Federated Learning Platform for Covid-19 detection

Project Plan

Ghosh Bratin 3035437692
Kwan Pok Man 3035477173
Khan Khondoker Araf Khan 3035477446

Under the humble guidance of Dr. S.M. Yiu

COMP 4801 Final Year Project 2020-21
Department of Computer Science
The University of Hong Kong
Contents

1. Background 1

  1.1 Introduction to Federated Learning 1

  1.2 Federated Learning in Covid-19 detection 2

  1.3 Three Types of Federated Learning 3

    1.3.1 Horizontal Federated Learning 3

    1.3.2 Vertical Federated Learning 3

    1.3.3 Federated Transfer Learning 4

2. Objective 5

3. Methodology 6

4. Schedule 7

5. Reference 8
1. Background

1.1 Introduction to Federated Learning

Artificial Intelligence technologies have been around for more than five decades, but advances in computing power, big data technologies and new state-of-the-art algorithms have led to major breakthroughs. In this modern day and age, AI technology is starting to show its potential in different industries, including finance, medical care and retail business.

Traditional machine learning methods need to concentrate training data in one machine or data center. However, advances in technology have made data ubiquitous. In recent years, countries have paid more attention to the privacy of data, and have formulated strict privacy provisions, making it more difficult to obtain data. Integration of data is also facing numerous obstacles, and it is difficult to share effectively, leading to the problem of isolated data islands in many fields. In order to comply with the right of privacy, it is very expensive to concentrate these data in the data center.

Under such challenges, Google proposed a brand-new concept called “Federated Learning” in 2016. It consists of a centralized server that coordinates the training activities and clients that are mainly edge devices which could count up to millions in numbers. The difference between traditional machine learning and federated learning is that data does not need to leave the client side and the client will be able to train the model locally. Connections between clients and the server are established on the cloud through a specific encryption mechanism where they will share a common global model and only model updates instead of the data are sent through the connection. The clients will first receive current global model’s weight from the server, then train it with their own local data. After the training is done, the updated parameters which are the local updates, will be sent to the server for aggregation. The server side will consolidate all the updates and produce a better global model. When the clients are idle, they will receive a new global model’s weight from the server side and the cycle continues. This not only protects privacy, but also reduces the cost of centralized transmission of large amounts of data.
1.2 Federated Learning in Covid-19 detection

AI InnoBio Limited is a Biotechnology Startup in Hong Kong that utilizes an industry-leading CMOS sensor technology to develop a novel hand-held spectrometer device. They hope to use this device with artificial intelligence to conduct saliva tests to determine in less than a second whether a certain patient is infected with the novel coronavirus. A hospital in Israeli conducted clinical trials with hundreds of patients with this new artificial intelligence-based device and is able to achieve a 95% success rate of identifying evidence of the virus in the body [1].

They hope to expand their business in Asia to allow different countries to use the device to perform a fast, accurate and low cost Covid-19 detection test. In order to obtain a better machine learning model, we need to obtain more test data across different countries. This is where the problem of data privacy comes into play, different countries and hospitals will not be willing to share the test results and therefore it is difficult to use the traditional machine learning method where the entire dataset needs to be centralized to train the model. Therefore, a platform is needed to perform federated learning with different clients to improve the accuracy of the model while keeping the data privacy intact.

1.3 Three Types of Federated Learning

Federated Learning is divided into horizontal federated learning, vertical federated learning, and federated transfer learning according to the type of data.

1.3.1 Horizontal Federated Learning

Horizontal federated learning is used when participants have datasets with the same feature space, but different samples. First, participants train the model locally, and then send the encrypted result to the server. Then the server aggregates the result securely and sends it back to the participants. The participants then update their own model with the decrypted result [2].
1.3.2 Vertical Federated Learning

Vertical federated learning is used when participants have datasets with overlapping samples, but different feature spaces. In this case a trusted third party collaborator is involved. First the participants confirm the common samples. Then the collaborator creates encryption pairs and sends public keys to participants. Participants encrypt and exchange the intermediate gradient and loss results, then compute encrypted gradients and add additional masks, and send the encrypted values to collaborator. The collaborator decrypts and sends the decrypted gradients and loss back to participants, and participants update their model accordingly [2].

1.3.3 Federated Transfer Learning

Federated transfer learning is used when participants have datasets that are different in both feature space and samples. The process is very similar to vertical federated learning, just the gradient computation and intermediate results exchanged between participants are different [2].
2. Objective

The main objective of this project is to build a federated learning platform where clients can train a machine learning model together on a single platform. The platform should ideally be able to handle Horizontal Federated Learning.

Subsequently, the platform will build on top of covid-19 detection machine learning model provided by AI InnoBio. However, since the platform is highly dependent on the model itself. If the company is not able to provide a model with enough data to work on in time, another similar machine learning model with sufficient data will be chosen to develop the platform instead. Once the model provided by the company is mature, we hope to eventually migrate that model into the platform.

After the platform is built, it will be evaluated using three criterias - accuracy, efficiency and privacy.

1. **Accuracy** - comparing the accuracy of the final model to the ideal case
2. **Efficiency** - speed of building the models given an increase in users
3. **Privacy** - assess the security of data and the user nodes

Comparisons and improvements will be researched and made according to the above three criterias.

When the planform can handle horizontal Federated Learning with satisfying results, we hope to use the remaining time to improve and modify the platform so it can also handle vertical learning. This feature will be implemented if the remaining time allows.
3. Methodology

In order to develop a federated machine learning platform, a significant amount of research needs to be conducted on a plethora of machine learning classification algorithms simultaneously on Google’s server side machine learning tools.

Following that, a simple classification algorithm, e.g. linear regression, SVM, etc. will be picked. Once we are certain about the classification algorithm, then we can move on to develop the specifications of the federated learning platform, e.g. aggregation protocol, server-client security, etc.

Subsequently, the server and client side of the application will be built using Node.js hosted on the cloud. The server should ideally be able to handle simultaneous requests from several clients and respond to them.

Once the platform is built, we will be comparing the accuracy, efficiency and privacy compared to traditional machine learning methods. We can then discuss and brainstorm different methods to improve the platform, if need be.

Upon completion of the whole cycle of making a platform for a given model using horizontal federated learning, we will move on to vertical federated learning if possible.
# 4. Schedule

<table>
<thead>
<tr>
<th>Date</th>
<th>Scheduled Work</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/10 - 10/10</td>
<td>Background research</td>
</tr>
<tr>
<td>11/10</td>
<td>Submission of project plan and project website</td>
</tr>
<tr>
<td>12/10 - 31/10</td>
<td>Inspect and analysis machine learning model to be used</td>
</tr>
<tr>
<td>1/11 - 30/12</td>
<td>Build a platform for Horizontal Federated Learning</td>
</tr>
<tr>
<td></td>
<td>1. Establish a cloud service for server and client communication</td>
</tr>
<tr>
<td></td>
<td>2. Figure out what should be sent for model updates between local model and global model</td>
</tr>
<tr>
<td></td>
<td>3. Incorporate model updates communication into cloud service between server and client</td>
</tr>
<tr>
<td></td>
<td>4. Testing of Horizontal Federated Learning on the platform</td>
</tr>
<tr>
<td>11/1 - 15/1</td>
<td>First presentation</td>
</tr>
<tr>
<td>24/1</td>
<td>Submission of interim report and preliminary implementation</td>
</tr>
<tr>
<td>25/1 - 28/2</td>
<td>Tuning of the platform to improve accuracy, efficiency and privacy</td>
</tr>
<tr>
<td>1/3 - 31/3</td>
<td>Implement Vertical Federated Learning into the platform if possible</td>
</tr>
<tr>
<td>18/4</td>
<td>Submission of final report and implementation</td>
</tr>
<tr>
<td>19/4 - 23/4</td>
<td>Final presentation</td>
</tr>
<tr>
<td>4/5</td>
<td>Project exhibition</td>
</tr>
</tbody>
</table>
5. Reference

