[Federated Learning Model for Multimodal Disease Prediction from IU-CXR Dataset]

[CS59-2]

Final Report

5703 Group Based Capstone Project

Group Members

1. Chenxuan Zhou (520562921)
2. Tingfeng Xie (530826048)
3. Yu Shi (500025662)
4. Huiqiao Zhang (520639784)
5. Rayne Zhu (530551405)
6. Xuan Cao (530294074)

School of Computer Science
Faculty of Engineering
The University of Sydney
Australia

11 November 2024

CONTRIBUTION STATEMENT

Our group, taking project CS59-2, with group members (Chenxuan Zhou, Tingfeng Xie, Yu Shi, Huiqiao Zhang, Rayne Zhu, Xuan Cao), would like to state the contributions each group member has made for this project during this semester:

- Chenxuan Zhou

  Chenxuan Zhou's contributions mainly lie in three aspects: core function development, design and implementation of innovative algorithms, and team collaboration.

  Chenxuan Zhou led the design and implementation of the multimodal model, combining the ResNet50 network and the bidirectional LSTM network to build a multimodal model that can process both images and text. In the process of model implementation, Chenxuan Zhou not only integrated different network structures, but also reasonably configured the model's hierarchy, activation function, and regularization. He added an early stopping mechanism to prevent the model from overfitting, and designed a training curve drawing function to help the team better observe the training dynamics of the model in the experiment.

  Chenxuan Zhou proposed and implemented an innovative dynamic weighted aggregation algorithm. The algorithm dynamically adjusts the weight of each client in the aggregation by calculating the weight change of each client to reflect the contribution of different clients to the global model. In order to improve communication efficiency and enhance data privacy protection, Chenxuan Zhou introduced the weight difference function in the aggregation algorithm, so that the client only uploads the weight change instead of the complete weight.

  In the process of teamwork, Chenxuan Zhou, as the team leader, actively participated in the allocation and coordination of tasks. In the function development, Chenxuan Zhou worked closely with team members, with clear division of labor, to ensure the smooth development and testing of various contents. The close cooperation between Chenxuan Zhou and team members effectively improved the efficiency of the team, making the various parts of the project seamlessly connected and successfully completed the project.

- Tingfeng Xie

  Tingfeng Xie's main contributions were in the design and development of the model, optimisation of the model and coordination of the team's work.

  Tingfeng was responsible for errata, optimisation and running several versions of the model throughout the project. He collaborated with other team members to construct a basic multimodal model for the first time in Week 6. Following this, he introduced several regularisation techniques and evaluated model performance. These include techniques such as weight decay, data augmentation and so on.

  In collaboration with his team members, Tingfeng ran the first multimodal federated learning model in Week 9, which used an aggregation algorithm, FedAvg, which is the most basic in the federated learning field. The following week, another common aggregation algorithm in federated learning was also successfully applied to the team's model, and it resulted in a better performance of the model.

  After that, the team leader proposed an innovative dynamic weighted aggregation algorithm. Tingfeng Xie was responsible for collating the innovative aggregation algorithm with the industry benchmark (FedAvg) aggregation algorithm into one model. After its smooth incorporation, it was found through experiments that its performance outperformed the industry benchmark (FedAvg) by a lot.

  In the later stages of the project, based on tutor's feedback on our project work, Tingfeng Xie proposed a post-processing technique which prevents the model from making logical errors in prediction. With this technique, the team's model accuracy was greatly improved.

  Tingfeng Xie was also actively involved in discussions within the team, helping to coordinate the division of labour among team members and providing assistance to those who needed it. His contributions enabled the team to run more smoothly in terms of model development and member collaboration.

- Yu Shi:

  Yu Shi's contribution to this project is mainly in the extensive literature review the construction of multimodal models and the construction and innovation of the federated learning model framework. In the literature review, Yu Shi focused on the application of federated learning models in healthcare. This work laid the foundation for defining the scope and objectives of the project. In addition, Yu Shi focused on sorting out and analysing the aggregation strategies of federated learning and

provided theoretical support and ideas for the project's innovation in aggregation methods by analysing the existing aggregation strategies and identifying the current deficiencies.

Yu Shi's contributions to multimodal integration include data preprocessing, model building, and model performance optimisation. She made many attempts at multimodal data integration, selected and tried many frameworks, and finally determined the multimodal framework of the project as the combination of ResNet50 and LSTM through the comparison of effects. Moreover, Shi Yu pre-processed the input data, including image resizing and normalisation, to ensure the efficiency of model training. In addition, she participated in the training process of the multimodal model, including parameter tuning and performance evaluation, to ensure the performance and stability of the model.

In the final report, Shi Yu ensured that the project met the predefined academic and project standards. She prepared a project status report, an abstract, and an introduction section, provided a concise and clear description of the project background and scope so that readers could clearly understand the scope and purpose of the project, and identified specific questions and research directions for the project.

In group work, Yu Shi actively participates in group discussions to enhance communication and collaboration among members. Her contribution not only enabled the team to obtain more specific research objectives and a solid theoretical background but also made the technology run more smoothly, enhancing the project's progress and efficiency.

- Huiqiao Zhang

  Throughout the project, Huiqiao Zhang played a key role in data preparation, building a federated learning model framework, innovating aggregation algorithms, and solving model problems. She studied and analyzed a large number of references, which promoted the development and implementation of the project.

  In the early stage of the project, she assisted the team members in finding a suitable data set for the project, and used the knowledge learned from NLP to preprocess the text part of the data set. She combined text feature extraction with CNN and LSTM technology to try to provide ideas for data set preprocessing. In addition, she also built the basic framework of federated learning. In the middle of the project, in order to support the development of the model, Huiqiao Zhang

studied a large number of documents and analyzed the advantages and disadvantages of various aggregation algorithms and their adaptability to the project, including FedAvg, FedProx, FedMA, and FedBayes. By comparing the performance of these aggregation algorithms, she provided ideas for the innovative weight algorithm of the model. For the problems encountered by the model in the later stage of the project, such as unstable network transmission and uneven client computing power, she proposed technologies such as weight quantization, asynchronous update, and dynamic client adjustment, and tried to implement and solve these problems through code. At the same time, It also laid the foundation for the weighted differential method of group model innovation.

For the final report, Huiqiao Zhang was mainly responsible for writing the "Related Literature" section. By reading a large number of references, studying the existing federated learning models in the market, and comparing and analyzing them with the group innovation model, the applicability and extensiveness of the group model were demonstrated.

- Rayne Zhu:

  Rayne Zhu contributes to data searching, identifying several potential datasets. Looking for multi-modal data is quite hard. Ultimately, the current dataset we are using was selected from among those options.

  Rayne Zhu contributed to writing sessions focused on data collection, analysis, and preprocessing, including relabeling over 100 labels down to 14. Additionally, she performed data cleaning to reduce irrelevant data while retaining useful information. Rayne Zhu has done data preprocessing to make sure various modalities of data could be integrated into the centralized model and federated learning model.

  Rayne Zhu was instrumental in developing the initial centralized machine learning model, which utilizes the latest models: ResNet50 and BERT. By week four, she demonstrated this model, which achieved an accuracy of at least 65%. She introduced various machine-learning techniques aimed at optimizing the performance of the centralized model, finally resulting in an increase in accuracy from 70% to 99%. Furthermore, Rayne contributed to the sections discussing limitations and future work. With a comprehensive understanding of the project, she succinctly summarized its limitations, incorporating feedback and questions from our client

and tutor. In week ten, Rayne conducted additional research on differential privacy, enriching the discussion in the final section regarding future work.

- Xuan Cao:

  Xuan Cao's main contributions to this project include code building, training and testing the model, model tuning, data visualization, and data analysis.

  At the beginning of the project, Xuan Cao constructed several feasible multi-modal model architectures and federated learning architectures. He used some existing datasets to test on these models. The results of these models help the progress of the whole project. After determining the model architecture, he participated in writing and refining the code of the model as well as tuning the model. Xuan Cao determined the best hyperparameters for the model and the optimizer through multiple tests. His optimizations further improved the accuracy and generalization ability of the model. In the later stage of the project, Xuan Cao was mainly responsible for the testing of the model and the comparison of the team's new algorithm with the traditional algorithm FedAvg. He added a full evaluation section to the code and fixed some problems with the code. Xuan Cao provided a variety of data to analyze and visualize the model's performance by training and testing the model. He provided a variety of charts and visualizations that visually represented the results of model training and testing, helping the team more clearly understand the performance of the model and the room for improvement. Through these, Xuan Cao analyzed the experimental results and compared the advantages of the new algorithm proposed by the team over the FedAvg algorithm.

  For the team cooperation, Xuan Cao actively participated in each group meeting and put forward suggestions and helped to solve the problems existing in the project. He was able to complete his weekly assignments on time. The results of his analysis of the models at each stage ensured that the project could proceed smoothly.

  In the final report, Xuan Cao was mainly responsible for the two sections "Results" and "Discussion". Through the data and images obtained from the experiment, he conducted an in-depth analysis of the performance of the model and systematically summarized the results. Combined with the experimental results, Xuan Cao discussed the practical significance of the new aggregation algorithm and its application potential in federated learning.

All group members agreed on the contributions listed on this statement by each group member.

Signatures:

Abstract

This project explores the application of federated learning models in disease prediction. With the rapid development of the healthcare industry and the field of deep learning, the application of deep learning in the medical field has become more and more common. However, due to the rapid growth and diverse access to medical data, traditional medical diagnosis and prediction models are gradually unable to work effectively in the modern medical field. Due to the sensitivity and privacy of patient data, the traditional centralised approach to data sharing and model training has significant limitations in the medical field, and traditional models will not be able to perform at their total capacity if they do not have access to sufficient data. Therefore, developing a federated learning model with high accuracy and without sharing private patient data becomes essential.

Currently, existing federated learning models face several challenges when used in healthcare applications; for example, they tend not to be very effective when dealing with datasets combining medical images and diagnostic texts. In addition, traditional aggregation algorithms for federated learning models (e.g., Fedavg) are usually unable to effectively handle data differences between different clients, which results in unstable aggregation of the global model.

To this end, this project proposes a novel multimodal model federated learning framework incorporating innovative dynamic aggregation algorithms based on weight changes for multi-category disease prediction. The fusion learning of medical images and clinical diagnosis text data is achieved by introducing ResNet-50 in the visual processing module and using a bidirectional LSTM model in the text processing module. Meanwhile, through the innovative dynamic weighted aggregation algorithm based on weight changes, the model can dynamically adjust the weights of the global model for the contribution of different clients, which improves the stability and accuracy of the model.

This project aims to build a multimodal federated learning model with high accuracy while protecting patient data privacy. The specific research scope is the recognition of lung tumour diseases, focusing on the innovation of federated learning aggregation algorithms and the fusion learning of image-text data. Through the design and optimisation of the federated

learning framework, it is hoped that the accuracy of federated learning models in disease prediction and the prevalence of federated learning model applications in the medical field will be improved.

Through experimental validation, the multimodal federated learning model proposed in this project exhibits excellent performance in the disease prediction task. Compared to the traditional Fedavg algorithm, the dynamic weighted aggregation algorithm significantly improves the accuracy of the global model in the multimodal data environment, and the convergence stability is also significantly improved. This demonstrates the effectiveness and stability of the model in multimodal healthcare prediction tasks.

Although our model performs well in the experiments, the performance of the model is still limited by the quality and quantity of data, and further improvement in generalisability is needed. In addition, the computational complexity of the dynamic weighted aggregation algorithm may put too much pressure on clients with limited computational resources. In the future, we would like to continue to work on optimising the computational efficiency of the dynamic weighted aggregation algorithm and expand the usage scenarios of the model by adding more types of identified diseases to enhance the model's flexibility and usefulness further.

Contents

CHAPTER 1

Introduction

In recent years, as the application of deep learning in medical diagnosis has become increasingly common, the legitimate use of medical data has become an important issue, especially in disease detection and prediction (Hassan et al., 2020). As traditional centralised training models require raw data to be uploaded to a server to train a high-performance model, and in real-world applications, medical data involves a large amount of patient privacy, data access restrictions within organisations and between organisations lead to the phenomenon of 'data silos', which limits the generalisation and accuracy of the model (Yang et al., 2020). and accuracy (Yang et al., 2019).

In this context, federated learning models are one of the solutions to the 'data silo' problem. Federated learning is a distributed training model that does not require healthcare organisations to share raw data; instead, it trains models locally and aggregates the model updates to a central server, alleviating the data privacy protection problem to a certain extent. Specifically, the federated learning model allows healthcare institutions to train local models locally using private data and upload local model parameters to a central server, which then updates the global model by performing weighted aggregation calculations on the locally trained models from multiple institutions and sends the updated model parameters back to the healthcare institutions, through repeated updating and aggregation, thus without sharing the original data enabling the model to gain better generalisation capabilities. Therefore, federated learning, as a distributed learning framework that protects private data, provides new ideas for the application of healthcare data and has received widespread attention (Li et al., 2020).

Despite the significant advances and advantages of federated learning in protecting data privacy, this brings new challenges. One of the main challenges is how aggregation algorithms can maximize efficiency and flexibility while maintaining global model performance. (McMahan et al., 2017). Traditional algorithms such as FedAvg have advantages in terms of simplicity of implementation and low computational overhead. Still, their limitation is

their inability to dynamically adapt to differences in data contributions between clients. This fixed weighting method may lead to inefficient model training in a more complex data environment, or even fail to exploit the potential value of different datasets from clients fully. Therefore, how to design intelligent and adaptive aggregation algorithms to improve global models' generalisation ability and stability remains a focus of current research. In addition, integrating multimodal data is also one of the challenges; most FL methods only focus on image processing and prediction. However, real medical diagnosis scenarios combine medical images with diagnostic text for disease detection and prediction. Therefore, how to make the model effective in processing both image and text information simultaneously is also an open question (Baltrusaitis et al., 2019). In recent years, the Fedprox algorithm proposed by MaMahan et al. has been widely used in distributed models. However, it still performs poorly in the non-independent identically distributed (Non-IID) case (McMahan et al., 2017). In terms of integrating multimodal learning with federated learning, researchers have typically used a combination of visual and textual processing, e.g., using CNNs for images and LSTMs for text, but attempts to combine the two in federated learning are still relatively underdeveloped (He et al., 2016; Hochreiter & Schmidhuber, 1997).

Therefore, this project aims to improve the accuracy of medical prediction while protecting patient data privacy. As existing federated learning models in healthcare prediction still suffer from inefficiencies in aggregation algorithms and difficulties in modal fusion, we hope to develop a more stable and accurate federated learning framework by improving the model structure and optimising the aggregation algorithms, thus addressing the challenges encountered by existing federated learning. Our specific research objectives include:

(1) Design and development of a multimodal model that supports simultaneous input of image and text data to enhance the accuracy of disease prediction.
(2) designing and developing a novel dynamic aggregation algorithm based on weight changes to enhance the model's performance.
(3) to build a patient data privacy-preserving federated learning framework to support cross-institutional collaboration in training high-performance models in healthcare organisations.

To achieve the above objectives, we use the ResNet-50 model for image data and a bidirectional LSTM model for the text part, thus achieving a multimodal model that fuses image and text. This method allows the model to input both image and text data, which expands

the application scenarios of the model and effectively improves the accuracy of the medical prediction model by combining multiple data types. Meanwhile, we designed and developed a dynamic aggregation algorithm based on weight changes, which is designed to dynamically adjust the aggregation weights of the global model by calculating the magnitude of the change of client weights, thus enhancing the stability and accuracy of the model. This method effectively alleviates the problem of performance degradation of federated learning models in complex data environments.

The expected contributions of this project are as follows: first, this project will provide innovative methodological and theoretical support for the combination of federated learning and multimodal models. Second, by developing an innovative dynamic weighted aggregation algorithm based on weight changes method, this project will significantly improve the performance of federated learning models in complex data environments. Finally, the research results of this project will provide valuable references for other cross-institutional collaborative learning on privacy protection and provide new perspectives and developments for the application of federated learning in healthcare.

The approach proposed in this project is not only theoretically innovative, but also has great potential for practical applications in the healthcare field. Combining a multimodal model and a federated learning model with a dynamic weighted aggregation algorithm based on weight changes allows the model to achieve efficient and accurate disease prediction while protecting data privacy, which provides an effective solution for the feasibility of cross-institutional healthcare collaboration. In the future, our research results may continue to expand with more attempts and studies on diverse disease prediction to advance the field of intelligent medical diagnosis further.

CHAPTER 2

Related Literature

2.1 Literature

This project aims to develop an innovative federated learning model that can identify lung diseases. In the literature review section, the application background of the federated learning model, the multi-modal methods of existing federated learning models in the market, the actual challenges and solutions encountered, and the innovative model of this project will be studied and analyzed to prove the market applicability of the model. This section will be supported by a large number of literature and theories, from the perspectives of "the application of federated learning in the medical field", "the application of multi-modal methods in the medical field", and "the challenges and methods of existing federated learning models", to prove the value of the new model proposed by this project based on multi-modal data processing combined with dynamic aggregation algorithm and weight difference to the medical industry.

2.1.1 Scope and standards of literature screening

In order to ensure the cutting-edge and reliability of the content of the literature review, this section selects high-impact academic journal articles and academic papers published in the past six years. Keywords include "Application of federated learning in the medical field", "dynamic aggregation algorithm", "data privacy protection", "multi-modal model", "Challenges of federated learning model", etc.

For the screening of literature, the main criteria are the publication year, number of citations, source of literature and author's authority. The selected journals and articles must have high credibility and citation rate in the academic community.

2.2 Application of Federated Learning in the Medical Field

Federated Learning (FL) plays an increasingly important role in the medical field, especially in data sharing, privacy protection, disease prediction and diagnosis (Guan et al., 2023). Traditional data transmission methods are susceptible to privacy leakage because of the highly sensitive nature of medical data and the relevant privacy issues. A federated learning model as a distributed machine learning method locally trains the devices distributed in different institutions and subsequently the updated parameters are uploaded to the central processor which makes cross-model collaborative training, thus, ensuring data privacy to a significant extent (Pennisi et al., 2022).

(1) Data Sharing and Privacy Protection

Data is a critical part of the medical field. Still, the biggest issue the medical community faces is the separation of data between different hospitals and institutions that highly sensitively treat the data of their patients. Gall et al. (2021) told the public that the untethered privacy model can be used on the peripheral level and a very low privacy-based one is at the core of the federated learning system because the privacy interference in data sharing is avoided by data not being stored and transmitted at the central hub, the CSR model. Hence, the main problem of data sharing as well as protecting privacy in the medical landscape is no longer a big problem. The Google and Mayo Clinic team reported, figuring their way, that Using one centralized system of medical images stored on a closed basis for without data sharing was the basis for intelligent cancer detection and diagnosis. Apart from that, Owki this French biotechnology firm ran a conceptual model of federated learning, where the by sharing physical data, it has allowed medical institutes worldwide their own tumor patient data sets to be used in collaboration without disclosing the patient data (Sheller et al 2020). This project has been a gigantic advance in the medical field.

(2) Disease prediction and diagnosis

Federated learning models also play an irreplaceable role in the application of disease prediction and diagnosis. While protecting patient privacy, it ensures the accuracy of the model's predictions in different patient groups through local client training and weight updates. Google Health has developed a method for early detection of diabetic retinopathy based on a federated learning model. They expanded the amount and scope of data by training retinal image data

of patients from multiple hospitals, thereby improving the prediction accuracy of diabetic retinopathy for different patient groups (Rieke et al., 2020). Dayan et al. (2021) mentioned that it is precisely because the federated learning model distributes patient information from 14 hospitals for training, The joint training of the brain tumor detection model has significantly improved the accuracy of the project model of Intel and the University of Pennsylvania, making a huge contribution to the prediction of brain tumors in the medical field.

2.3 Application of Multi-modal methods in the medical field

Multi-modal methods combined with federated learning models are increasingly widely used in the medical field. Due to the complexity and multi-category of different modal medical data from different medical institutions in the medical industry, including X-ray images, genomes, text medical records, experimental data, etc., multi-modal technology is needed to process, analyze and model complex data.

(1) Personalized medicine

Multi-modal models are crucial in providing personalized medicine to patients. Multi-modality can comprehensively analyze genomic data, case images and experimental test data to provide patients with personalized treatment recommendations. Xu et al. (2021) reported that during the COVID-19 pandemic, the application of such models was a total breakthrough. The technique can merge all the submitted data by all the hospitals and patients who own the tests like X-rays, ultrasound images, and clinical information to the model. As a result, they can easily retrieve personalized treatment and diagnosis which may lead to a positive outcome for patients.

(2) Disease prediction and diagnosis

In figuring out diseases at an early stage, blending data from different modalities illuminates more detailed case characteristics, which is also the function of multi-modality. Federated learning items with multi-modality have a big win when they deal with a complex and diverse field. According to Sheller et al. (2020), more comprehensive medical data could be analyzed by the prediction models based on multi-sensor data and, thus, a generalization that covers a wider variety of patient groups becomes possible. Dealing with the information obtained from different medical institutions where lung cancer patients are diagnosed, the utilization

of multi-modality encompasses capabilities of various medical images, genetic analysis, and clinical record management for exploring the information with enhancing the prognosis of the patient.

2.4 Challenges and methods of existing federated learning models

Even though federated learning models have been extensively used in the medical domain and they have a promising future, there are, nevertheless, the few problems that are supposed to be overcome. These are network transmission issues, especially in the aspects of efficiency and cost, unbalanced distribution of computing power among various clients, and a choice of the weight update and aggregation algorithm for various scenarios (Rieke et al. 2020).

2.4.1 Unstable network transmission

In decentralized learning models, it is necessary for various clients to transfer updated model parameters to the central server. This means that both the massive data model and the variations in the scenario will impose communication burdens on network transmission. This frequently results in data delay and data loss, which are the principal causes of the poor performance of the model and the slow training speed (Almanifi et al., 2023). Here are the main options:

(1) Compression and quantization

Compression and Systems of quantization can form layers of data compression to local clients, facilitating communication overhead and ensuring model parameter updating and transmission are efficient (Mao et al., 2021). The gradient compression approach operates by importance sampling technology, which means transmitting the most significant gradient parameters only, hence, the overall data transfer becomes dramatically lower. It is a strategy to achieve robustness in network transmission by introducing sparsification into the gradient vector (Wang et al., 2018). Besides it, quantization update technology lowers communication expense by compressing the gradient data from floating point numbers into binary numbers (Bernstein et al., 2018). This can, in turn, be used as another technique to ensure the network transmission in federated learning model training.

(2) Asynchronous update

Asynchronous update technology is the key to dealing with the network transmission problem in federated learning. By each local client independently updating parameters, it can avoid synchronous delay and thus reduce network delays Xie et al., 2019. Li et al. (2020) suggested an adaptive asynchronous federated learning algorithm to adjust the frequency of updates for different clients depending on their network conditions, which will in turn, improve the stability of the federated learning model network environment.

(3) Reduce aggregation frequency

The aggregation frequency reduction is achieved by minimizing the communication overhead, which is done by lowering the interaction frequency between the server and the client (Wang et al., 2019). Karimireddy et al., in 2020, proposed the SCAFFOLD algorithm. Including the client in local updates beside running several iterations at the client's end, global aggregations can be reduced and stability transportation will be enabled at the same time.

2.4.2 Uneven client computing power

In Federated Learning, each device related to the model training has its own computing power, storage capacity, etc. and thus the model update time is uneven (Rieke et al., 2020). This, in turn, will alter the convergence speed of the global model. Here are the most frequently used solutions:

(1) Combining asynchronous updates with adaptive computing

The asynchronous update method breaks the traditional concept of synchronous updates for all clients and allows clients to train and update independently. This also avoids the situation where clients with weak computing power participate in each round of updates and affect the overall efficiency. At the same time, through adaptive computing, the computing power is allocated according to the strength of the client computing power and the amount of available resources, and clients with strong computing power are assigned more training tasks (Wang et al., 2020). Li et al. (2020) proved the reliability of this combination method. They discovered that the training speed of the federated learning model can be significantly improved by increasing the update frequency of clients with strong computing power, which is of great help to the training of the federated learning model.

(2) Hierarchical aggregation

Hierarchical aggregation is the process of clustering and ordering the customers according to their types into various categories and layers. After collecting information within each group, the group results are sent to the central processor (Liu et al., 2023). With the introduction of this technology, not only that client with less powerful computers but the entire network can share between them. Hierarchical federated learning has already been introduced by Luo et al. (2020). They discovered that this is not just only the method that decreases the communication overhead but furthermore it makes the model to be well trained in a diverse computing environment.

2.4.3 Weight update and aggregation algorithm of federated learning model

Zhao et al. (2018) suggested that client data is usually not independent and identically distributed (Non-IID) which means that data from different clients are not the same and hence, just averaging out the weights or gradients of the model may result in a decrease in model accuracy and thus, could act as a detriment to model stability. Thus, a judicious selection of the most appropriate weight updating and aggregation algorithm depending on the specific needs of various projects and the exact data conditions is the main method to apply to develop a top-notch federated learning model.

(1) FedAvg

This algorithm makes the local client practice several times, then updates the average parameter and sends them to the central processor. It works in cases where the client data allocation remains stable as well as the computing power relatively evenly shared. Hard et al. in 2016 used the FedAvg algorithm where each mobile device is first trained, then updates the average parameters to the central server for the reduction and besides privacy protection, savings on communication costs have been recorded and trained on the predictive text task model is successful.

(2) FedProx

This algorithm introduces regularization terms so that the client can perform different training steps. It is suitable for situations where the client computing power is uneven or there are

large differences in data distribution. In the case of medical data and heterogeneous data, the FedProx algorithm can adjust the client's training steps, thereby ensuring the convergence performance of the model (Li et al., 2020).

2.5 Comparison of the innovation of our model

(1) Weight difference solves the network transmission problem

For the problem of unstable network transmission faced by the federated learning model, we chose the weight difference method. This is a compression and quantization method. Since the core of weight difference is to reduce the amount of communication data, we only transmit the change in model weight since the last update each time, rather than the complete model weight. This reduces the amount of data for each communication and reduces the overall communication cost of the model.

(2) Solve the problem of uneven client computing power

Since the number of clients in this project is small, there is no impact on model performance caused by uneven client computing power. Therefore, we did not deal with this problem when building the federated learning model. For this problem, in subsequent actual scenario applications, it is still necessary to judge the actual number of clients and the degree of impact of this problem on model performance to determine whether further model optimization is needed.

(3) Innovative weight adjustment and aggregation method

According to project requirements, we have innovated a federated learning aggregation method based on dynamic weight changes, which can better adapt to heterogeneous environments of data and computing power. This method is to calculate the total amplitude of client weight changes after each round of local training and upload it to the central server, The central server then dynamically allocates the aggregation weights according to the magnitude of the change. This method allows clients with greater contributions to account for a higher proportion in the global model.

Unlike the traditional FedAvg and FedProx algorithms, this method can adjust the aggregation weights in real time according to the client's training situation. Compared with these two algorithms, our model has the following advantages:

- Reflecting the client's contribution through the weight change, it is more adaptable to data heterogeneity.
- Clients with strong computing power have high weights, which adapt to clients with different computing power.
- The weight differential mechanism reduces communication costs and data leakage risks, and ensures communication efficiency.

Therefore, our innovative algorithm model not only improves the generalization ability of the model, but also provides customers with high communication efficiency and information privacy.

CHAPTER 3

PROJECT PROBLEMS

3.1 Project Aims & Objectives

The final aim of this project is to design and implement a multimodal federated learning framework with privacy preserving mechanisms and high performance for disease prediction. Federated learning differs from traditional machine learning by distributed training, allowing clients (healthcare organisations) to use raw data for local model training, and each client only needs to upload updates to parameters to a central server after completing the model training locally, rather than the data itself. The central server performs aggregation after receiving the parameters from the clients to form a global model. The most commonly used algorithm in this aggregation process is Fedavg, which performs a simple weighted average of the client models. Still, the different data distributions and learning contribution levels of each client may result in low-quality data or low-contributing clients affecting the global model. Thus, the model often fails to achieve the desired performance in real-world applications.

Therefore, to improve the efficiency and accuracy of model aggregation, one of the objectives of this project is to propose an innovative aggregation algorithm, a dynamic aggregation algorithm based on weight changes. During this algorithm, each client calculates the model weight change locally, and the weight change rate is obtained by calculating the L2 paradigm of each layer's weight change. The weight change rate represents the client's contribution to a training round. Next, the central server collects the weight changes of all clients, normalises them and calculates the weight coefficients for each client. The larger the coefficient, the larger the magnitude of change represents, indicating that the client contributes more to the global model, and the weight coefficient will be higher. During the aggregation process, the central server uses these weight coefficients to obtain new global model weights by performing a hierarchical, weighted summation of all weight changes. This process ensures that clients with large weights have a more significant impact on the global model, and clients with

small weights have a negligible impact on the global model. Meanwhile, while the dynamic aggregation algorithm based on weight changes brings higher performance, the computational overhead is also relatively large, and to reduce the communication overhead, we add the weight difference function. Specifically, instead of uploading the complete weight matrix, the client only uploads the weight update changes. This not only saves computational overhead but also enhances data privacy protection. Furthermore, in tasks such as disease prediction, a single data source (e.g., using only medical image data or clinical description text data) may not provide sufficiently complete and diverse information for model training. Therefore, using multimodal models to improve the accuracy of lung tumour recognition models is also the objective of this project. To this end, we use a ResNet50 network for medical image feature extraction in the visual data processing module and a bi-directional LSTM network for relevance capture and semantic understanding of clinical diagnostic text information in the text processing module.

In summary, our aim is to improve the generalisability and superiority of multimodal federated learning disease prediction models for medical diagnostic applications by designing and implementing multimodal disease prediction models and innovative dynamic aggregation algorithms based on weight changes.

3.2 Project Questions

The core research questions of this project focus on the optimisation of federated learning frameworks, data privacy protection and multimodal data fusion. Firstly, not many attempts have been made to fuse image and text data in the field of medical prediction, but to achieve higher accuracy in disease recognition, it is necessary to train the model using multifaceted data sources and data types. Therefore, we designed a multimodal diseases prediction model. The challenge of achieving multimodal fusion is to overcome the differences in information representation of different modal data. This difference leads to the heterogeneity of the data, posing a challenge for feature extraction and fusion of the model. To address this challenge, we use ResNet50 for image processing and bi-directional LSTM for text processing to extract features from image and text data, respectively, and then achieve multimodal feature integration through joint learning, which ultimately improves the model's accuracy.

Second, data privacy protection is a key requirement of the project. Practical applications of deep learning models in the medical field often encounter the problem of access restrictions to sensitive data, and the incompleteness of data access limits model training and performance to a large extent. Therefore, achieving high-performance medical prediction models while protecting data privacy is one of the challenges of this project. First, in order to protect data privacy, we apply the existing federated learning framework and use the features of federated learning models to build a federated learning model that only requires model sharing without sharing raw data to complete model training. The method well achieves the balance between privacy protection and model effectiveness, and ensures the privacy and security of medical privacy data. Furthermore, another problem we face is the limitation and low performance of federated learning models under different arithmetic powers of different clients. Traditional federated learning models are prone to situations where specific clients have too much or too little influence on the global model when there are different data types and differences in computational power between different clients, thus affecting the model's performance. Therefore, to better balance the influence of each client on the global model, we innovate and enhance the aggregation algorithm in the federated learning framework. We developed a dynamic aggregation algorithm based on weight change, which dynamically adjusts the weight of each client in the global model by calculating its rate of change instead of fixing a constant weight based on the number of samples or the average distribution, to achieve a more reasonable allocation of aggregation weights. This strategy not only improves the model's performance in heterogeneous data environments, but also improves the aggregation efficiency and generalisation ability of the model.

In conclusion, this project has problems and challenges in multimodal data fusion, data privacy protection, and optimisation of the federated learning framework. However, by identifying project problems and developing targeted strategies, we take the project problems as a starting point and make the model more accurate and robust in practical applications through technical optimisation and innovation.

3.3 Project Scope

In this project, we have limited our research to constructing a multimodal fusion model for medical image and text data and the innovation and optimisation of aggregation algorithms in federated learning. Our goal is to create a disease prediction model that applies to most

disease types. Still, due to the availability of clinical data and the project's time constraints, we cannot achieve coverage of all disease types. However, there is great potential for the multimodal federated learning model developed in this project to be helpful in identifying and predicting a wide range of diseases.

Technically, this project only focuses on combining ResNet50 and bi-directional LSTM in designing and implementing multimodal models. Although more complex deep learning models may bring more outstanding research results for the project, considering that there may be insufficient client computational resources after being put into real-world applications, we finally chose these two lighter models to ensure the system's practicability and stability in various environments. In the development of federated learning aggregation algorithms, we only designed and implemented the dynamic aggregation algorithm based on weight changes and the comparison between this algorithm and the traditional aggregation algorithm (FedAvg). We did not design the development and comparison of other aggregation algorithms.

In the selection of the dataset to ensure that the problem solved by the model is consistent with the actual requirements, we selected medical datasets that are publicly available and include medical images and clinical diagnostic texts for a wide range of disease types. This dataset fully meets the requirements of the multimodal model inputs and is representative. By using this dataset to train and test the model, the project team can effectively apply the performance of this multimodal federated learning model in the medical disease recognition task.

Overall, the scope of the project is clearly defined as the design and performance enhancement of multimodal federated learning models for disease prediction. .

CHAPTER 4

METHODOLOGIES

4.1 Methods

This project uses a multimodal model and an innovative dynamic aggregation algorithm to build a federated learning framework that can effectively process heterogeneous data. The core of the multimodal model is to combine image and text data. By using the ResNet50 network in the visual processing module and the bidirectional LSTM in the text processing module, the joint learning of different types of data is successfully realized. This model design can extract and fuse features between different modalities, enhancing the expressiveness of the model under multimodal data. At the same time, the project introduces an innovative dynamic aggregation algorithm, which dynamically adjusts the weight of each client in the global model according to its contribution in training by calculating the weighted aggregation strategy of the client weight change amplitude. This algorithm not only improves the stability of the model under the conditions of heterogeneous data distribution and uneven computing power, but also effectively alleviates the performance problems caused by uneven data distribution in traditional federated learning. In addition, to further improve the robustness of the system, we implemented an early stopping mechanism and weight change amplitude control in the project to avoid overfitting of the model and ensure privacy security during the federated learning process. Overall, the methods of this project have demonstrated innovation and practicality in multimodal data fusion and dynamic aggregation strategies, and have provided new ideas for the expansion of federated learning in practical application scenarios.

4.1.1 Model for images — ResNet50

This figure shows the core structure and functions of the ResNet-50 network. ResNet-50 is a residual network architecture used to extract deep features in images, especially suitable for complex image classification and recognition tasks. The input image first passes through

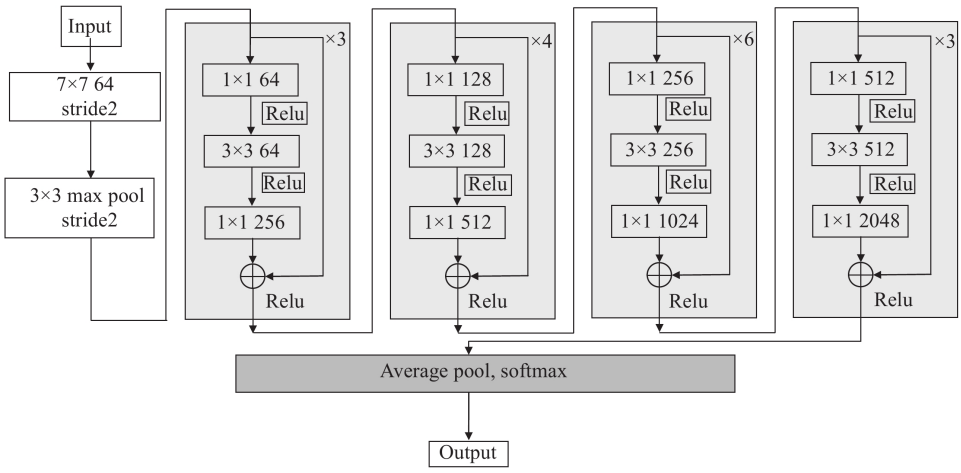

FIGURE 4.1: Structure of ResNet50(MIN J, WU Z D, ZHENG L, et al, 2023)

a 7x7 convolutional layer (with 64 filters and a stride of 2) to extract the initial low-level features. Next, it passes through a 3x3 max pooling layer with a stride of 2 to reduce the size of the feature map. The function of this part is to compress the size of the input image while retaining the most important edge and texture features.

The core of ResNet-50 lies in its residual block structure. Through residual learning (i.e., introducing shortcut connections), the input of each convolutional block can be directly passed to the output of the block. This design enables the network to "skip" learning, thereby alleviating the gradient vanishing problem in deep networks. The ReLU activation function is connected after each convolutional layer to ensure the nonlinear expression ability of the network.

The figure shows the four main stages of ResNet-50, each of which contains multiple residual blocks. Stage 1 is used to capture basic features. Stage 2 captures mid-level features in the image. Stage 3 extracts deeper features. Stage 4 is the deepest feature extraction module, which is used to identify high-level features, especially complex patterns in high-dimensional space. The number of residual blocks and convolution kernels in each stage increases gradually to gradually enhance the network's feature expression ability, thereby achieving efficient extraction of complex image features.

After all residual blocks, the network uses a global average pooling layer to globally average the feature maps of each channel, further reducing the number of parameters and computational cost. The feature vector after average pooling passes through a Softmax layer for the output of the classification task. The function of this output layer is to convert the deep

features into probability distributions, thereby outputting the predicted probability of each category.

4.1.2 Reasons for choosing ResNet-50

ResNet-50 is a convolutional neural network with a mature structure and reliable performance. It can efficiently extract high-level features of images through a 50-layer residual block design, and is suitable for medical image analysis that requires deep features. At the same time, compared with deeper networks such as ResNet-101, ResNet-50 achieves a good balance between model complexity and computational cost. Considering the high-resolution requirements of medical image data, the depth of ResNet-50 is sufficient to extract key pathological features without the computational burden caused by the network being too deep.

In the framework of federated learning, model training needs to be run in a distributed environment. ResNet-50 can still guarantee the performance and stability of the model with less computing resources, and is suitable for running on client devices. For some client devices with limited computing power, ResNet-50 has higher executability and stability than more complex networks (such as ResNet-101).

The ResNet series of networks have been widely verified in the field of image recognition, especially in medical image analysis tasks. Based on this, we chose ResNet-50 as the image processing backbone network of the project to help ensure the effectiveness and performance of the model.

4.1.3 Model for text — LSTM

[h] Long Short-Term Memory (LSTM) is a special type of recurrent neural network (RNN) designed to process and predict long-distance dependencies in time series data. LSTM can effectively control the flow of information by introducing three gate structures (forget gate, input gate, and output gate), avoiding the gradient vanishing problem common in ordinary RNNs. The figure 2 shows the structure of an LSTM cell.

The internal structure of LSTM includes the following "gates":

The forget gate determines which information needs to be forgotten in the memory cell state $Ct1$ at the previous moment. The calculation of the forget gate depends on the current input Xt

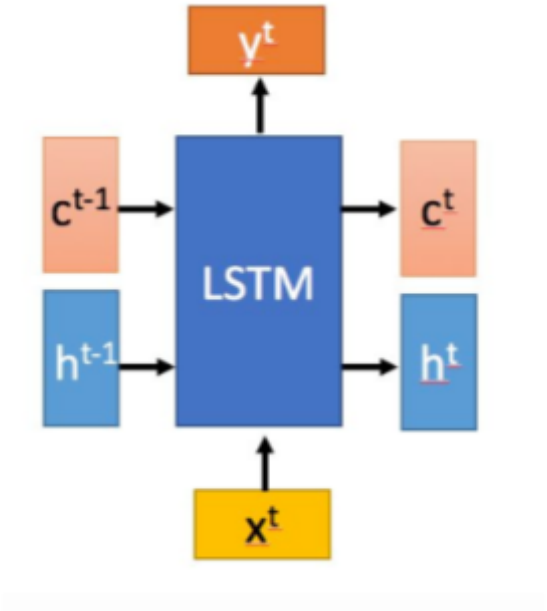

FIGURE 4.2: Structure of LSTM

and the hidden state *ht*1 at the previous moment, and outputs a value between 0 and 1 through a sigmoid activation function. If the output is 0, it means that the information is completely forgotten; if the output is 1, it means that the information is completely retained.

The input gate controls the impact of the current input information on the memory cell. It determines what new information can be added to the memory so that the model can gradually update and supplement its memory.

The output gate controls which information of the current memory unit *Ct* needs to be output to the next hidden state *ht*. Through this update, LSTM is able to selectively output important information at the current moment to the next unit while retaining valuable long-range information. The existence of the output gate helps the model selectively output important information at the moment without leaking all the details.

4.1.4 Reasons for choosing LSTM

The gating mechanism of LSTM enables it to retain important long-term dependency information while avoiding interference from unnecessary information, making it suitable for processing sequence data that requires contextual association. Medical text data usually has contextual semantic associations. For example, in medical records or diagnosis descriptions, symptoms or diagnosis information mentioned earlier may affect the interpretation

of subsequent descriptions. Therefore, LSTM can more effectively capture these long-term dependencies, thereby improving the effect of text understanding. Ordinary RNNs are prone to gradient vanishing or gradient exploding problems when processing long sequences, while the forget gate structure of LSTM effectively controls the gradient flow while retaining useful information. For long sentences or multi-segment records in medical texts, LSTM can stably learn in longer sentence structures, ensuring the gradual transmission of information in the sequence and avoiding information loss or instability.

This project uses a bidirectional LSTM, which can capture information from both directions of the sequence (forward and backward). For medical text data, a bidirectional LSTM can simultaneously obtain contextual information before and after the sequence, enhancing the understanding of the medical record description. For example, the description of some diseases may have related information at the beginning and end of the sentence. The bidirectional LSTM can focus on this information at the same time, further improving the model's understanding ability.

LSTM can generate contextual feature representations on a sequence of time steps and can be combined with other models to form multimodal feature representations. In this project, the text features of LSTM and the image features of ResNet-50 can be naturally combined to form a powerful multimodal model that comprehensively utilizes image and text information for diagnostic prediction.

4.1.5 Federated Learning Framework

In this project, we built a federated learning framework for multimodal medical data (including images and text). This framework aims to use local data from different clients (such as hospitals, medical devices, or institutions) to collaboratively train a global model, thereby improving the adaptability and performance of the model under different data distributions. Based on the framework design of federated learning, data does not need to be processed centrally on the server, but is dispersed on the client for local training, and the server only aggregates the model weights. This design not only protects data privacy, but also effectively utilizes distributed data resources, which is particularly suitable for application needs in the medical field.

4.1.6 Architecture and components

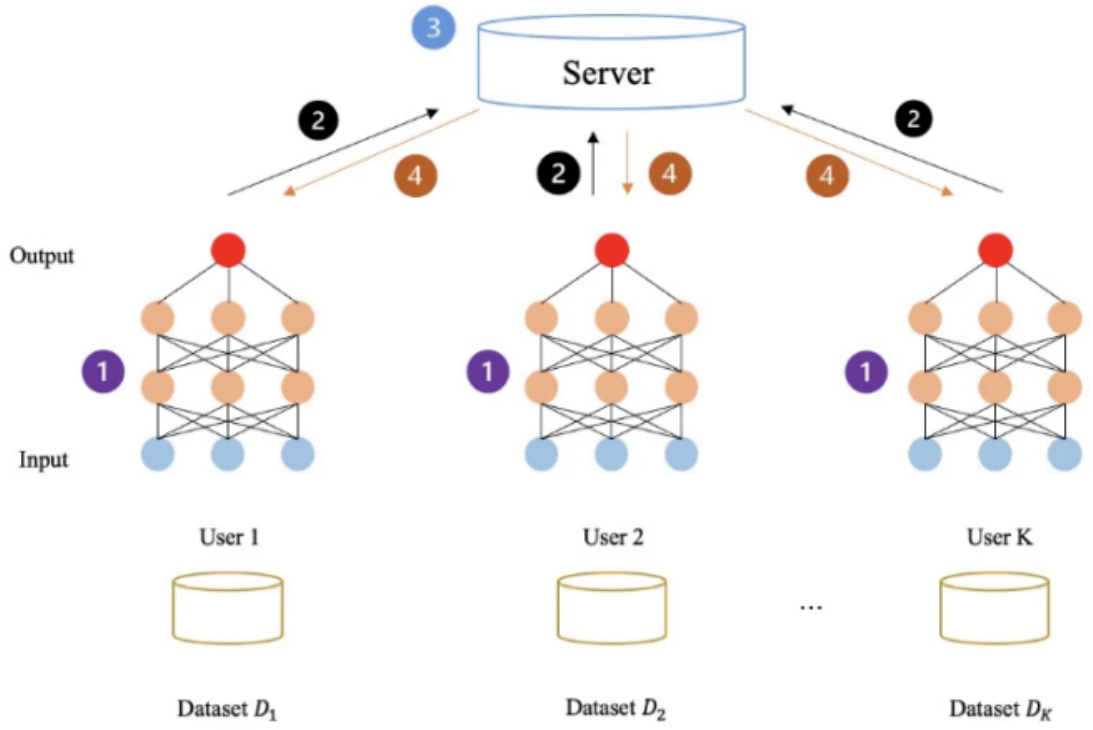

FIGURE 4.3: Federated Learning Components and Processes

This diagram shows the overall architecture of the federated learning framework and the interaction process of each component. The architecture consists of a client and a server, and completes the training of a distributed model through multiple steps.

Each client represents a data holder (such as User 1, User 2,..., User K). These clients each have a local dataset (such as Dataset D1,D2,..., DK) and train the model locally. Each client trains the model locally based on its own dataset Dk (step 1 in the figure) and updates the model weights. Since the data is stored locally, this process protects data privacy and avoids centralized data upload. After completing local training, the client sends the updated model weights to the server (step 2), which then aggregates these weights. After completing local training, the client sends the updated model weights to the server (step 2), which then aggregates these weights. The client receives the updated global model weights sent by the server (step 2), applies the updated weights in the local model, and then enters the next round of training.

The server plays a core role of coordination and aggregation in the federated learning framework. It is responsible for collecting model weights from different clients and distributing the aggregated global model weights back to each client. The server first initializes a global model (such as a multimodal model of ResNet-LSTM) and distributes the model to each client for initial local training. The server receives model weights from the client in each round of training (step 2) and stores these weights in preparation for weighted aggregation. The server aggregates the collected client model weights (step 3). In this project, the server uses a dynamic weighting algorithm based on model weight changes to dynamically adjust the weight ratio of each client according to the magnitude of the change in the client model weight, thereby generating an updated global model. The server distributes the aggregated global model weights to each client (step 4) so that the client can use the updated model weights in the next round of training.

4.1.7 Data flow and training process

Step 1 - Local training of the client: Each client performs a round of model training based on its local data to generate locally updated model weights. Since the data does not leave the client, data privacy is protected.

Step 2 - The client uploads the model weight to the server: After completing local training, the client sends the model weight or weight update to the server.

Step 3 - Server aggregate weights: The server dynamically weights and aggregates the weights of all clients to generate new global model weights. The dynamic weighting algorithm assigns different aggregate weights to each client based on the different training contributions of the client.

Step 4 - Server distributes updated global models: The server distributes the aggregated global model weights to each client, and the client uses the updated global model weights for the next round of local training.

4.1.8 Characteristics of Federated Learning in Projects

One of the biggest advantages of the federated learning framework is that data does not have to be uploaded to a central server, thus protecting the privacy of sensitive medical data. This

is particularly critical for medical data, which is dispersed among various clients (such as hospitals), reducing the risk of leaking patient privacy.

After using the dynamic weighting algorithm, low-contribution clients will not excessively affect the global model, thereby accelerating the convergence of the model. In addition, the server only needs to receive the weight change instead of the complete data or model weight, which reduces communication overhead and improves overall efficiency.

The server does not need to process data directly, but instead performs weighted aggregation on the model weights, greatly improving the system's training efficiency and resource utilization.

4.1.9 Dynamic weighting algorithm based on model weight changes

The dynamic weighting algorithm based on the change of model weights is an innovative method for aggregating model weights in the federated learning framework. Traditional federated learning methods (such as FedAvg) usually average the model weights of each client, failing to fully consider the differences in client data distribution and training contributions. This algorithm analyzes the change in client model weights after each round of training and dynamically adjusts the weighted proportion of each client in the global model update, thereby achieving a more accurate and adaptable aggregation strategy in heterogeneous data distribution and uneven computing power environments.

4.1.10 Algorithm goal

The goal of the dynamic weighting algorithm based on model weight changes is to dynamically adjust the aggregation weight according to the change in the model weight of each client, so that clients with greater training contributions have higher aggregation weights, thereby improving the generalization ability and stability of the global model on heterogeneous data.

4.1.11 Algorithm steps and ideas

The dynamic weighting algorithm based on model weight change dynamically adjusts the weight coefficient of each client in the global model aggregation by calculating the change in the model weight of each client during the local training process. First, after each client completes local training, the change in the weight of each layer is calculated and summarized

to quantify its contribution in this round of training. Then, the server collects the weight change amplitude of all clients and calculates the weight coefficient of each client. The weight coefficient is normalized based on the weight change ratio of each client to better reflect the relative influence of each client on the global model. Next, the server aggregates the weight change of each layer according to the weight coefficient to obtain the weighted update of the global model. Finally, the server applies the weighted update to the global model weight to generate an updated global model. The following details are shown.

4.1.12 Calculate the weight change of each client

In each round of local training, client *k* trains the model based on its local dataset and updates the weights of each layer of the model. For the *i*th layer of the model, the weight change of client *k* is recorded as the difference between the weight of the next layer and the weight of the previous layer, which is the weight update generated in the current round of training without involving the weight comparison of the previous round.

The weight change amplitude δk of client *k* is the sum of the *L*2 norm of all its layer weight changes, expressed as:

$$\delta_k = \sum_{i=1}^{L} ||W_k^{i+1} - W_k^i||$$

Among them, *L* is the number of layers of the model, *i*represents the index of each layer of the model, and *k* represents the identity number of the client.

4.1.13 Calculate the weighting coefficient for each client

On the server side, after collecting the weight change amplitudes δk uploaded by all clients, the sum of the weight change amplitudes of all clients is first calculated to normalize the weight coefficient of each client. Then, based on the weight change amplitude δk of each client, its weight coefficient in the global model aggregation is calculated. Specifically, the weight coefficient ωk of client *k* is the ratio of its weight change amplitude to the sum of weight changes, and the formula is:

$$\omega_k = \frac{\delta_k}{\sum_k \delta_k + \epsilon}$$

Among them, ϵ is a very small value to prevent the denominator from being zero to ensure the stability of the calculation. In this way, the weight coefficient ωk of each client reflects its relative influence in the global model aggregation. A larger weight change corresponds to a higher weight coefficient, indicating that the client contributes more to the model training, thus obtaining a larger weight in the aggregation process. This weight coefficient will be used in the next step to perform a weighted average of the weight changes of each client.

4.1.14 Calculate the weighting coefficient for each client

The server uses the weight coefficient ωk of each client to perform hierarchical weighted aggregation on the weight updates of each client, thereby obtaining the final weight update $\Delta W i$ of each layer of the global model. In addition, we introduced the weight differential function in this step to reduce the amount of transmitted data and improve communication efficiency. The core of the weight differential function is that the client only uploads the change in the weight of each layer in its local training, rather than the complete weight, thereby reducing the demand for transmission bandwidth.

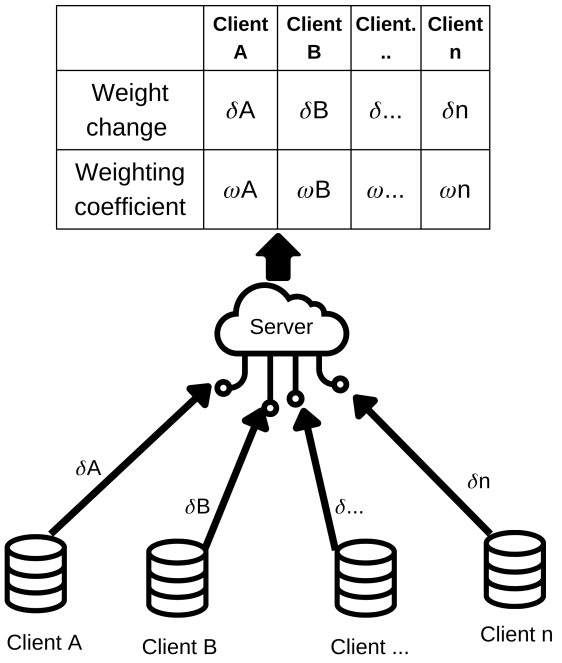

FIGURE 4.4: Weighting coefficient calculation process

The global model updates weights layer by layer

Layer Index	Weighted update weights
1	ΔW1=ωA·ΔWA1 + ωB·ΔWB1 + ... + ωn·ΔWn1
2	ΔW2=ωA·ΔWA2 + ωB·ΔWB2 + ... + ωn·ΔWn2
...	...
n	ΔWn=ωA·ΔWAn + ωB·ΔWBn + ... + ωn·ΔWnn

FIGURE 4.5: Global model update weight calculation process

For each layer *i* of the model, the server calculates the weighted aggregate weight change ΔWi of the *i*th layer of each client, and the calculation formula is as follows:

$$\Delta W_i = \sum_k \omega_k \cdot \left(W_k^{i+1} - W_k^i\right)$$

Among them, ωk is the weight coefficient of client *k*, which is determined by the weight change amplitude of each client. The difference in brackets represents the update amount of the weight of the *i*th layer by client *k* in the current round of local training, that is, the weight difference.

This formula states that each client only needs to upload its weight updates (i.e., differential values), and the server performs a weighted sum of these weight updates to obtain the aggregated updates ΔWi for each layer. By introducing weight differentiation, the client no longer needs to transmit complete weight information, but only the variation, thus significantly reducing communication overhead.

The introduction of the weight differential function brings obvious advantages. The client uploads only the weight update instead of the complete weight matrix, which reduces the data transmission volume and is suitable for federated learning environments with limited bandwidth. Transmitting weight differentials instead of complete weights can reduce the potential risk of data leakage to a certain extent, because the weight update amount often does not directly expose sensitive information of the original data. Due to the reduction in

transmission volume, the server-side computing pressure when processing aggregation is relatively reduced, and a round of aggregation can be completed more quickly.

4.1.15 Load the complete global model

The server uses the weighted aggregate update ΔWi of each layer calculated in the last section(4.1.6.2.3 Layered weighted aggregation weight change) to update the weights of each layer of the global model. This step is the key to updating the global model. The weight of each layer is adjusted by the weighted aggregate weight change, so that the global model can integrate the training contributions of all clients to obtain an improved model. For each layer *i* of the model, the server updates the weight of the global model according to the aggregated weight update ΔWi. The update formula is as follows:

$$W_i^{t+1} = W_i^t + \Delta W_i$$

Among them, t represents the training round of the global model, and i represents the layer index of the global model. ΔWi is the weighted aggregate update of the *i*th layer calculated in the fourth step, which represents the contribution of all clients to the weight change of this layer.

This operation is performed independently on each layer, so that each layer of the global model can absorb the training information of each client and complete the weight update of this round.

4.2 Data Collection

Compared with single-modality data, the medical multi-modal data would enhance the model's performance and robustness. (Ramachandram, D., & Taylor, G. W. , 2017.) Demner-Fushman et al. (2016) prepared a radiology dataset, named the Indiana University Chest X-ray Collection (IU-CXR), to facilitate easier distribution and retrieval. The dataset comprises chest X-ray (CXR) images and reports, each linked to a corresponding patient ID. It is publicly accessible from the Open Access Biomedical Image Search Engine (OpenI). For convenience, the dataset can be fully downloaded from Kaggle. Upon comparison, the data downloaded from Kaggle is exactly the same as the data retrieved from OpenI when checked the amount

of data. Another reason we directly retrieved data from Kaggle is, that the data format has been already transferred to the format we could directly use.

Except the images folder, contains a total of 7,470 unique images. There are two CSV documents. One contains each image's file name, associated with a unique identifier (UID), indicating that multiple images may correspond to the same patient. Another is a comprehensive view of radiology records for chest X-rays. The dataset's radiology reports are divided into key sections: Problems, Comparison, Indication, Findings, and Impression, each offering distinct insights into the patient's medical condition.

The IU-CXR dataset is distributed under the Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license, ensures that the data is accessible for academic and research purposes while preserving the integrity of the original work.

4.3 Data Analysis

4.3.1 Data Re-labeling

In the original dataset, the "Problems" column lists various disease names that represent symptoms and diagnoses. These entries can be used as labels for a predictive dataset.

However, there are over 100 unique disease names, making it impractical to analyze all of them given the limitations of our computational resources and the size of the dataset. Since our research focuses on multi-modal classification and prediction, it is essential to re-label and reduce the number of labels. This step allows us to streamline the dataset for efficient analysis while maintaining the relevance of the labels for our predictive tasks.

We categorized different diseases based on their semantics and literal meanings, grouping terms into categories such as gastrointestinal diseases, pulmonary diseases, dermatological conditions, and neoplasms, among others, thereby organizing the data into broader, clinically relevant groups. This classification approach ensures that each disease type accurately reflects its true meaning within the dataset, facilitating subsequent analysis and processing, supports more effective multi-modal classification, and aligns with the goals of predictive modelling in medical imaging.

The categories are as follows:

Based on the analysis of medical terms within the "Problems" column, a total of 14 distinct categories were defined to facilitate re-labelling and improve the dataset's usability for predictive modelling.

4.3.2 Label Encoding

Each category is encoded as a number for future analysis. The **labels** column encodes each re-labeled category as numerical values, where multiple categories are separated by semicolons (e.g., "3;1" for multiple conditions).

4.3.3 Data Cleaning

The mentioned columns for "Comparison", and "Indication" will be deleted, as patient information and background details are not relevant to our project objectives and do not contribute useful data. The "Findings" and "Impression" columns were merged into a single column to retain valuable medical information, which we treat as textual medical data and renamed as "notes."

Now, the images, "notes", and encoded labels are aligned with corresponding UIDs. It is important to note that each UID may be associated with multiple images but will have the same notes and labels. Consequently, the total number of entries in the dataset will match the number of images. Finally, we merged the two CSV documents and ensured that all data corresponded to the UIDs.

With these modifications, we have a final cleaned dataset with a well-structured design.

4.3.4 Text Preprocessing

In our project, text preprocessing begins with the creation of a tokenizer designed to handle a maximum of 10,000 unique words. This tokenizer is essential for converting raw text data into a numerical format that can be easily processed by machine learning models. It is fitted on the 'notes' column of our dataset, enabling it to learn the vocabulary and assign integer values to each word. Following this, the text data is transformed into sequences of integers, where each sequence corresponds to a particular note. To ensure uniformity in the input size, these sequences are then padded to a fixed length of 100. This padding is applied at the end

of each sequence, with any excess words truncated, creating a consistent input format for subsequent analysis

4.3.5 Image Preprocessing

For image preprocessing, a dedicated function is employed to manage the loading and preparation of image data. Each image is opened and converted to the RGB color format to standardize the color channels. The images are then resized to a target dimension of 224 by 224 pixels, a common requirement for many convolutional neural network architectures. After resizing, the pixel values of the images are normalized by dividing them by 255.0, which scales the values to a range between 0 and 1. This normalization step is crucial as it helps improve the convergence of the model during training. The processed images are collected into an array, ready for integration into the machine learning pipeline.

4.4 Deployment

The reason why this project does not need to be deployed is mainly due to its research nature and experimental purpose. The main purpose of this project is to verify and explore the effectiveness and performance improvement of new algorithms (such as dynamic weighting algorithms based on model weight changes). This kind of project is usually completed in a controllable experimental environment to test the improvement effect of algorithms and models, rather than directly applying them to actual production environments. During the experimental phase, the main goal of the project was to observe and record the model training process, aggregation effect, and changes in various performance indicators by simulating the federated learning framework of the client and server. Therefore, the project delivery content focused on experimental data, performance evaluation, and visualization results, rather than actual product deployment. During the experiment, researchers may frequently adjust algorithm parameters or model structures, and may even need to replace models, update aggregation methods, etc. The lightweight architecture of the experimental environment allows these modifications to be completed quickly, while a more complex testing and release process is required in a production environment. After the experiment, the results of the project are often presented in the form of academic reports, papers, visual data, etc., rather than actual software systems, so there is no need for production-level deployment.

4.5 Testing

In our project, we tested the learning effect of the model. Our test objects include our algorithm and FedAvg algorithm under the same model architecture and data allocation. Here, we use the validation set for testing after training in each epoch for each client, and obtain evaluation data including accuracy, loss, F1 score, etc. We calculate the average of the evaluation data for each client to avoid the impact of performance differences across clients. We use the data from the last test on the validation set as the final test result. The specific test results can be found in Section 7.

CHAPTER 5

RESOURCES

5.1 Hardware & Software

The model code can be run in local environments such as Jupyter Notebook, Visual Studio Code, or some cloud platforms such as Google Colab. Simply adjust the read path according to the actual location of the dataset. In addition, it may be necessary to unify the versions of TensorFlow and Keras when running on some cloud platforms to prevent version incompatibility issues.

5.1.1 Software

Programming Language: Python - Python is the programming language used for our project. Its versatility and the wide range of libraries that can be called provide great support for our project.

Libraries and Packages:

- **Pandas:** It is used in the data preprocessing phase and supports our team in data manipulation and analysis, helping us to deal with large datasets efficiently.
- **NumPy:** It facilitates numerical computation and is used to work with arrays and matrices in data.
- **TensorFlow:** It provides the framework for building, training, and deploying our deep learning models. In our model, the version used is 2.5.0.
- **Scikit-learn:** It is used to provide various data preprocessing and model evaluation tools to help in feature selection and model performance evaluation.
- **PIL:** It is used to process image data to enable image preprocessing and enhancement for better performance of the model.

IDE: Our team mainly uses Visual Studio Code for code development and debugging, which as a popular IDE with many plug-in extensions provides a lot of convenience to our team during the development of the project.

Code Repository: GitHub is used to store model code and dataset files as well as instructions for using the code.

5.1.2 Hardware

Primary computing device: The computer that our team primarily uses to train models is equipped with an Nvidia GeForce RTX 4090 with 24GB of VRAM. This GPU allows for efficient local model training that takes about 5 seconds per period.

Auxiliary computational resources: We use Google Colab's high-RAM environment with an A100 GPU, mainly for the model design phase. In addition, we specified the runtime type as tf25. However, due to the slow training time, which takes about 60 seconds per epoch, Colab was used as a supplementary development tool for a trial environment during the development of some of the model's features, rather than a primary training environment.

5.2 Materials

The dataset used for our project is called 'Indiana', which was initially 14.2 GB. However, after a thorough data cleaning by our team, we reduced it to 13.2 GB by removing irrelevant and redundant data points. This cleaned dataset has been uploaded to GitHub to ensure accessibility and version control. The data cleaning process included filtering out null values, removing duplicate entries, and discarding data points that did not contribute to the training objectives.

Supporting Documentation: Documentation (including code comments and supporting notes) is stored on GitHub along with the code to allow team members to collaborate effectively and track updates. In addition, our team collaborates on team tasks via Google Drive, such as the preparation of a number of reports based on group submissions, the preparation of presentations for presentations to tutors and clients, and the updating of the model code for each version, all of which are operated in the midst of this. All members can edit these files at any time to carry out our teamwork.

5.3 Roles & Responsibilities

- Chenxuan Zhou

  As the team leader and lead developer, Chenxuan Zhou holds a dual role that combines project management and technical development. The responsibilities are critical to ensuring the successful execution of the project from both strategic and technical perspectives. Chenxuan brings leadership in organizing the team, overseeing project milestones, and making key technical decisions.

  Chenxuan Zhou is responsible for setting project goals, defining milestones, and organizing team activities to ensure steady progress toward deliverables. Establishing timelines, assigns tasks based on each member's strengths, and maintaining communication with all team members to keep them aligned on project objectives. Chenxuan also organizes regular team meetings to discuss progress, address challenges, and provide guidance, ensuring that each team member has the support they need.

  As the developer, Chenxuan Zhou oversees all aspects of the project's technical implementation. Chenxuan Zhou plays a pivotal role in designing the core architecture, particularly focusing on the integration of the multi-modal ResNet50 and bi-directional LSTM models. Chenxuan Zhou expertise in model selection and customization helps the team implement an architecture that is both robust and aligned with the project's complex data requirements.

  Chenxuan Zhou leads the development of an innovative dynamic weighting aggregation algorithm, a core component of the federated learning framework.Chenxuan's work on this algorithm includes researching relevant techniques, coding and testing its implementation, and continuously optimizing it to improve aggregation efficiency and effectiveness.

  Chenxuan acts as the primary point of contact with external stakeholders, such as clients and advisors. Responsible for gathering requirements, providing progress updates, and ensuring that project outcomes align with stakeholder expectations. This role includes preparing and presenting project reports, facilitating Q&A sessions, and ensuring transparency in project goals and achievements.

- Tingfeng Xie

  As a project planner, I am responsible for controlling the direction of the project work in terms of project planning. In the early stages of a project, I am responsible

for defining project objectives and milestones, developing the model development plan, and establishing milestones and timelines to ensure that the project meets the requirements. During the project progress, I am responsible for monitoring the progress of the project and identifying potential schedule risks. When the actual work on the project deviated from the timeline, I immediately developed countermeasures to ensure that the project would always be able to move forward in a stable manner.

In terms of model design, as a model developer and data analyst I was involved in developing the model and functionality. I designed a post-processing function in the model that allowed our model predictions to conform to real-world logic and helped improve model performance significantly. In addition, I was responsible for testing the code, including debugging the model and fixing potential problems with the model. In addition, I am also involved in optimising the models, by analysing the model performance metrics and some key data, I have proposed a number of optimisation strategies and directions for the models.

As the communication coordinator, I act as a bridge in terms of team collaboration. I will update the progress of the project on a regular basis to make sure that the information of the project can be fully shared among the team members. At the same time, I also actively report our project progress and milestones to the tutor, and adjust the future direction of the project based on the tutor's feedback. In addition, in order to ensure that the team members can make the most of their individual abilities, I assign tasks and provide assistance to those who need it to ensure that the whole team operates efficiently.

- Yu Shi:

  Yu Shi's role as Project Research Analyst was to play a key role in the literature survey, data processing, analytical model conceptualisation and establishment of the project's research direction. Yu Shi conducted an in-depth literature survey focusing on the latest advances in multimodal and federated learning, particularly the latest applications of these techniques in healthcare prediction. Through this work, yushi provided the foundation for the project's theory and identified and clarified the gaps and barriers in the current technology, thus making an important contribution to confirming the project's research direction.

  On the data side, yushi was responsible for data collection and pre-processing. The task involved finding and screening suitable medical image and text datasets

and pre-processing them for the needs of multimodal modelling. This work ensured that the data format and quality met the model training requirements, laying the foundation for high-performance multimodal development.

Yushi was also responsible for building and optimising a multimodal model. During construction, she experimented with various frameworks to integrate medical images and text data. In addition, she optimised and tuned the multimodal model with ResNet50 and bidirectional LSTM to ensure the stability and accuracy of the model in dealing with specific medical application scenarios.

In the federated learning-building task, yushi was responsible for collecting ideas for the innovation of the aggregation algorithm. She proposed the direction of dynamically adjusting the aggregation algorithm through extensive literature surveys, which provided ideas for the development of the core technology of the project.

The above roles and responsibilities reflect Yu Shi's contribution and expertise in the project work, and she made significant efforts to ensure the project's success.

- Huiqiao Zhang

  Zhang Huiqiao served as the model framework designer, solution proposer and literature writer throughout the project, mainly engaged in data preparation, federated learning model framework construction, innovative aggregation algorithm solution research, model problem solution research and paper writing.

  In the early stage of the project, she assisted the team in finding data sets that met the project requirements, and tried to use NLP technology to preprocess the text part of the data, and proposed to extract text features by combining CNN and LSTM technology. In addition, she also contributed to the construction of multimodal models and processed text and image data sets. In the middle of the project, she built the basic framework of the federated learning model with the team leader, and conducted in-depth research on a variety of aggregation algorithms, including FedAvg, FedProx, FedMA and FedBayes, etc. By comparing the advantages and disadvantages of these algorithms, she provided ideas for the team's innovative weighted aggregation method. At the same time, the group model algorithm was compared with the FedAvg and FedProx algorithms to prove the applicability of the group's innovative algorithm. In the later stage of the project, she mainly explored solutions to network transmission problems and client computing power imbalance

in the model, laying the foundation for the team's innovative weighted differential method.

In addition, her contribution to literature research is also particularly outstanding. She has conducted in-depth research on existing federated learning models, multimodal models, network issues, computing power issues, and aggregation algorithms in the market, providing a lot of theoretical support for the writing of the paper.

- Rayne Zhu:

  Data Researcher:

  – Conducted extensive data searches to identify potential datasets for the project.
  – Selected the current dataset from multiple options, focusing on the challenges of finding multi-modal data.

  Data Analyst:

  – Contributed to writing sessions focused on data collection and analysis.
  – Performed data preprocessing, including:
    * Relabeling over 100 labels down to 14.
    * Cleaning the data to reduce irrelevant information while retaining valuable data points.
    * Ensured the integration of various modalities of data for the centralized and federated learning models.

  Machine Learning Developer:

    * Played a key role in developing the initial centralized machine learning model.
    * Utilized advanced models, specifically ResNet50 and BERT, to enhance the model's capabilities.
    * Presented the initial model by week four, achieving an accuracy of at least 65%.
    * Introduced various machine-learning techniques to optimize the performance of the centralized model, increasing accuracy from 70% to 99%.

  Project Manager:

    * Contributed to the documentation discussing the project's limitations and future work.
    * Summarized the project limitations comprehensively, incorporating feedback and questions from the client and tutor.

- Xuan Cao:

  Xuan Cao's position is code developer and data analyst. He made comprehensive contributions to this project, especially on the key aspects of code construction, model training and testing, model tuning, data visualization, and data analysis. His work throughout all stages of the project, from the initial architecture design to the final experimental analysis, played a crucial role in the success of the project.

  In terms of code construction, Xuan Cao participated in the design and implementation of the project's core code framework. He built several viable multimodal model architectures and federated learning architectures at the beginning of the project, and tested these models using existing datasets. The results of these preliminary models provide important reference and help for the direction and schedule of the project.

  Xuan Cao was responsible for the training, testing and tuning of the model. He was deeply involved in all stages of the model building, training and validation process. After deciding on the final model architecture, he refined the model code to ensure the best performance of the model. After that, Xuan Cao helped the team to implement the hyperparameter adjustment of the model and the selection of the optimizer, ensuring that the model can achieve stable performance in the existing data environment. Through multiple rounds of testing and tuning, he provided reliable support for model accuracy and training efficiency, which laid a solid foundation for the project results.

  Xuan Cao took charge of data analysis and visualization, where he analyzed the experimental results and compared the advantages and disadvantages of the new algorithm proposed by the team with the FedAvg algorithm. He made a data visualization of the training and testing results of the model. He generated a variety of graphs to visually present the performance of different algorithms and models, especially the comparison of the team's new algorithm with the traditional algorithm FedAvg. These charts helped the team more clearly understand the strengths and weaknesses of the model and provided a basis for further optimization. His analysis results and conclusions helped the team to verify the performance improvement of the new algorithm under non-independent and identically distributed data, which proved the innovation and practicality of the project in this field.

CHAPTER 6

MILESTONES SCHEDULE

Milestone	Tasks	Reporting	Date
Week-1	Review project coverage and project purpose • Define project deliverables and overall objectives • Assess the resources needed to complete the project and potential challenges	Completion of the student deed	04-8-2024
Week-2	Explore the literature review and data sets provided by the project in groups • Comprehensively review the literature review related to the project • Document initial findings and team members' personal insights	There is no client meeting this week	11-8-2024

Week-3	Develop project work plan & Client meetings • Develop a detailed project work plan and roles for each member of the team • Determine project milestones and timelines for each phase of the project • Prepare for the first meeting with the client and present our work plan	Client meeting to review the project work plan	18-8-2024
Week-4	Explore the dataset and set up the initial experimental environment • Each member of the team participates in the search for datasets that fit the project goals and explores their datasets • Prepare resources for the project and set up the environment to run the model • Document and discuss any problems encountered	There is no client meeting this week	25-8-2024

Week-5	Proposal Report Due & Establish Final Dataset & Client Meeting • Establish the final dataset through a group meeting • Prepare a detailed proposal report including project objectives, methodology and expected deliverables • Gather client feedback on the proposal • Adjust the project plan and direction of the work based on the client's suggestions	Client meeting to review the project proposal report	01-9-2024
Week-6	Build a preliminary multimodal model • Start building the multimodal model based on the final dataset • Document the performance of the initial model • Look for potential model optimisation solutions	There is no client meeting this week	08-9-2024

Week-7	Optimise the model & Client meeting • Optimise model architecture and incorporate regularisation techniques to improve model performance • Conduct multiple experiments and tests to validate performance improvements • Gathered feedback from the client on the optimisation results and formulated directions for further work	Client meeting to review our initial version of the multimodal model and client suggestions for possible future directions of our work	15-9-2024
Week-8	Continue to optimise the local model and plan to build a federated learning model environment • Introduce more regularisation techniques and analyse their performance • Collect literature review in related fields • Explore ways to build a federated learning environment	There is no client meeting this week	22-9-2024

Week-9	Progress Report Due & Implementation of the first version of the multimodal federated learning model & Client Meeting • Prepare a progress report documenting what we have accomplished, what obstacles we have encountered, why we have deviated from the timeline, and what milestones we have reached • Presenting the progress report to the client and adjusting the subsequent work plan based on client feedback • Completing the first version of a multimodal federated learning model where the aggregation algorithm uses the industry benchmark FedAvg	Client meeting to review our project progress report and get an idea of the project tasks we have completed and the progress of our work. He also reviewed the initial version of the multimodal federated learning model we developed and made suggestions for follow-up work.	29-9-2024

Week-10	Introduce additional federated learning aggregation algorithms • Read literature reviews in related fields to explore other aggregation algorithms that are feasible for the project • Successfully introduced another federated learning aggregation algorithm, FedProx • Plan to develop our team's home-grown aggregation algorithm	There is no client meeting this week	06-10-2024
Week-11	Implemented our team's own federated learning aggregation algorithm & Client meetings • Successfully developed our team's home-grown aggregation algorithm • Created a presentation to show our development results to the client • Gathered feedback from clients and prepared to draft the final report	The client meeting reviews our project development results and focuses on showcasing our team's innovative federated learning aggregation algorithm. The client is aware that our team demonstrated our innovative ideas to the client through a Q&A session, and the client gave feedback for our team's subsequent upcoming final report.	13-10-2024

Week-12	Final Presentation Integrate self-invented aggregation algorithm with industry benchmark aggregation algorithm (FedAvg) and perform performance comparison and analysis • Integrate both aggregation algorithms into the same model • Add evaluation metrics and analyse their performance differences • Prepare materials for the final project report	There is no client meeting this week	20-10-2024
Week-13	Final Report (thesis) & Client Meeting • Creating presentations and presentation videos to show the client the final results and deliverables of our team • Record the final project presentation video • Submit final project report with all deliverables and supporting materials	For the client meeting, our team created a presentation and a video of about ten minutes to show the client the final results of our team and the final deliverables. The client expressed satisfaction and provided many valuable suggestions for our final presentation.	27-10-2024

graphicx

CHAPTER 7

RESULTS

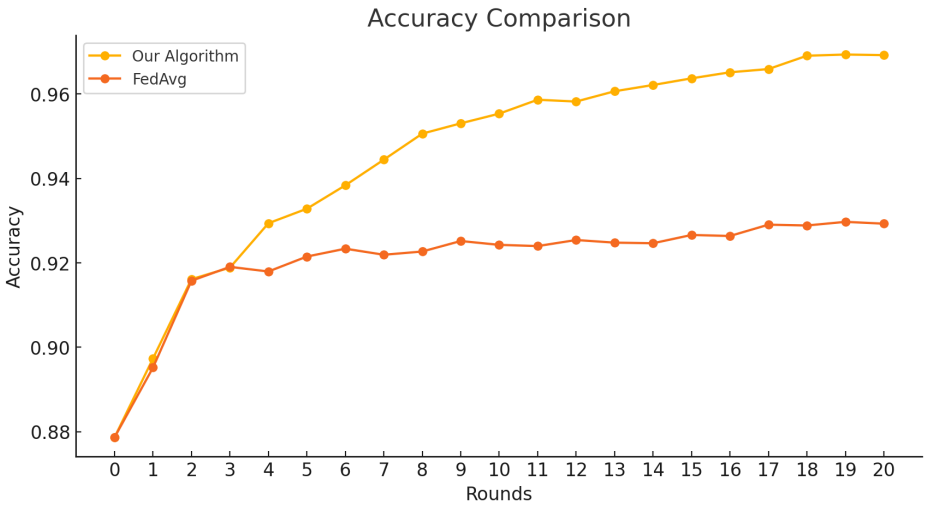

FIGURE 7.1: Validation accuracy comparison

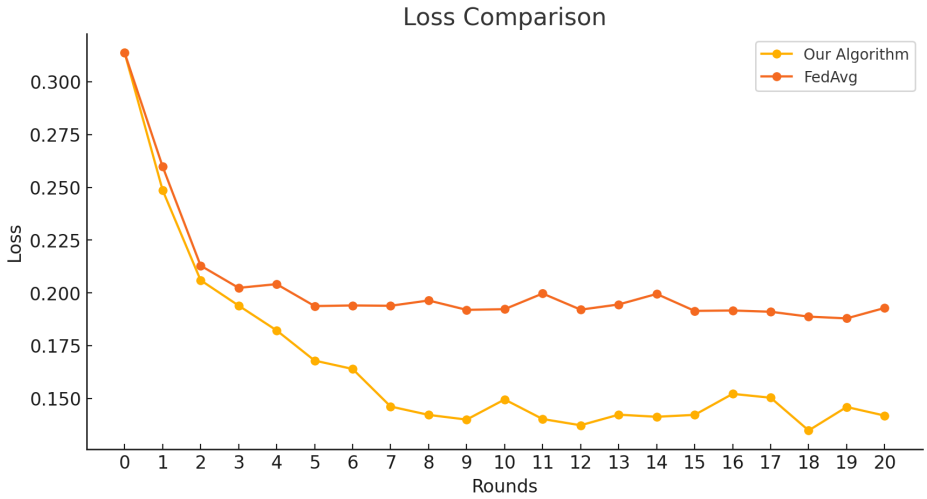

FIGURE 7.2: Validation loss comparison

Figure 7.1 and Figure 7.2 show the comparison between our algorithm and the traditional FedAvg algorithm in terms of verification accuracy from and loss. Here, we adopt the same data allocation and model structure for training for each algorithm. In terms of hyperparameter setting, we set the number of rounds to 20, training each round 10 epochs, and set the number

of clients to 3. We plot the results of the test on the untrained global model as round 0 to help compare the results of the two models.

From the two figures, we can see that our model performs better than Fedavg algorithm in terms of accuracy and loss. In terms of accuracy, our model can reach 97%, while FedAvg tends to converge at 93%. In terms of loss, our model can drop the loss to 0.14, while FedAvg only drops to 0.19. From an overall point of view, our algorithm allows the model to converge faster while achieving better performance. Figure 7.3 shows us the results of the

	Loss	Accuracy	F1 Score	Recall	Precision
Our Algorithm	0.141855	0.96921	0.86234	0.853426	0.884028
FedAvg	0.192859	0.929289	0.581716	0.566498	0.782063

FIGURE 7.3: Detailed evaluation between two algorithms

two algorithms tested on the dataset at the end of training, which include loss, accuracy, F1 score, recall and precision. It can be seen that our algorithm performs better than FedAvg in all indicators. Especially in terms of the recall rate, our algorithm reaches 0.85, while the weighted average algorithm only reaches 0.56, showing that the improved algorithm has stronger sensitivity in identifying the positive class samples (high-risk patients), and can more effectively avoid missed diagnosis. In addition, the improvement of F1 score also indicates that the algorithm has advantages in balancing precision and recall.

CHAPTER 8

DISCUSSION

The main contribution of this project is to propose a new federated aggregation algorithm that dynamically adjusts client weights based on the proportion of weight changes for each client to solve the disease prediction problem of multimodal data. Our method further optimizes the performance and convergence speed of the model by calculating the weight change of each client and dynamically allocating the aggregated weight, so that the clients with larger weight change receive higher weights. This innovative aggregation method provides an efficient and accurate solution in the integration of multi-center medical data, and improves the stability and prediction ability of the global model.

In terms of experimental results, our aggregation algorithm is better than the traditional FedAvg algorithm in terms of accuracy and stability. Our model shows stronger predictive performance and adaptability in the face of multimodal data, and especially performs more sensitivity in identifying high-risk patients, reducing the potential risk of missed diagnosis. These results show that the new algorithm has significant advantages in the medical data environment, and improves the reliability and accuracy of the federated learning framework in multi-modal data fusion.

The significance of this project is to provide a new solution for the application of federated learning of multimodal models in the medical field. Through innovative aggregation algorithms and multi-modal data fusion, our method not only effectively improves the model performance, but also realizes the collaborative processing of image and text data, which improves the comprehensiveness and accuracy of disease recognition. In addition, the aggregation algorithm is widely adaptable and applicable to other data scenarios that require cross-institutional collaboration, providing an efficient integration scheme for federated learning models in medical prediction tasks.

CHAPTER 9

LIMITATIONS AND FUTURE WORKS

9.1 Limitations

9.1.1 Quality of Data

Our project specifically applied the designed federated learning framework to a cleaned multi-modal medical dataset. The structure of this dataset must align with our federated learning model, which is focused on identifying diseases from medical multi-modal data. This requirement precludes us from utilizing standard publicly available datasets, such as CIFAR-10 and MNIST, because switching to a different dataset would necessitate substantial modifications to the code, including the centralized model, which is time-consuming. Notably, most data used for federated learning algorithm trials, like CIFAR-10 and MNIST, are single modality, consisting only of images and their corresponding labels.

While we considered using similar multi-modal datasets, the availability of cleaned medical datasets is limited. For example, although the MIMIC-CXR dataset exists, it requires significant cleaning and effort to access. Moreover, obtaining permission for access can take at least 2-3 weeks—approximately one-quarter of our project timeline. Thus, we prioritize the development of our centralized model and the enhancement of our federated learning algorithm and framework.

9.1.2 Participants

An alternative solution we explored is generating images or text from a single data modality, such as creating images from text or vice versa. However, most medical images are in black and white, and the features within these images can only be accurately identified by trained radiologists. Unfortunately, our team lacks access to professionals with expertise in medical

image detection. This absence raises concerns about the likelihood of our machine learning model successfully learning to detect these features without expert input.

9.1.3 Time Constraints

Moreover, generating data from single modalities requires a large volume of data to support effective machine learning. Our group currently lacks the computational resources necessary to manage such a large-scale project, which further limits our options within the project's timeframe. Therefore, focusing on our existing cleaned multi-modal medical dataset is essential for the success of our project.

9.2 Future Works

9.2.1 Broader Dataset Integration

Currently, our work relies on a single medical multi-modal dataset: the IU-CXR dataset. In the future, it is encouraged to explore additional datasets, such as ChestX-ray14 and the COVID-19 Radiography Database. These datasets, all originating from the medical field, can significantly enhance our study by broadening the scope of our federated learning research to include a more diverse array of chest X-ray images. This integration will help substantiate the generalizability of our findings.

By incorporating such datasets, we can train the model on a wide range of medical conditions, improving its ability to generalize across different populations and clinical contexts. This broader dataset integration is crucial for developing robust predictive models that can effectively address various health issues.

9.2.2 Multi-Modal Data Fusion

It is possible to investigate methods for combining data from different modalities, such as integrating imaging data with electronic health records (EHR) or clinical notes. This process will require collaboration with trained and professional radiologists, as their expertise is vital for accurately interpreting and integrating complex medical data. By fusing these diverse data sources, we can provide richer context for model training, allowing the federated learning

framework to leverage both structured and unstructured data for more informed predictions. This multi-modal approach will enhance the model's performance and applicability in real-world healthcare scenarios.

9.2.3 Differential Privacy

Our project utilizing federated learning is designed to address privacy concerns arising from communication between centralized data centers and hospitals, thereby protecting patient information. However, it is important to note that federated learning is not entirely foolproof in ensuring privacy. One advanced strategy to enhance privacy is the implementation of Differential Privacy, which involves adding noise to the stochastic gradient descent process. Ren (2024) suggested that combining differential privacy with federated learning could offers a robust approach to safeguarding sensitive information in collaborative machine learning environments. This approach helps to obscure individual data contributions, making it more difficult to identify specific patient information while still allowing the model to learn effectively from the aggregated data.

CHAPTER 10

Reference

Almanifi, O. R. A., Chow, C.-O., Tham, M.-L., Chuah, J. H., & Kanesan, J. (2023). Communication and Computation Efficiency in Federated Learning: A Survey. Internet of Things, 14, 100742.

Baltrusaitis, T., Ahuja, C., & Morency, L. P. (2019). Multimodal machine learning: A survey and taxonomy. IEEE Transactions on Pattern Analysis and Machine Intelligence, 41(2), 423-443.

Bernstein, J., Wang, Y.-X., Azizzadenesheli, K., & Anandkumar, A. (2018). signSGD: Compressed optimisation for non-convex problems. Proceedings of the 35th International Conference on Machine Learning, 80, 560–569.

Brisimi, T. S., Chen, R., Mela, T., Olshevsky, A., Paschalidis, I. C., & Shi, W. (2018). Federated learning of predictive models from federated electronic health records. Artificial Intelligence in Medicine, 102, 101756.

Che, S., Peng, H., Sun, L., Chen, Y., & He, L. (2021). Federated Multi-View Learning for Private Medical Data Integration and Analysis. arXiv preprint arXiv:2105.01603.

Dayan, I., Roth, H. R., Zhong, A., Harouni, A., Gentili, A., Abidin, A. Z., ... & Costa, A. B. (2021). Federated learning for predicting clinical outcomes in patients with COVID-19. Nature Medicine, 27(10), 1735-1743.

Demner-Fushman D, Kohli MD, Rosenman MB, Shooshan SE, Rodriguez L, Antani S, Thoma GR, McDonald CJ (2016) Preparing a collection of radiology examinations for distribution and retrieval. J Am Med Inform Assoc 23(2):304–310

Guan, H., Yap, P.-T., Bozoki, A., & Liu, M. (2023). Federated Learning for Medical Image Analysis: A Survey. Pattern Recognition, 110424.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.

Karimireddy, S. P., Kale, S., Mohri, M., Reddi, S., Stich, S. U., & Suresh, A. T. (2020). SCAFFOLD: Stochastic controlled averaging for federated learning. Proceedings of the 37th International Conference on Machine Learning (ICML), 119, 5132–5143.

Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., & Smith, V. (2020). Federated learning with adaptive asynchronous updates. Proceedings of the 8th International Conference on Learning Representations (ICLR).

Li, T., Sahu, A. K., Talwalkar, A., & Smith, V. (2020). Federated learning: Challenges, methods, and future directions. IEEE Signal Processing Magazine, 37(3), 50-60.

Li, T., Sahu, A. K., Talwalkar, A., & Smith, V. (2020). Federated optimization in heterogeneous networks. Proceedings of Machine Learning and Systems (MLSys), 2, 429-450.

Liu, L., Zhang, J., Song, S., & Letaief, K. B. (2023). Hierarchical federated learning with quantization: Convergence analysis and system design. IEEE Transactions on Wireless Communications, 22(1), 2–18. doi:10.1109/TWC.2022.3190512

Luo, X., Cai, H., He, Q., & Chen, J. (2020). Hierarchical federated learning: Implementing personalized and efficient collaborative learning. Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI), 4402-4408. doi:10.24963/ijcai.2020/608

Mao, Y., Zhao, Z., Yan, G., Liu, Y., Lan, T., Song, L., & Ding, W. (2021). Communication efficient federated learning with adaptive quantization. Wireless Algorithms, Systems, and Applications, 559–571. doi:10.1007/978-3-030-85928-2_44

McMahan, B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. In Artificial intelligence and statistics (pp. 1273-1282).

MIN J, WU Z D, ZHENG L, et al. Identification analysis of radio propagation links for rainfall environment based on deep learning[J]. Chinese journal of radio science2023383453-462 + 484. in Chinese. DOI: 10.12265/j.cjors.2022150

Pennisi, M., Proietto Salanitri, F., Bellitto, G., Casella, B., Aldinucci, M., Palazzo, S., & Spampinato, C. (2022). FedER: Federated Learning through Experience Replay and Privacy-Preserving Data Synthesis. arXiv preprint arXiv:2206.10048.

Ramachandram, D., & Taylor, G. W. (2017). Deep multimodal learning: A survey on recent advances and trends. IEEE Signal Processing Magazine, 34(6), 96-108.

Ren, X., Yang, S., Zhao, C., McCann, J., & Xu, Z. (2024). Belt and Brace: When Federated Learning Meets Differential Privacy. ArXiv.org. https://arxiv.org/abs/2404.18814

Rieke, N., Hancox, J., Li, W., Milletari, F., Roth, H. R., Albarqouni, S., ... & Cardoso, M. J. (2020). The future of digital health with federated learning. NPJ Digital Medicine, 3(1), 119.

Rieke, N., Hancox, J., Li, W., Milletari, F., Roth, H. R., Albarqouni, S., ... & Cardoso, M. J. (2020). The future of digital health with federated learning. NPJ Digital Medicine, 3(1), 1-7.

Sheller, M. J., Edwards, B., Reina, G. A., Martin, J., Pati, S., Kotrotsou, A., & Bakas, S. (2020). Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data. Scientific Reports, 10(1), 12598.

Wang, H., Sievert, S., Liu, S., Charles, Z., Papailiopoulos, D., & Wright, S. (2018). ATOMO: Communication-efficient learning via atomic sparsification. Advances in Neural Information Processing Systems, 31, 9871–9882.

Wang, S., Tuor, T., Salonidis, T., Leung, K. K., Makaya, C., He, T., & Chan, K. (2019). Adaptive federated learning in resource constrained edge computing systems. IEEE Journal on Selected Areas in Communications, 37(6), 1205–1221. doi:10.1109/JSAC.2019.2904348

Wang, J., Liu, Q., Liang, H., Joshi, G., & Poor, H. V. (2020). Optimizing federated learning on non-IID data with reinforcement learning. IEEE INFOCOM 2020 - IEEE Conference on Computer Communications, 1698–1707. doi:10.1109/INFOCOM41043.2020.9155495

Xie, C., Koyejo, O., & Gupta, I. (2019). Asynchronous federated optimization. Proceedings of the 36th International Conference on Machine Learning (ICML), 97, 5413-5422.

Xu, J., Glicksberg, B. S., Su, C., Walker, P., & Chen, R. (2021). Federated Learning for Personalized Healthcare Applications. Journal of Biomedical Informatics, 111, 103653.

Yang, Q., Liu, Y., Chen, T., & Tong, Y. (2019). Federated machine learning: Concept and applications. ACM Transactions on Intelligent Systems and Technology (TIST), 10(2), 1-19.

Zhao, Y., Li, M., Lai, L., Suda, N., Civin, D., & Chandra, V. (2018). Federated Learning with Non-IID Data. arXiv preprint arXiv:1806.00582.