ViT model got a accuracy of 0.55 with $\epsilon = 20$, and accuracy of 0.505 with $\epsilon = 0.25$ (when δ and C are fixed).

11.3 Discussion: Comparison between the models with DP

11.3.1 How much the DP affects the performances of the models?

Now, lets compare between the 2 models after adding DP. After adding the DP mechanism (forcing a privacy budget), and run experiment of many sets of privacy hyper-parameters, it seems that the CNN model had a lot more runs with high accuracy than the ViT model. This can be seen in figures 3 and 7 - in fig 7 we can see there a more red runs than in fig 3, and the highest accuracy of the CNN model is 0.774, compared to the highest ViT accuracy - 0.66.

We can see that forcing a privacy budget "hurts" the ViT model more that the CNN model - even when looking at the highest accuracies, the differences (in accuracy) of adding and not adding a DP are:

$$CNN : 0.801 - 0.774 = 0.027$$

$$ViT : 0.748 - 0.66 = 0.088$$

So the performance of the ViT model suffered 3 times the performance of the CNN model. There are several reason for this difference:

1. Sensitivity to Noise: ViT relies on the attention mechanism, which is more sensitive to the accuracy of gradient updates. The meaningful features in chest X-ray data are mostly local, so even small corruptions to the gradient updates (caused by the Gaussian noise in DP-SGD) can disproportionately affect the model's ability to learn the correct global relationships between image patches. The CNN model is more robust to noise because it process images locally, focusing on neighboring pixel information. In chest X-rays, where much of the diagnostic information is localized, the CNN can still detect useful features despite the added noise.

2. Gradient Clipping: Chest X-rays are gray-scale images with high spatial redundancy, meaning neighboring pixels often contain similar information. CNNs is good at extracting local features like edges and textures, making them more robust to gradient clipping. When the gradients are clipped, CNNs can still retain sufficient information about the local spatial structure of the X-rays. ViT rely on patch-wise processing and global attention, treating the chest X-ray as a sequence of image patches.

   Gradient clipping can limit the ability of the ViT to understand those connections, leading to lower accuracy. This also explains the results we saw in figure 10 vs figure 6 - the gradient clipping had more impact on the ViT model than the CNN model.

11.3.2 Same Performance, different budgets

I thought it was interesting to also look at the difference between the budgets we need to give to each model in order to get the same accuracy. This comparison can be seen in table 8.

We can see that there is a significant difference of budgets in most of the cases. To achieve accuracy of 0.66, the CNN model can be given a small privacy loss budget of ($\epsilon = 3$, $\delta = 0.000001$), while the ViT model need a bigger budget, leading to significantly bigger privacy loss - ($\epsilon = 12$, $\delta = 0.001$). We can see similar result with accuracy of 0.56. It is interesting to note that as the accuracy lowers, the difference of needed budgets seems to become smaller - to achieve accuracy 0.518, the different needed epsilons are closer (1 for the CNN and 0.25 for the ViT), but there is still a difference in the privacy loss in most cases.

12 Conclusion

All in all, we can conclude that those experiments were very interesting. Using the chest X-ray data, I compared between the more-commonly-used CNN model to the transformer-based ViT model, with and without differential privacy, and with different privacy budgets. I compared the accuracies achieved by each model in each experiment, and compared between the different budgets needed for achieving the same result. We can see that the CNN model was better both with and without the DP, because of its architecture's nature of looking for local patterns, oppose to the ViT, that tends to look for global features. By comparing the ViT's performance to the CNN'S, we got some idea on the limitation that adding a differential privacy adds to the ViT model. Adding the privacy mechanism hurts ViT more than CNN. I think this kind of comparisons is useful, because it can help researchers and developers decide what kind of model they want to use, according to what strength of privacy they want, and according to the task, domain and data they want.

13 Bibliography

1. Chest X-ray Dataset: https://www.kaggle.com/datasets/jtiptj/chest-xray-pneumoniacovid19tub
data

2. Male vs female chest x-ray (from radiology key): https://radiologykey.
com/chest-11/

3. Age from chest x-ray (from radiology masterclass): https://www.radiologymasterclass.
co.uk/gallery/chest/quality/chest-x-ray-age

4. "Deep Learning with Differential Privacy " paper: https://arxiv.org/
pdf/1607.00133

5. CNN paper - "An Introduction to Convolutional Neural Networks": https:
//arxiv.org/pdf/1511.08458

6. "Visualizing and Understanding Convolutional Networks" by Matthew D.
Zeiler and Rob Fergus (2013)": https://link.springer.com/chapter/
10.1007/978-3-319-10590-1_53

7. ViT paper - "An Image is Worth 16x16 Words: Transformers for Image
Recognition at Scale": https://arxiv.org/abs/2010.11929

8. Transformer paper - "Attention Is All You Need": https://arxiv.org/
abs/1706.03762

9. ViT implementation: https://github.com/tintn/vision-transformer-from-scratch/
tree/main

10. SOTA paper 1: https://www.frontiersin.org/journals/medicine/
articles/10.3389/fmed.2024.1409314/full

11. SOTA paper 2: https://arxiv.org/pdf/2304.11529