Chapter 5

Secure Deep Feature Classification using Convolutional Neural Network

The previous chapter discussed a deep learning-based feature extractor for breast cancer histopathology images. However, data privacy and security of the predictive model is a major concern when sharing sensitive healthcare data over the cloud. Therefore, this chapter focused on the privacy and confidentiality of medical data, as well as model security for a cloud-based approach.

5.1 Introduction

The advancements in the Internet of Things (IoT) and cloud services have enabled the availability of smart e-healthcare services in a distant and distributed environment. The integration of IoT and powerful infrastructure like the cloud into the health sector has provided new opportunities for designing new frameworks for different applications, with a view to limited storage space and fast computing resources. However, this also raised major privacy and efficiency concerns that need to be addressed. While sharing

clinical data across the cloud that often consists of sensitive patient-related information, privacy is a major challenge. Adequate protection of patients' privacy helps to increase public trust in medical research. Privacy concerns are related specifically to confidential input data during training or inference, as well as to the sharing of a trained model with other people. Security threats to AI systems may include influence attacks, security violations, attack specificity, and adversarial attacks [133, 134]. Additionally, deep learning-based models are complex, and in a cloud-based approach, efficient data processing in such models is complicated. The discovery that deep learning models are not secure hinders their practical use in safety-critical applications such as predictive healthcare. The patients expect the healthcare agencies to follow the guidelines to safeguard the confidentiality of their data and prevent its malicious use by unauthorized agencies. Thus, in the modern era of the health-IoTs and Cloud-based clinical diagnostic model, security and privacy of the data and of the prediction model without compromising its accuracy has emerged as an open research area. Recent research also validates this point by proposing private machine learning or privacy-preserving deep learning models [139].

To address these challenges, we propose an efficient and secure cancer diagnostic framework for histopathological image classification by utilizing both differential privacy and secure multi-party computation. For efficient computation, instead of performing the whole operation on the cloud, we decouple the layers into two modules: one for feature extraction using the VGGNet-16 module at the user side and the remaining layers for private prediction over the cloud. The efficacy of the framework is validated on two datasets comprising of histopathological images of the CMT and HBC. The application of differential privacy to the proposed model makes the model secure and capable of preserving the privacy of sensitive data from any adversary without significantly compromising the model accuracy. Extensive experiments show that the proposed model efficiently achieves the trade-off between privacy and model performance.

5.2 Motivation and Significant Contribution

Automated histopathological image analysis for cancer diagnosis is one of the most challenging fields in the healthcare domain due to complex structures in the histopathological images, different staining methods adopted by different laboratories, and the limited availability of labelled datasets. Nowadays IoT collaborates with health sectors and gives rise to new domains of research in e-healthcare. To promote high-quality healthcare at an affordable cost, IoT-assisted mobile, cloud-based e-health services are making great strides through the use of advanced technologies such as medi-cloud, big data in healthcare, and IoT in healthcare. These involve the distribution of health-related information either in the form of images or reports and services through telecommunication technologies. In this scenario, the patients get advice, and the doctors also get expert opinions from across the globe for diagnosis without their physical presence. The complete process involves sharing sensitive data over the internet, and hence data security is a prime concern that needs to be focused on. Although deep learning in healthcare acts as a fast diagnostic tool to support doctors in making critical decisions, it must also be secured against adversary attacks. Finally, IoTs facilitate sharing of the data from different sources over the cloud and provide an opportunity for secure, private and distributed deep learning to research communities associated with deep learning and the healthcare industry. Additionally, a secure and private learning model for histopathology image classification remains less explored. These are the challenges that motivated us to address the aforementioned issues and thereby explore the new possibilities by designing a secure deep framework.

In this work, we have introduced a novel, secure, deep feature classification framework based on VGGNet-16 by decoupling its layers to form two modules, feature extractor at user side and private classification module in a distributed environment for cancer diagnosis. In this work, we use VGGNet-16 as a feature extractor for CMT and HBC classification. Security concern is addressed by incorporating security at the

model level as well as at the feature level by adding an even stronger layer of privacy using differential privacy. The efficacy of the proposed model is tested on CMTHis and publicly available BreakHis datasets. Its performance is also validated against performance evaluation metrics like accuracy and privacy measure. The major contributions of our work can be summarized as follows:

1. This work addresses the secure multi-party differentially private cancer diagnostic framework for canine mammary histopathological image classification. To the best of our knowledge, this is the first work for CMTHis classification while preserving the privacy of data as well as prediction model over cloud services.

2. In this work, a novel framework is developed using cryptographic concepts such as secure multi-party computation, differential privacy, deep learning, and distributed approaches over the cloud for cancer diagnosis.

3. In the proposed secure, hybrid user-cloud based framework for histopathological image classification, layers of VGGNet-16 are decoupled into two parts where initial layers are used for feature extraction at the user side from original and sensitive tumour histopathology images. Remaining layers are deployed over the cloud for private prediction.

4. In the proposed framework, privacy of extracted features at the user's device is enabled at the time of offloading to the cloud for prediction. Original datasets or raw images are also secure because instead of sharing original images, extracted features are shared over the cloud.

5. The proposed framework is capable of reducing the risk of leakage of sensitive medical data and model under consideration by using tensor flow privacy to facilitate training with differential privacy. Results show that the proposal maintains the trade-off between privacy and performance significantly.

5.3 Theoretical Background

5.3.1 Privacy preserving in deep learning

Deep learning is an efficient modelling technique in the health domain for automated features extraction of datasets under consideration. Its performance is directly influenced by the vast amount of training data, which is usually taken from the different users that are stored on the cloud server and contain sensitive information. Nowadays, IoTs facilitate deep learning as a service where users receive the prediction detail after sending queries to the cloud server.

In the case of cancer histopathology, these queries contain sensitive information in the form of images or patient's records that are processed on a remote deep learning server. The privacy of sensitive information is a major concern during the training as well as prediction phase. One way for the user to tackle this is to download the deep learning model and execute it on their own platform. However, it would not be feasible in a real-world setting because the model learns by processing a huge amount of training samples.

Therefore, to get the maximum advantage of deep learning in this new era of cloud-based healthcare, an efficient approach for private training and prediction of deep learning is important. The process is termed as privacy-preserving deep learning and has three major requirements: firstly, user-sensitive data should not be disclosed to the central server during the training phase of the deep learning model. Secondly, during prediction, the individual's queries should not be revealed to the server, and finally, the server's deep learning model should not be disclosed to the user. Recent works addressed this issue of privacy and presented privacy preserved deep learning through the implementation of complex privacy-preserving constructs such as federated learning, Secure multi-party computation (SMPC), Differential Privacy (DP) while exposing the end-user to a familiar deep learning application program interface [181, 182, 183, 184].

5.3.2 Secure multiparty computation

Secure multi-party computation, also known as secure computation, is a cryptographic sub-field that allows a set of parties to jointly perform a distributed computation while ensuring accuracy, the privacy of the parties' input, and more. This framework differs from traditional cryptography because, in this case, the information is to be protected from other participants rather than from an adversary outside the system. SMPC is based on the fundamental concept of secret sharing presented in [185]. This technique demonstrates how single data D can be divided into n parts and distributed secretly among the participants so that any k data parts can easily reconstruct D but even users having complete detail of any $k-1$ data parts are not able to reveal any information about D. It implies that the result of every participant, which is computed on their own data, can be grouped together to get the required outcome without revealing the secret input.

This technique also allows the construction of a reliable key management system for a cryptographic model that can work securely. In 1982 Yao [186] introduced multi-party computation and contributed to a new and thriving research domain that offers secure solutions for a wide range of applications. Generally, MPC is defined by four constructs, i.e. functionality, security type, adversarial model, and communication model, which answer the following questions: i) what is needed to be computed? ii) how strong should be the required protection? iii) what do we want to protect against? and iv) in which setting will it be done? Here, three types of security – statistical, computational and perfect can be implemented. Adversarial model can be categorized on the basis of adversarial behaviour, number of corrupted parties, adversarial power, and adversarial corruption, which may be static, adaptive or mobile. The chosen communication model can be either point to point or broadcast. SMPC has a big impact on the privacy and security of data due to the combined approach of encryption, distribution, and distributed computing.

Further, SMPC is categorized as two-party computation and multi-party computation. SMPC mostly follows two approaches. The first approach is usually based on secret sharing and operating on the computed function, which is represented as an arithmetic circuit. It is typically applicable when there is an honest consensus among the participants, which is possible only when more than two parties participate in the protocol. In the other approach, a binary circuit is used to represent the function, and it is executed in a constant number of communication rounds. These two approaches are discussed in the following sections.

5.3.2.1 Two- party computation (2PC)

Secure two-party computation (S2PC) problem and its solution were formally introduced in [145]. Its generalized and extended version for multi-party computation (MPC) problem was presented in [146, 187, 188]. S2PC is also known as Secure Function Evaluation (SFE). It allows two non-colluding parties P_1 and P_2 to evaluate an arbitrary function g on their respective private inputs (x), (y) without revealing anything except the result $g(x, y)$. Yao's basic protocol based on a garbled circuit is secure against partially honest adversaries. Further, active S2PC are presented in [189, 190, 191], and in another work, [192] cut and-choose oblivious transfer protocol is used for S2PC and also provides an exact and concrete analysis of the proposed method. Besides these protocols, recent work [149] also present a new efficient protocol to design privacy preserved machine learning, SecureML, by following the concept of the two server model.

5.3.2.2 Multiparty computation

Secret sharing is the basic building block of multi-party computation (MPC). The two most commonly used methods are Shamir's secret sharing [185] and additive secret sharing. MPC protocol is widely used by recent MPC based machine learning and deep learning framework. MPC allows a set of n mutually mistrusting parties to $(P_1, P_2, \cdots, P_n)$ jointly compute the function g of their input data $D_i : i \in \{1, 2, \cdots, n\}$

where each P_i holds its private data D_i in such a way that correctness and security are ensured. In other words, all n parties collaboratively calculate a function $g(D_1, D_2, \cdots, D_n)$ such that at the end of interaction, party P_i which holds D_i gets only i^{th} component of outcome of $g(D_1, D_2, \cdots, D_n)$ and nothing else.

By SMPC, we can effectively minimize the risk of vulnerability and prevent others from seeing individual data as well as the parameters of the trained model. We can keep the remaining shares required to retrieve the original secret as input to the joint function and secure them until needed. An important aspect of MPC is that security should be retained even when an adversary is present and controls a few of the participants and, in the case of malicious security, takes full control of the corrupted parties, inducing their actions in arbitrary ways. MPC consists of several desirable features which may negatively impact the performance, which suggests that the selection of an MPC protocol may be considered as an important trade-off. Among these protocols, an important parameter t_h is the number of the corrupted participants out of total number of n participants that can be tolerated within the range of $t_h < \frac{n}{3}$, $t_h < \frac{n}{2}$ for honest majority and $t_h < n$ for unbounded number of corrupted participants.

There are several works that utilize MPC with secret sharing, and one such popular work is SPDZ [193], which follows the $'share-compute-reveal'$ paradigm with additive secret shares and is secure against a dishonest majority. Other existing works reported in [194, 195] are the independent framework that focuses on MPC protocol. However, they do not help in designing integrated private machine learning/deep learning. Few works are presented in [152, 147] where authors developed an MPC framework for efficient three-party protocols tailored for state-of-the-art neural networks. The first robust framework for privacy-preserving machine learning, FLASH, considered a four-party honest majority setting [196]. Recently, TF-encrypted [197] which provides SMPC directly in TensorFlow, CRYPTFLOW [198] and CryptoSPN [199] that support the new secure deep learning framework were developed.

5.3.3 Differential privacy

Differential privacy (DP) is a rigorous mathematical framework that refers to the mechanism of adding random noise to perturb the released statistical result computed from the sensitive dataset in order to protect the private data. It prevents the adversary with ample prior knowledge from obtaining any accurate information about the individuals in the dataset with high confidence. It has been intensively studied from the theoretical perspective [200, 201]. It was introduced formally in [202], and recently DP has become a standardized measure to obstruct the leakage of sensitive data during training. The majority of the works in privacy-preserving machine learning and deep learning use some form of DP. Additionally, the DP mechanism comprises of the Laplace mechanism [203], the exponential mechanism [204], and the Gaussian mechanism [205]. It is mostly defined in terms of the application-oriented concept of neighbour datasets. It is quite useful for healthcare applications due to its different characteristics like group privacy, composability, and robustness to auxiliary information. To avoid privacy breaches several research works for DP have been proposed such as differentially private stochastic gradient descent algorithm (DP-SGD)[206], private aggregation of teacher ensembles (PATE) [207], moment accountant[208], hyperparameter selection [209], exponential noise differential privacy mechanism and differentially private alternating direction method of multipliers (P-ADMM)[210]. Besides these, collaborative deep learning with DP and deep private autoencoder (dPA)[211] were also proposed and analyzed.

For defining DP we consider the database of histopathological images as D which represents whole set of images with total number of patients N_T and each patient database D_i which consists of n_i tissue images of i^{th} patient such that $D_i = (X_p, y_p)_{p=1}^{n_i}$ and $D = \bigcup_{i=1}^{N_T}(D_i)$. Further, two databases are said to be adjacent at the image level if they differ in only a single entry, i.e., differ in single image-label pair. User adjacent datasets, which are in our case patient adjacent databases, are defined as mentioned in

[184]. The overall concepts can be understood easily from the following definitions.

Definition 5.1 ***Patient Adjacent Databases*** *: D_1 and D_2 which consists of training datasets associated with a patient respectively, are said to be adjacent if D_2 can be created by adding or removing all images of a single patient from D_1.*

Definition 5.2 ***Differential Privacy****: A randomized algorithm $K : D \rightarrow R$ with domain D consist all possible training samples and range R contain all possible trained models, satisfies (ϵ, δ)- differential privacy, iff for any two neighbouring (adjacent) databases $D_1, D_2 \subseteq D$ that differ in only a single entry and for any subset of outputs $O \subseteq R$ it holds that:*

$$Pr[K(D_1) \in O] \leq e^{\epsilon} Pr[K(D_2) \in O] + \delta \tag{5.1}$$

Here δ accounts for the probability that plain ϵ- differential privacy is violated, and the lower value of ϵ enforces a higher privacy guarantee of algorithm K. If $\delta = 0$, K is said to satisfy ϵ- differential privacy. To achieve DP, the real-valued function $g : D \rightarrow R$ would be perturbed through additive noise. This noise depends on the sensitivity of the output.

Definition 5.3 ***Sensitivity*** *: The sensitivity of any real valued function $g(\cdot)$ is the measure of the maximum change of the output due to the inclusion of single instance, Formally it is defined as:*

$$s_f = \max_{D_1, D_2} \|g(D_1) - g(D_2)\|_2 \tag{5.2}$$

where D_1 and D_2 are any adjacent databases and $\|\cdot\|_2$ denotes L_2 norm.

Definition 5.4 ***Gaussian Mechanism*** *: A gaussian mechanism is a common approach of satisfying DP by introducing Gaussian noise calibrated to the sensitivity (s_f)*

of the real-valued function. It is defined as follows:

$$K(D) = g(D) + N(0, \sigma^2 s_f^2) \tag{5.3}$$

where, $N(0, \sigma^2 s_f^2)$ *represents the Gaussian distribution with* 0 *mean and* $\sigma^2 s_f^2$ *variance.It achieves* (ϵ, δ)*-DP with* $\epsilon, \delta \in (0.1)$ *and* $\sigma \geq \frac{\sqrt{2\ln(\frac{1.25}{\delta})s_f}}{\epsilon}$.

5.4 Proposed Methodology

5.4.1 VGGNet-16 as a feature extractor

Feature extraction and feature selection are crucial tasks that highly influence the performance of the machine learning-based framework. Generally, machine learning-based framework utilizes handcrafted features. However, to extract different handcrafted features, domain expertise is required, while CNN automates this task with the help of convolutional layers along with filters of different sizes. Among these hierarchical layers, lower layers learn low-level features such as edges and corners, while features like shape, colour etc., are learned by middle layers. High-level features like the presence of an object in an image are learned by higher layers. Instead of image classification, we have used VGGNet-16 for feature extraction in the proposed method because of its uniform and superior architecture with fewer number of parameters and computations. It is also able to incorporate more nonlinearity. It is used as a feature extractor by taking the available activation map before the fully connected layer because higher layers provide less generic characteristics with reduced performance. Also, VGGNet-16 is the most preferred choice for feature extractor and is used in a variety of applications. VGGNet-16 has a uniform architecture that makes it more appealing, understandable, and explainable, which motivates us to choose VGGNet-16 over other deep learning models. It has been observed from the experiments, a ResNet has a faster inference time, but on histopathology data (BreakHis), the VGGNet-16 model is more stable

than ResNet50. As a result, we chose it as a feature extractor with a modified feature consideration strategy. Furthermore, the goal of this study is to create a low-cost and efficient solution for histopathology cancer classification that pathologists can use in a low-cost clinical setting and that is also easily understandable; for this, VGGNet-16 demonstrated its suitability. These models, such as VGG and ResNets, were created to demonstrate the potential of deeper architecture for classification, with a focus on convolutional operations and the fusion of spatial and channel-wise information. A few other models, such as SENet, introduced a block that dynamically calibrates the channel-wise feature interdependency. We were able to achieve higher accuracy on 40 x with the proposed model than with SENet, achieving the goal of providing an efficient solution for a low-cost clinical setting. Thus, these are the promising reasons for using the VGGNet-16 model as the base model, which was then modified for feature extraction and classification. We have applied the same VGGNet-16 framework as explained in the previous chapter. However, for a better understanding of the readers, we again providing its brief overview again.

Fully connected layers from the network are removed, and instead of max-pooling, global average pooling (GAP) is used after each extraction layer. This helps in limiting the features numbers, and the GAP layer acts as a regularizer that minimizes overfitting by incorporating dimensionality reduction. These extracted features, instead of raw images, are then uploaded to the cloud for further processing as shown in Figure 5.1. Hence, without sharing original and sensitive medical images, users can use the prediction model deployed over the cloud by supplying features only. In a cloud-based approach, the service provider employs a feature extractor module to the user at the client-side and determines how the feature extraction is done. Using this module for feature extraction, the user extracts the features from data on the client-side and uploads it to the cloud. Thus instead of performing the whole computation on the cloud, this framework enables IoT devices to execute the initial layers of VGGNet-16 for fea-

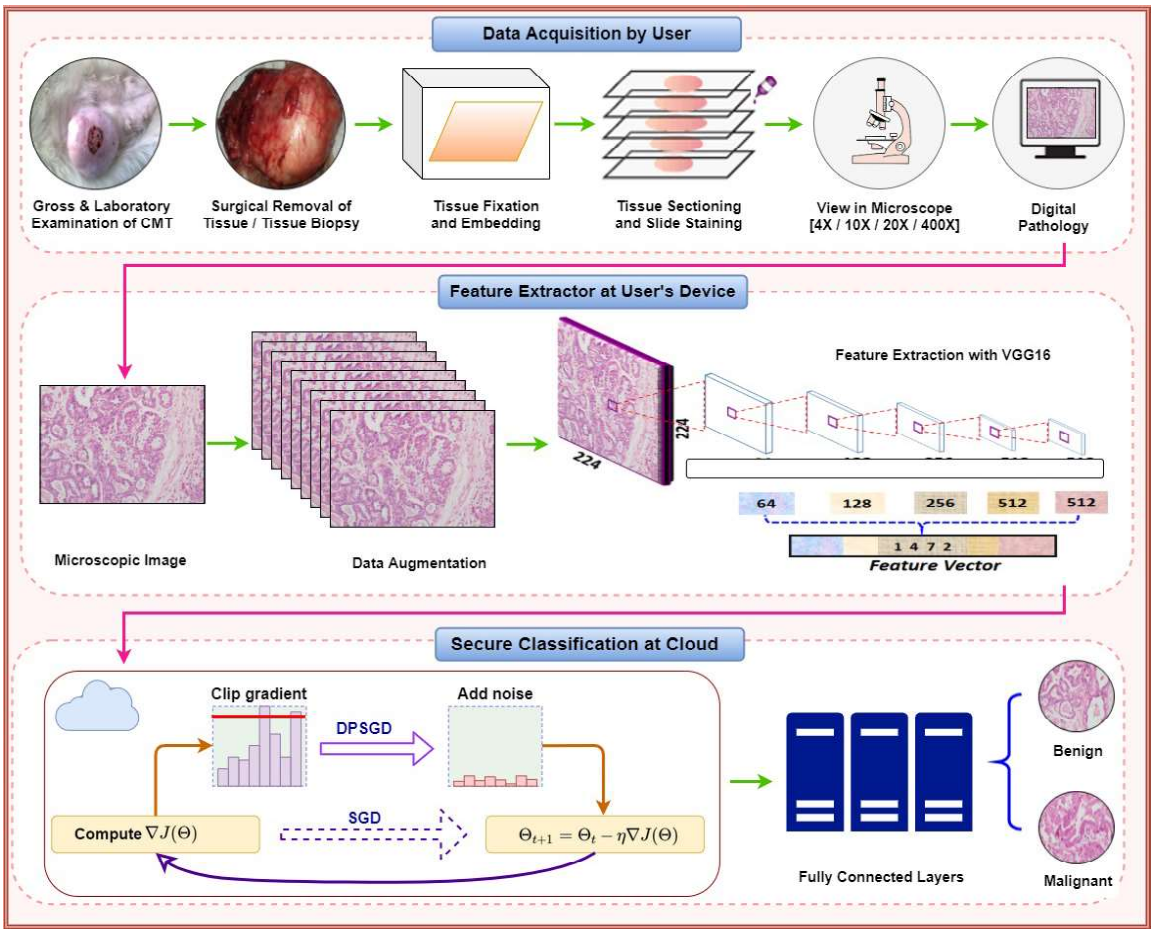


Figure 5.1: Overall Framework

ture extraction, and the obtained features are sent securely to the MPC engine for classification. Considering original dataset at the user side consists of N samples of histopathological images is denoted by $\{(X_i, y_i)\}_{i=1}^{N}$ where X_i is the i^{th} sample image and y_i is corresponding class labels. After applying $FE - VGGNet_{16}$, feature vectors $(K_i, y_i)_{i=1}^{N}$ are obtained where $K_i \in R^n$ is n dimensional feature vector of i^{th} sample image at user side.

5.4.2 Privacy preserved model training with differential privacy

Noise embedding is a traditional method for preserving privacy [202]. Adding noise to the feature vectors is the one possibility that enhances the uncertainty in the sensitive

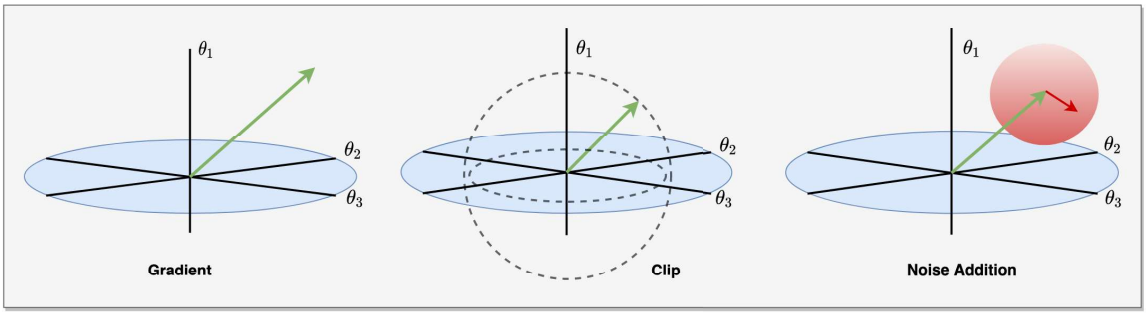


Figure 5.2: Representation of differentially private stochastic gradient descent (DPSGD) algorithm.

features, and thus privacy will be better preserved. This approach is used because when a model is trained on data, encrypted or raw, it might memorize some sensitive information of the original dataset. Once the model is deployed, it could reveal this sensitive information while querying it or by inspecting the weights. Training the model using DP prevents it from memorizing this sensitive data with a specific privacy guarantee. Any model trained with DP performs similarly, whether or not any sample (individual's private information) in the training data is included. Typically, DP works by adding noise to the dataset. More uncertainty is achieved by increasing the variance of the added noise to the features. That means more the noise, more the anonymity of the features. However, high variance noise could also reduce the prediction accuracy. Hence balance between accuracy and noise is controlled using a parameter called epsilon (ϵ), which is also the specific privacy guarantee. Smaller the ϵ, the better is the privacy, but worse is the accuracy. The other parameter is Delta (δ), which limits the probability that the privacy guarantee does not hold.

Therefore, we have implemented and applied DP as described in [206] to train the fully connected layers. This provides training of deep neural networks within a small privacy budget. To achieve this, we trained the model on features by applying differential private stochastic gradient descent optimizer (DPSGD) presented in Algorithm 5.1 [206]. To protect privacy, we also employed Gaussian Mechanism for noisy injection in the features as described in (7.3). Basic steps that were followed in DPSGD are illus-

Algorithm 5.1: Stochastic Gradient Descent with Differential Privacy

Input: Training Features: $(K_i, y_i)_{i=1}^{N}$, Empirical loss: $\jmath(\theta) = \frac{1}{N}\sum_i \jmath(\theta, K_i)$, Learning rate (γ_t), Batch size (L), Noise scale (σ), Gradient norm bound (C_u), Sampling probability (p) : $\frac{L}{N}$

Output: θ^T and overall privacy cost (ϵ, δ) compute using privacy accounting approach

Initialize: Model parameter θ^0 randomly

for $t \in [T]$ **do**

  Take a random sample subset L_t with sampling probability p.

  for *each sample* $i \in L_t$ **do**

    Calculate $g_t(K_i) \leftarrow \bigtriangledown_{\theta^t} \jmath(\theta^t, K_i)$ ▷ Gradient computation

  end

  $\tilde{g}_t(K_i) \leftarrow g_t(K_i)/max(1, \frac{\|g_t(K_i)\|_2}{C_u})$ ▷ Clip gradient

  $\tilde{g}_t \leftarrow \frac{1}{L}(\sum_i \tilde{g}_t(K_i) + \mathcal{N}(0, \sigma^2 C_u^2 I))$ ▷ Noise insertion

  $\theta^{t+1} \leftarrow \theta^t - \gamma_t \tilde{g}_t$

end

trated in Figure 5.2. The trained model parameters and hyperparameters with features are saved for further processing.

5.4.3 Multi party secure tumor classification with cloud services

For private prediction or encrypted machine learning, Tensor flow encrypted (TFE) is the common framework built on the tensor flow that helps to deploy a secure model. So we used TFE after training the model with normal Keras. TFE under the hood uses an encryption technique called MPC, and thus we have created three TFE servers to secure and serve the proposed framework based on multi-party protocol. For the tumour classification, we used Secure NN [152] as the underlying protocol.

Basically, we follow the idea of creating shares by splitting model weights and extracting features of data as an input. Then the share of each value is sent to different

servers. After finishing the process of model encryption and parameters sharing, the model is served to the client by setting a queue server. The best aspect of this approach is that even if the share of one server is known, nothing can be revealed about the original data or model parameters. The overall private histopathology image classification based on MPC is shown in Figure 5.3.

5.5 Experimental Results and Analysis

5.5.1 Experimental setup

The framework proposed in this study was evaluated on BreakHis and CMTHis datasets, which are described in Chapter-3. The images in the dataset are resized before being used as an input to the model. We resized the images to 224 × 224 before being used as an input to the model. The data augmentation for proposed method was done simi-

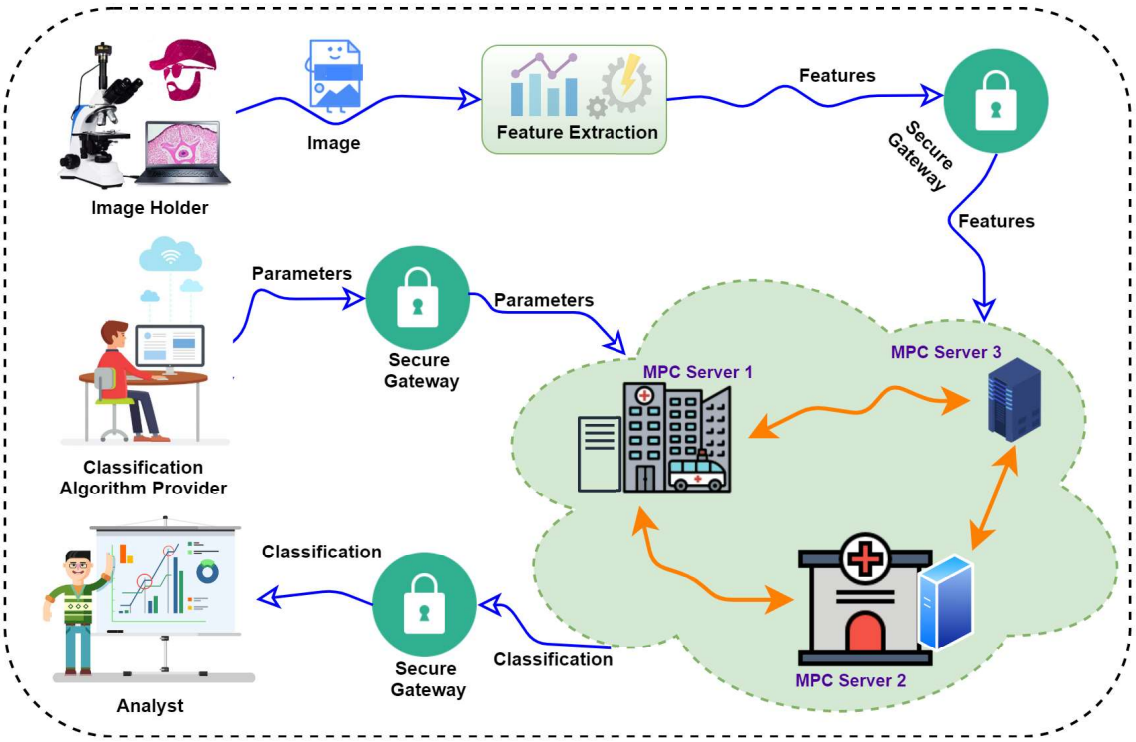


Figure 5.3: Representaion of private image classification framework.

larly as described in the preceding chapters. The results were generated on a 2.60 GHz Intel-Xeon E5-2660v3 GPU with 128 GB DDR-4 ECC RAM and the 12 GB NVIDIA Tesla K40C graphics processor.

5.5.2 Model evaluation

In this study, we have proposed a secure framework for the classification of cancer histopathological images by decoupling the layers of VGGNet-16 into two parts where initial layers are used for feature extraction and the remaining layers (fully connected) are deployed over the cloud for private prediction. Feature extraction from the external convolutional layers is followed by GAP in VGGNet-16, resulting in a single vector of 1472 features after concatenation as outlined in Section 5.4.1. These extracted features are mapped by a subset of secure, fully connected layers in the cloud to the final outputs of the network, such as the probability for each class in the classification process. In this architecture, each fully connected layer is followed by a nonlinear ReLU function, and the final fully connected layer has the same number of output nodes as the number of classes present in the considered dataset. Fully connected layers were trained for 200 epochs with a model checkpoint to save the best model in terms of validation accuracy.

The architecture of fully connected layers is shown in Figure 5.4. Also, the plot of accuracy vs epoch and loss vs epoch of the model training at the fully connected layers are shown in Figure 5.5 (a) and Figure 5.5 (b), respectively. Further, TFE was used for making private predictions.

5.5.3 Comparison of the efficacy of the proposed framework with other CNN models

The efficacy of the framework was evaluated on two datasets, namely CMTHis and BreakHis datasets, comprising of images captured at different magnifications, and the experimental protocol was applied independently to every magnification. The proposed

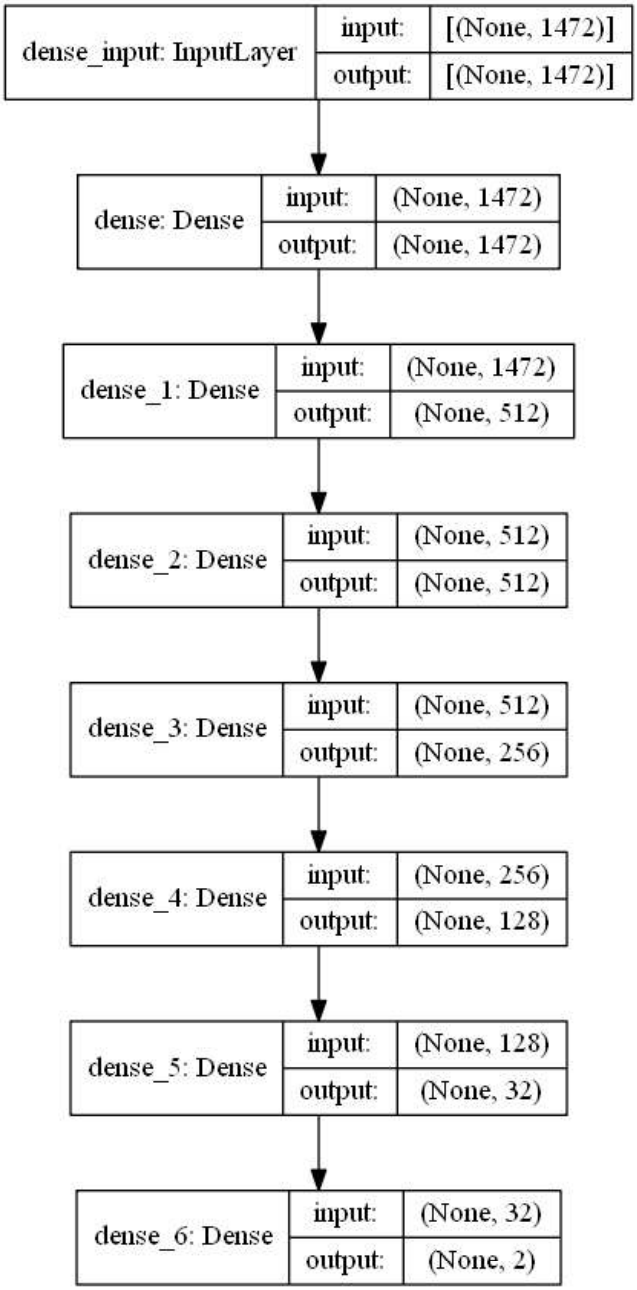
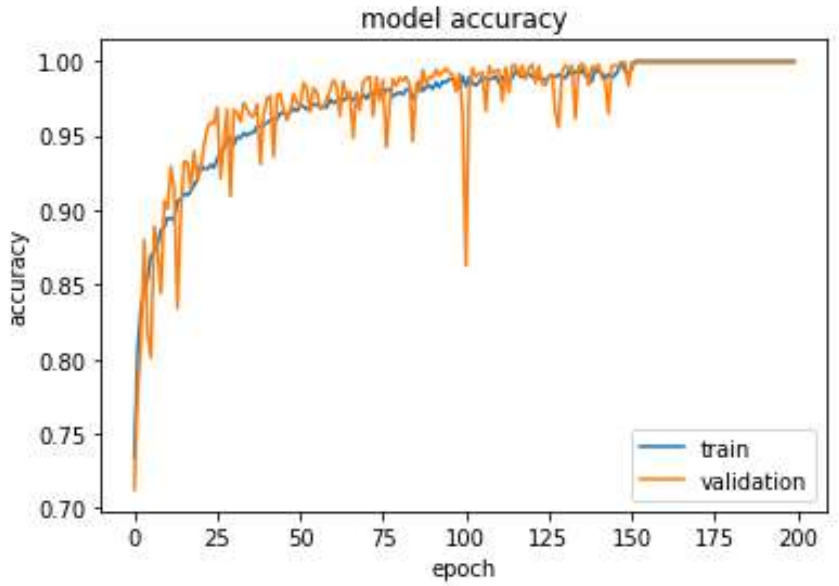
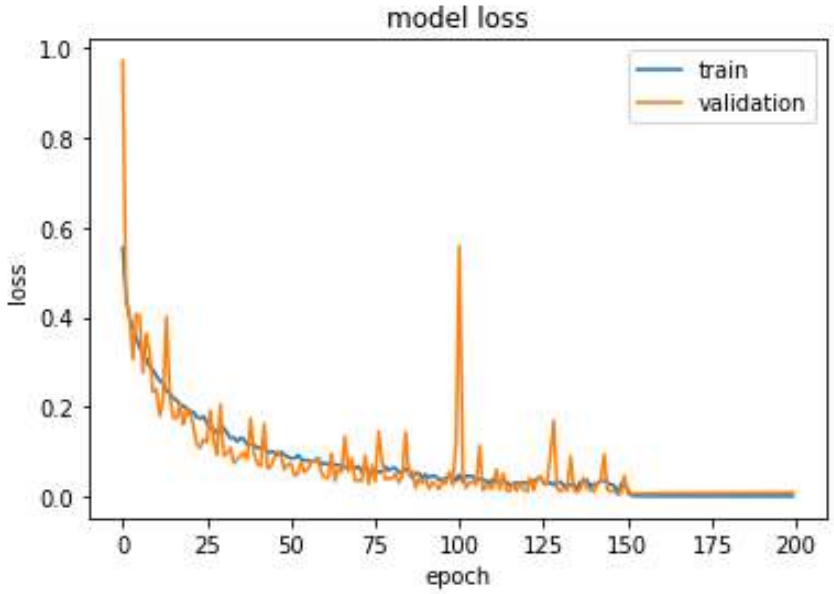


Figure 5.4: Architecture of fully connected layers in the cloud.

model resulted in state-of-the-art accuracy of 91 ± 1.8% on the BreakHis dataset at 200 X magnification and 87.1 ± 2.3% on CMTHis dataset at 40X magnification. The accuracy of the framework was affected by the magnification factor, with superior results observed at lower magnifications (40X, 100X, 200X) as compared to higher magnifica-



(a) Accuracy vs epoch curve in the training phase.



(b) Loss vs epoch curve in the training phase.

Figure 5.5: The accuracy v/s epoch and loss v/s epoch curves of proposed model on the CMTHis dataset at 40X magnification.

tion (400X). Figure 5.6 compares the classification accuracy of the proposed framework at different magnification factors for CMTHis and BreakHis datasets. The accuracy of the proposed model was also compared to other CNN architectures used by different researchers for the classification of HBC images that reported their results on the BreakHis dataset. Specifically, six architectures are included: VGGNet-16, $VGGNet_{19}$ [115], CNN [6], Fisher Vector (FV) + CNN [175], Squeeze-and-excitation networks (SEnet) [212], and Component Selective Encoding (CSE) + CNN [213]. As can be seen from Table 5.1 the proposed model achieved higher accuracy in comparison to other CNN based models. It was observed that most of the CNN architectures showed poor performance at 400X magnification. The main reason is that the 400X magnification image is more likely to have incomplete tissue structures with a smaller receptive area for collecting information from the original image, which in some situations may contribute to misclassification. The comparative analysis clearly indicated that our

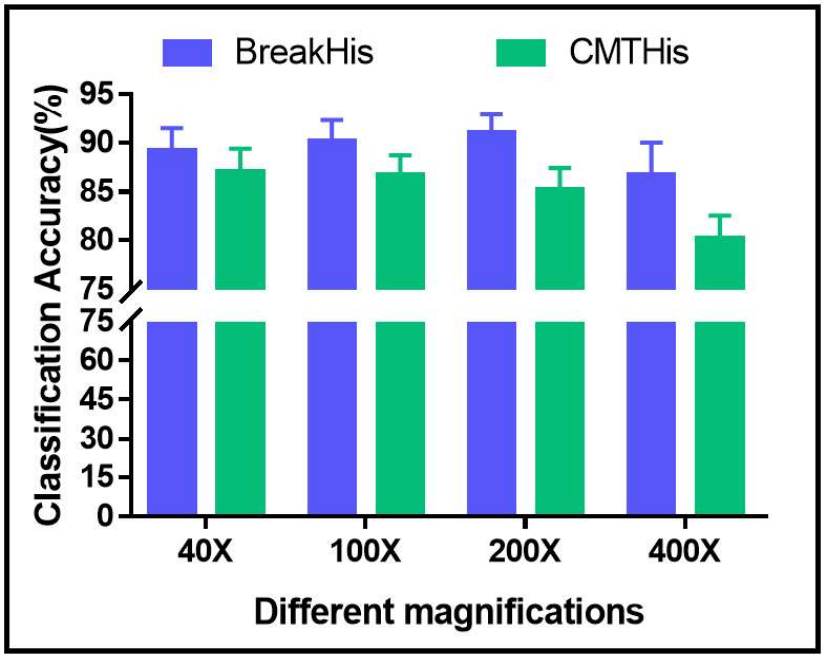


Figure 5.6: Comparison of the accuracy of the proposed framework (w/o DP) on BreakHis and CMTHis datasets.

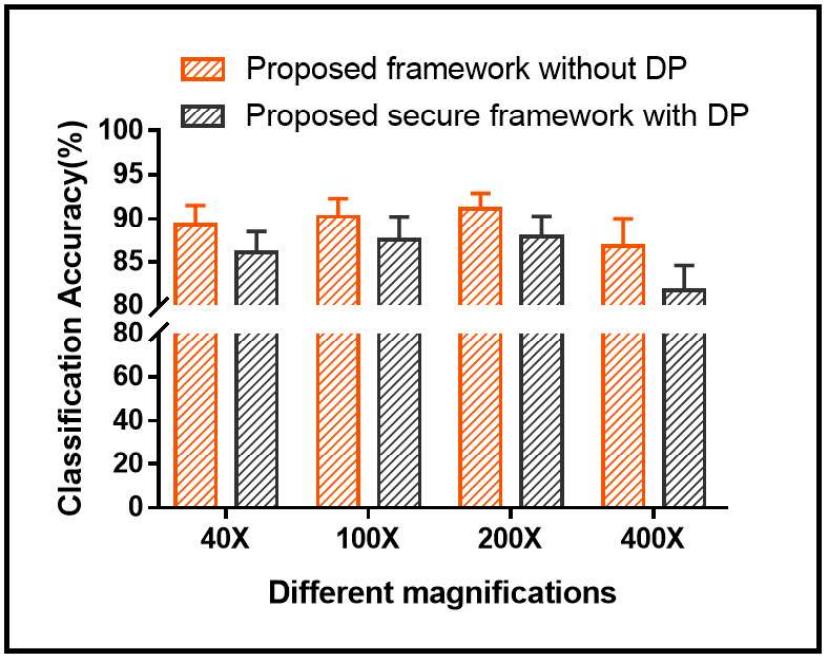
Table 5.1: Comparison of the model accuracy with other state-of-the-art methods.

Models	Magnification Factors							
	40X		100X		200X		400X	
	Mean	SD	Mean	SD	SD	SD	Mean	SD
Without DP								
VGGNet-16 [115]	72.2	2.8	79.5	2.6	81.5	2.2	87.7	2.4
VGGNet-19 [115]	80.2	1.8	80.8	2.8	81.4	2.1	79.5	3.4
CNN [6]	85.6	4.8	83.5	3.9	83.1	1.9	80.8	3.0
FV + CNN [175]	86.8	2.5	85.6	3.8	83.8	2.5	81.6	4.4
SENet [212]	86.4	2.0	91.5	1.8	89.5	2.4	92.8	2.0
CSE+ CNN [213]	87.5	1.6	88.6	3.6	85.5	2.0	85.0	4.6
Proposed	89.3	2.2	90.2	2.1	91.1	1.8	86.8	3.2
With DP								
Convolutional Block (encrypted) [214]	-	-	81.77	-	-	-	-	-
Proposed	86.10	2.4	87.50	2.7	87.90	2.4	81.70	2.9

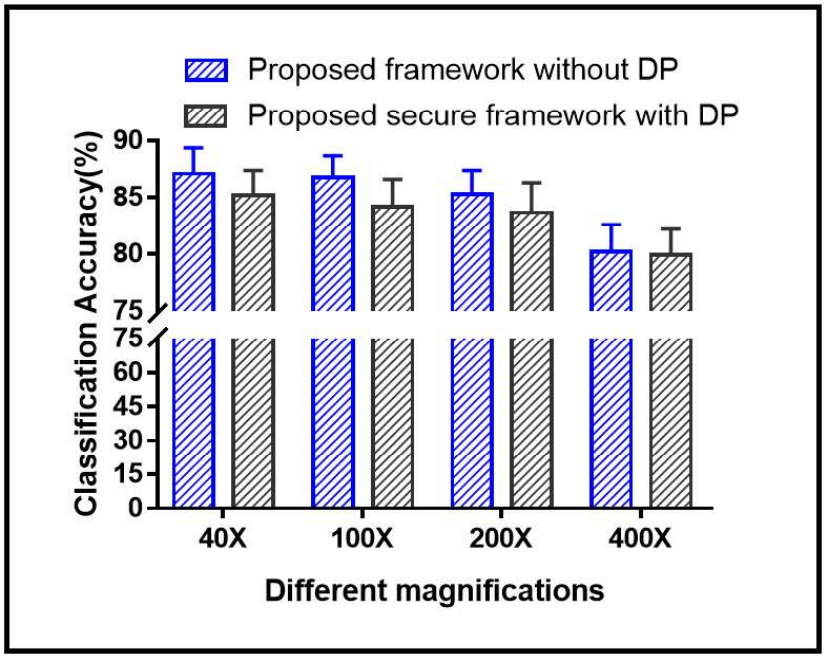
approach significantly outperforms several deep learning-based frameworks.

5.5.4 Application of differential privacy learning to the proposed model

To make the model secure, deep privacy-preserving and multi-party computation was integrated with the proposed model. This was achieved with minimal changes to the original model by replacing the original SGD optimizer with its DPSGD equivalent. In the proposed model, extracted features were sent to the cloud for prediction, and the privacy of features was enabled at the time of offloading to the cloud. Original dataset or raw images were also secure because instead of sharing original images, extracted features were shared over the cloud. The application of DP to the proposed model made the model secure as indicated by epsilon (ϵ) values 2.5-2.9 across different



(a) BreakHis dataset



(b) CMTHis dataset

Figure 5.7: Effect of magnification on the efficacy of the proposed framework with and without DP.

magnifications in CMTHis and BreakHis dataset. All these empirical results validate our theoretical findings, i.e., the model trained using DPSGD can protect sensitive data from any adversary. The results of the proposed model with and without differential privacy preserving were compared.

The test accuracy of the proposed secure framework was 87.9 ± 2.4 % for the BreakHis dataset at 200x magnification. For the CMTHis dataset, the proposed secure framework resulted in the highest test accuracy of 85.2 ± 2.2 % at 40X magnification. A training procedure was run for 200 epochs with $\sigma = 0.8$, $\delta = 1e - 5$, and a learning rate of 0.01. Figures 5.7 (a) and 5.7 (b) compare the test accuracies achieved using the proposed framework with and without application of DP learning on BreakHis and CMTHis datasets, respectively. The accuracies were slightly lower in comparison to the accuracies observed without the application of DP, though the differences were not significant. This may be due to the addition of noise in the data while training using DP. Several other studies have shown that DP application slightly reduces the accuracy of the deep neural networks [215]. Though the accuracy is slightly compromised, there is a reduced risk of the leakage of sensitive medical data as well as the model under consideration.

5.5.5 Ablation study on Activation Function

Activation functions have a very important role to play in interpretation. Ablation study is performed by considering the different activation function applied to the last fully connected layer is distinct from other layers. Depending on each task, an appropriate activation function must be selected. So, the effects of three different activation functions, namely sigmoid, softmax, and linear, were also evaluated on the classification performance of the proposed framework. The results are compiled and presented in Figure 5.8. The results indicate that the linear activation function has lower accuracy on the proposed secure framework as compared to the sigmoid and softmax activation

functions, which were found to be equally good as Sigmoid function is the special case of Softmax function where the number of classes are 2. The highest test accuracy of 85.2% (± 2.2) was achieved using the sigmoid activation function at 40X magnification of the CMTHis dataset.

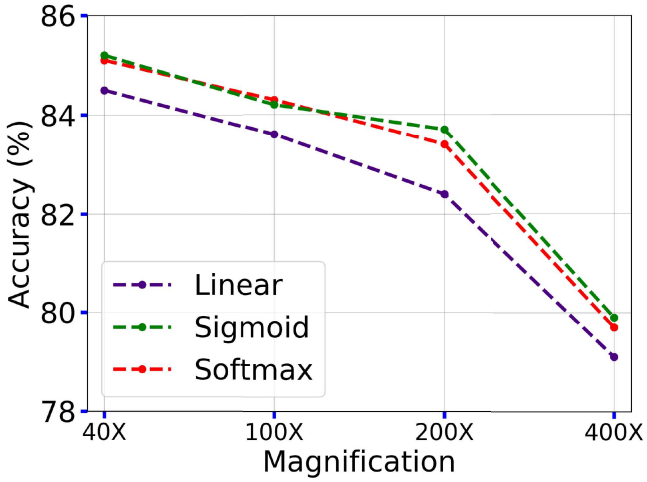


Figure 5.8: Comparison of different activation functions on the efficacy of the proposed secure model on CMTHis Dataset.

5.6 Summary

Recent developments in IoTs and cloud services have influenced medical domain and machine learning research communities, resulting in an automated e-healthcare and distributed machine learning/deep learning based framework. However, the sharing of healthcare data without violating the patient's privacy is crucial in e-healthcare services. Hence, in this work, we addressed the privacy concern of deep learning-based framework for histopathology image classification over cloud and IoT. We proposed and demonstrated a novel, secure user-cloud based deep framework VGGNet-16 with DP and SMPC for CMTHis and BreakHis classification. This framework provides a potential way to share and make a private prediction of sensitive healthcare data while maintaining the confidentiality of the patient's data and model's parameters. Experi-

mental results show that the model trained using DPSGD can preserve the privacy of sensitive data from any adversary without significantly compromising the classification accuracy of the proposed model.