APMRS – A Framework to Ensure the Secure Transfer of Covid-19 Data for AI Embedded Analysis and Prediction

Weiting Li
Virginia Polytechnic Institute and State University
Blacksburg, Virginia, USA
weitil6@vt.edu

Jiayue Zhou
Virginia Polytechnic Institute and State University
Blacksburg, Virginia, USA
jiayuez@vt.edu
Group:WhoKnowsTheName

Drew H Klaubert
Virginia Polytechnic Institute and State University
Blacksburg, Virginia, USA
kdrew17@vt.edu

Abstract

The turn of the 21st century has seen an explosion in the amount of data created by new and emerging technologies. Everyday enormous data were collected from every single one of us by these big companies like google, meta to help make their applications more user friendly and in the end make our lives better. On the other hand, people may have unconsciously allowed companies to "steal" information from them. It seems to be a dilemma for users: choose privacy versus convenience? Therefore, privacy of data is a trending topic today in our modern society. In this paper, we incorporate this issue with one of the biggest crisis in human history; the Covid-19 global pandemic, a corona-Virus that spread all over he world in just a couple of months. In our project, we aim not only to create a highly efficient data sharing system that is simple and trustworthy, but also able to provide help and guidance to doctors in need to make informed decisions quickly. Therefore, to help protect patients' privacy, we proposed a AI Patients Medical Records System(APMRS) to use a novel technology that is created to aid in solving this dilemma, the Blockchain technology, a shared, immutable ledger system that facilitates the process of recording transactions and tracking assets in a public accessible network. In addition to that,we also propose to incorporate our framework machine learning algorithms to help our doctors with better and faster decisions to treat a specific patient. In conclusion, our framework consist of a database system that collects covid19-cored data and also possess the functionality to help doctors to respond more efficiently and save time from analyzing the raw data by applying machine learning models. Possibly after the pandemic end, future researchers could consider broaden our project to apply to other medical disease such as cancer etc. The source code of APMRS can be found on https://github.com/Jiayue-Zhou/Patients-Medical-Records-System.

ACM Reference Format:
Weiting Li, Jiayue Zhou, and Drew H Klaubert. 2021. APMRS – A Framework to Ensure the Secure Transfer of Covid-19 Data for AI Embedded Analysis and Prediction. In *Proceedings of Blockchain Technologies.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/1122445.1122456

1 Introduction

We have learned an awful lot having lived through the era of Covid-19. Things like resilience and dealing with uncertainty and rapidly changing scenarios. Uncertainty is something that will forever make humans feel uncomfortable. There are many things that make us feel uncertain. Many things change far too fast for us to ever be able to become comfortable. The pandemic situation has put us in a position that requires that we be alright with such rapidly changing factors in our lives.

Luckily, as quickly as times are changing and problems are facing us, humans are responding in an all time quick manner. It is no secret that data plays a major role in the type of decision making that responds quickly to historically large roadblocks for humans. Since data driven solutions have been at the forefront of technological advancements, the new problem has now turned to data integrity. There has been at least one time during the Covid-19 pandemic that most of us were a little skeptical of some form of data or another that was reported to us. We put a lot of trust in these centralized entities that are reporting the information that is affecting our daily lives.

The rise of Blockchain technology has been able to address some of these aspects of trust with its decentralized and immutable nature. Blockchain is a a shared, immutable ledger system that facilitates the process of recording transactions and tracking assets in a business network. An asset can be tangible (a house, car, cash, land) or intangible (intellectual property, patents, copyrights, branding). It was first designed

Blockchain Technologies, May, Blacksburg, VA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/1122445.1122456

by two mathematicians Stuart Haber and W.Scott Stornetta in 1991. They wanted to introduce a computationally practical solution for time-stamping digital documents so they could not be tampered. Then it finally has the potential to become the foundation of future record-keeping system only since a decade ago. In 2010, an anonymous programmer whose pseudonym is Satoshi Nakamoto designed and programmed bitcoin. To summarize, Blockchain technology has these benefits: (1) Better transparency; (2) Reduced cost; (3) Enhanced security; (4) True traceability; (5) Improved speed and efficiency. In health care area, there are many applications in the field already or can be explored in the future, such as:

1. Give patients more control over their data
2. improve accuracy and security of Electronic Health Record(EHR)
3. make healthcare services more accessible
4. reduce health care costs

Generally speaking, Blockchain technology can facilitate national-wide interoperability of electric health records which allow providers access to patient's history, medications, images etc. According to a study, full interoperability could possibly save the US healthcare system $77.8 billions per year.

In the scenario of covid-19, as we all know, dealing with a infectious disease that both spread and mutate so quickly, it is inevitable and also paramount that doctors and experts need to collect data rapidly and respond to the crisis in the same manner.

Coronavirus disease 2019 (covid-19) is a contagious disease caused by the severe acute respiratory syndrome coronavirus. The first known case was identified in Wuhan, China. As of today, the whole globe has identified 517 millions cases and 6.2 million cases of death. The virus is known for its fast spread and high false-negative possibility. In the beginning of the pandemic when we had little to no knowledge about this virus, human race were at stake because we could not manage such a number of patient and their data that are both exponentially growing. And it is almost impossible to integrate them. As our data were taken to further study and fight against the virus, people cannot help bring up their concerns of integrity and privacy of the data that was collected.

Therefore, blockchain is the perfect solution because it is a decentralized, immutable system with public accessibility. By using blockchain as our database, we can protect privacy of the patient but also be able to provide Covid-19 related data to the doctors that are allowed to access it to make accurate decisions with the treatment without any geographical limitations.

In this paper, we propose a framework that consists of a blockchain database to which patients can self-report symptoms or doctors can collect and save medical information, without any worries about their privacy violated by third companies or hackers. In addition, this framework is also be able to streamline the outputs from machine learning algorithms so medical officials with access to the Blockchain are able to respond to real time data and make informed decisions. We think our framework is going to save time for the doctors to diagnose and assign specific treatment. In addition, it could increase doctors' decision's accuracy and also provide extra privacy protection to the patients.

2 Related Work

Wang et al. [4] presented a framework combining the decentralized storage approach, the Ethereum blockchain, and ABE technology. In this paper, the general idea for distributing secret key for data users is proposed. While it is not for any specific using, it did give out the basic idea for blockchain-base data collection and using. This is a great work for discussing using the blockchain as a way to store lots of data however it does not have a very nice way to streamline the actual use of the data that is stored. Since we able to make the data more readable by many people who may not have much experience in accessing data from any sort of database, this information is able to reach a wider audience.

Liang et al. [2] discussed an innovative user-centric health data sharing sultion by blockchain to protect privacy. And in this paper, a mobile application is deployed to collect health data from personal wearable devices. However, they did not incorporate medical data into the system, which is a crucial factor when doctors make a decision. The useful part of this work is the fact that it is a system using wearable technology which is designed for the ease of the user. We aim to create functionality that differs from wearable devices and is able to compile hospital records and other self reported health data. This is more of manual entry of data but this is able to make the data more rich in information than just simple measurements.

Fan et al. [1] proposed a blockchain-base information management system MedBlock, and used it to handle patients' information. In this paper, their product is more efficient in accessing and retrieving the EMR of patients. And it also decreases the energy consumption and network congestion. However, we believe that it is too costly to have two blockchain databases to make this product work. We like this work's ability to be able to take data for a single patient from multiple hospitals. This is what we plan to have in our system as a blockchain application is able to compile data from different sources in such a way that is does not breach the privacy of patients.

Marbouh et al. [3] proposed, implemented, and evaluate a blockchain-based system using Ethereum smart contracts and oracles to track reported data related to the number of new cases, deaths and recovered cases. In this paper, the authors associated blockchain and its related properties with

current pandemic, COVID-19. The paper made great advances in being able to collect large amounts of data and report basic information. This is what we would also like to do in our basic framework. However, they did not have personal information for medical purposes. Their work focuses heavily on reporting totals of infections, recoveries etc. of an entire population but does not include other personal health data to be able individualize the cases for each person. We like the use case of covid-19 for this work. We plan to also have Covid-19 be our primary use case to demonstrate how this framework can be very fast in aiding in important decisions.

In this paper, we propose a blockchain-base system which generally includes three parts and aims to have both personal health data and medical data to facilitate doctors diagnosis. Moreover, our framework implements an application uploading data and downloading data, simplifying the process of collecting and distributing information. Also, doctors are able to view all the data by charts, histograms and more approaches as visualization. This mechanism is aiming to improve the efficiency and the accuracy for a doctor's diagnosis along with ensuring the integrity of the data.

The main idea that we propose that the literature survey does not hit on is the idea of the extra layer of data analysis that can be done with the data collected from the Blockchain. Being able to have a streamlined way to provide summary statistics and data visualizations is extremely valuable for doctors and regular patients to make quick informed decisions. This is what will set our work apart from the related work. We will be aiming to make this system something that inexperienced people can use in order to make sure they are getting correct information. Making the system less complex may seem like it will reduce the efficacy of the system, but we believe it will help in achieving our goal of usability.

3 APMRS Design and Implementation

Figure 1 shows the basic access control of APMRS. Only doctors are able to upload data to the blockchain and both patients and doctors are able to use APMRS to get predicted results.

Our approach for building the system can be broken down into four main components:

3.1 Approach Overview

1. **The Data**
   - Collected from online resources
2. **The Blockchain**
   - By using solidity and Ethereum smart contracts to create our blockchain
   - Scrapper: To bridge the data from the blockchain to perform data analysis and ML algorithms
3. **The web page to facilitate users**
   - To connect the doctors and the patients to the blockchain
4. **The machine learning model**
   - Utilize the data to perform ML algorithms, e.g. predictions of covid 19 test outcomes as guidance for the doctors

As below, we discuss these three aspects of the project in greater detail.

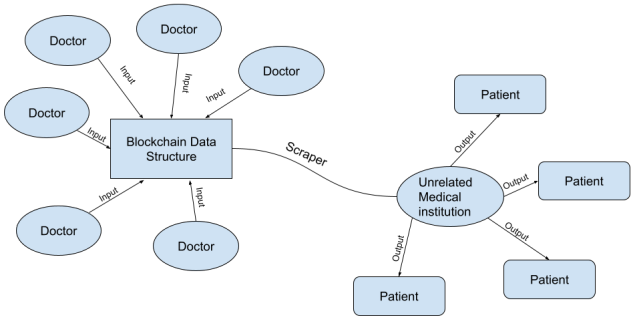


Figure 1. Diagram of the flow of data and users when using the system

3.2 The Data

If we refer back to the system architecture, we can recall that the system we are creating allow doctors to add patients visit to the medical facility as data on the blockchain. This means that in order to create a working prototype, we need to be able to populate our blockchain with some data. We had a couple of options as to where this data could come from:

1. Finding an online dataset that gives some medical data about covid-19 and truncate it to make it a small enough scale to fit within our CPU resources
2. Making up data ourselves since this prototyoe will be only on a small scale

After some thought, we found that it would be better for our ease of creating this system if we were to find a dataset from a third party source and truncate it down drastically so we can add a few of the datapoints to our blockchain for demonstration of the system. This way, if for some reason we need even more data to be able to use, we can just pull more samples off the dataset and not have to spend more time creating our own data which can prove to be tedious.

We aim to implement a adaptive machine learning model that evolves through time and changes as more data is inputted into the blockchain database. For our one semester project, we narrowed our target down to just Covid-19 crisis. However, we do not want to use our model towards the general statistics. Our goal is to produce a effective tool for

the doctors and the hospitals to work with. And here comes our first challenge: Public covid data is basically everywhere, but personal test data which we are looking for is very hard to get, It is easy to think it through because these medical data should be protected and only between a patient and his doctor.

When doing our research in choosing which data we should enact, a few datasets came before our eyes. But in the end, we choose the " sample lab results in Early 2020 to detect Coronavirus " as our final dataset, taken from the Kaggle website which is an online community of data scientists and has lots of content as far as sample data is concerned. We made our decision based on the fact that the data is more objective and privacy-protection needed. We believe that the more objective the data is, the more useful our model can be in real life scenarios. And we think it is only reasonable to embrace the importance of blockchain when the data is privacy sensitive. Therefore, as shown in the table of the variables in our data, eventually after pre-processing our data consists of 6 categorical variables(e.g. different disease detection & target variable). And the rest of the variables are numerical representing the various measurements of vitals we can obtain from blood samples.

Column Name	Description
Covid19_res	Binary 0 or 1 for postive and negative respectively
Hemoglobin	Float, hemoglobin levels
Platelets	float, platelet measurement‘
MPV	float, MPV measurement
RBC	float, RBC measurement
lymphocytes	float lymphocytes measurement
MCHC	float, MCHC measurement
leukocytes	float, leukocytes measurement
basophils	float, basophils measurement
MCH	float, MCH measurement
eosinophils	float, eosinophils measurement
MCV	float, MCV measurement
monocytes	float, monocytes
RDW	float, RDW measurement
detection_coronaviridae	indicates the detection of condition 0 for negative 1 for positive
detection_orthomyxoviridae	indicates the detection of condition 0 for negative 1 for positive
detection_paramyxoviridae	indicates the detection of condition 0 for negative 1 for positive
detection_picornaviridae	indicates the detection of condition 0 for negative 1 for positive
detection_pneumoviridae	indicates the detection of condition 0 for negative 1 for positive

Sheet 1. All of the variables in the chosen sample data and their corresponding brief description.

The raw data has 5644 rows, 110 discrete variables and a bunch of categorical variables as well. However, 90% of the rows are missing at least one variable. And 60 % of them have over 90% of the columns missing.So the first challenge here is to pre-processing the raw data into a dataset that is ready to be trained with our machine learning model. To achieve this, we did:

1. feature engineering
2. deleting rows with too many missing values
3. data imputation

3.2.1 Feature engineering. To evaluate the correlation between the variables and also their contributions to our

target variable: covid-19 test result, We used the SNS package from the python library to do a pair-plot. If there is no obvious variance, we get rid of them.

3.2.2 Deleting rows with too many missing values. We evaluated the number of rows that have 50% more data missing. As a result, out of 5644 rows, 5112 rows satisfy this argument and we delete them accordingly.

3.2.3 Data imputation. After the procedures above, we end up having 534 rows and 19 different columns(excluding the target variable: covid test result). For binary variables, the value is 1 to represent true and 0 to represent false. Most of the data are floats to represent measurements that can be taken at any routine hospital visit vie a blood sample etc. All of the information about the variables is featured in table 1 at the start of the next page.

However, to use this dataset in a machine learning is still not enough, because it is way too small, in addition to its imbalanced class —- number of positive cases is only 10% of the negative cases. It is hard for our model to effectively learn the decision boundary. Thus, we need to perform **data imputation** to improve our model's performance later. To do this, we enact the SMOTE package in python which is a oversampling technique.

SMOTE refers to "Synthetic Minority Oversampling technique". SMOTE first selects a minority class instance a at random and finds its k nearest minority class neighbors. The synthetic instance is then created by choosing one of the k nearest neighbors b at random and connecting a and b to form a line segment in the feature space. The synthetic instances are generated as a convex combination of the two chosen instances a and b. In the end, we successfully increased our positive and negative ratio to be **0.5: 1**.

3.2.4 How data work with the blockchain. Now that we have explained the data itself and why we need it, we need to clarify how this is going to interact with the blockchain. As we previously mentioned, each row in the data will be representing a visit to a medical institution whether this be any ordinary doctors office or a hospital visit. Which means that each row in this data will be taking up a block in the blockchain. Each of the blocks will be put together when the doctor obtains the information from a visiting patient. The information in the block will be the attributes that are laid out in the dataset. This is not too much different of a concept than if we were to store traditional transaction data in the blockchain. It is very much similar except instead of use recording the sender and the receiver, we will be able to record the attributes needed to add a row to the dataset.

3.3 Thoughts towards Blockchain

The first step in this process for streamlining the use of medical data is for us to create a Blockchain structure. We originally planned on constructing this Blockchain in Java due to the fact that we all have experience with the language and would be able to interact with it easily. We eventually decided that it would be best for use to use smart contracts and the Ethereum blockchain. To be clear, Ethereum is a decentralized open source blockchain with smart contract functionality. This blockchain is one of the most popular ones in the world and is second in popularity to only bitcoin. Where ethereum takes an advantage over bitcoin is in the programmability. This means that we can store more than just transaction sender-receiver data and expand the application to what we are looking for. We have more complex data that has more than just the fields that would be required for transaction data.

As mentioned we create a framework very similar to the basic distributed ledger systems that we have discussed in class. A blockchain that is used for transaction data can be adapted into one that reports medical data. An example of such a Blockchain is shown below:

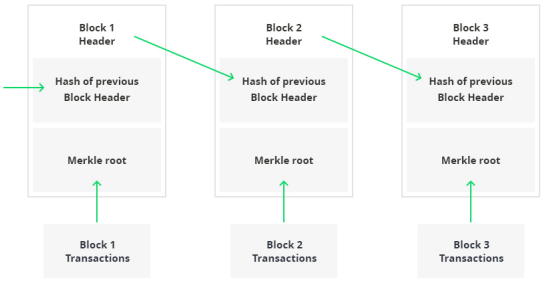


Figure 2. Example of simple structure of Blockchain to be implemented

Figure 1 shows the basic idea of how blocks are chained together to make this data structure. Each block contains its own header, the cryptographic hash of the previous block and a timestamp. Our chain structure itself does not deviate much at all from any other traditional blockchain structure. The general structure of the blocks does not change much either. The only difference in our blocks is that instead of holding data about transactions, it now holds medical records. The data still looks like transaction data when putting it together in the blockchain. The idea is that we want to send medical information over the blockchain, so this transfer is still like a transaction of currency at its core.

As mentioned in the previous paragraph, we stores medical data fields in the blocks of our blockchain. Usual transaction data usually has a format of <Sender, Name> and <Recipient, Name>. We will be using a similar tactic of storing data in those fields. We will show what the different blocks of our blockchain could look like.

```
{'index': 1,
 'transactions': [{'covid19_Res': 0,
   'hemoglobin': 0.056260044,
   'platelets': -0.457015881,
   'MPV': 0.015419052,
   'RBC': 0.144051816,
   'lymphocytes': 0.388455753,
   'MCHC': -0.903608342,
   'leukocytes': -0.132505017,
   'basophils': -0.185302487,
   'MCH': -0.241050077,
   'eosinophils': 1.388798821,
   'MCV': 0.195766034,
   'monocytes': 0.312408973,
   'RDW': -0.675847006,
   'detection_coronaviridae': 0,
   'detection_orthomyxoviridae': 0,
   'detection_paramyxoviridae': 0,
   'detection_picornaviridae': 1,
   'detection_pneumoviridae': 0}],
 'timestamp': 1651632027.4501476,
 'previous_hash': '1ee577fc39476660e7864532f841a01bf88f49261a3d61167b2b5322f1c733b1',
 'nonce': 481,
 'hash': '0028234d602a78b92312f483068d74433eb0ee0425c9ae8b7b527c0e7424b039'}
```


Figure 3. Sample block Display

3.4 The Implementation of Blockchain Database

We have a high-level design of blockchain based on Solidity. Solidity is a high-level and object-oriented language implementing smart contract on Ethereum Virtual Machine(EVM). Ethereum is a decentralized open source blockchain with smart contract functionality. It is one of the most popular blockchains in the world. Solidity has many good features such as simplicity of the syntax and various very supportive functionalities. For the back-end concerning about blockchain, we use Solidity to build our blockchain part on Ehtereum. And moreover, we have built a scrapper to extract data from blockchain. As we know, Ethereum nodes make the blockchain and the data in blockchain immutable. And the public-private key mechanism helps with the privacy of our data. With EVM, our system runs based on Ethereum and our smart contract is able to interact worldwide.

All the features we use for building APMRS is displayed on Sheet 1. In practice, we aggregate all the features together as a string, as Figure 4. Delimitered by hash signs, we can easily encode and decode with the data downloading from the blockchain.

```
1    struct People {
2        uint256 id;
3        uint8 covid19_Res;
4        string aggregation;
5    }
```

Figure 4. People Structure in the Solidity Code

In this smart contract, we have several functions. They run together have people to store and download data from the Ethereum. In the smart contract, People is structed with id, covid-19 result, and aggregation. We have two mapping data structures, one is id-map-people, and the other is private-password-map-people. The id is a inner index for people while the private password will be displayed once the record is created. And the private password will be displayed only once then it will be deleted forever from the blockchain so that to keep it safe and private. The users are responsible to keep it in a safe place and they are able to use the private password to query their records in the future. The private password is a 16 bits number string.

3.5 The Front-end Design

As a system easy to use for both doctors and patients, we have a web page to simplify their work to interact with the blockchain.

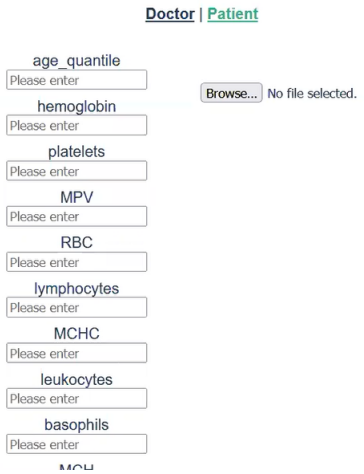


Figure 5. The patient Portal

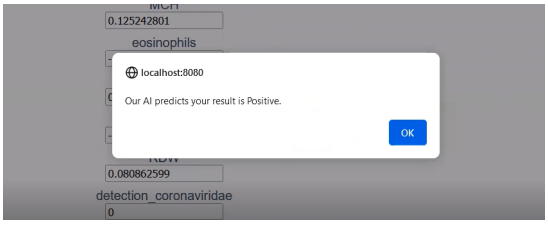


Figure 6. The predicted result

As the Figure 6 shows, patients are able to use their own data to get a prediction from APMRS.

For doctors' portal as Figure 7, in addition to have a predicted result from APMRS to help them diagnose, they are able to use two more features. One is uploading a data point to the blockchain with doctors' authoritative test result, so that the data point could be used in the future to train the inner-system machine learning model. The other feature is a query function, which is able to retrieve a data point from the blockchain by the private password.

Besides, as Figure 7 shows, APMRS supports MetaMask, which is easy for users to use to monitor and manage their accounts.

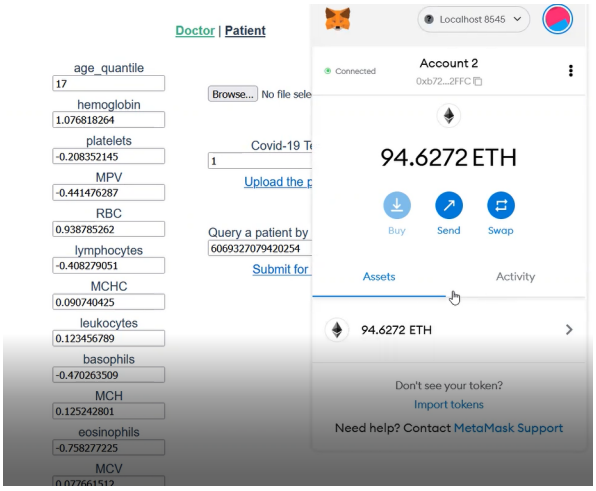


Figure 7. The Doctor result

The front-end is created and ran by Vue.js and NPM. It has two views for doctors' portal and patients' portal.

To facilitate users, we designed a tool for uploading csv file. The user of APMRS is free to choose whether use this tool to fill the text boxes.

3.6 The Back-end Design

The back-end of APMRS includes a smart contract, and all Python programs to deploy the contract, upload data and download data from the blockchain. Besides, there is an inner machine learning model accepting the whole dataset from the blockchain for training, and it will predict its covid-19 result.

The data workflow from Ethereum to back-end of APMRS is implemented by Web3. And the interaction between the front-end and the back-end is supported by Flask, a light-weight framework of Python.

3.6.1 Importing existing data. This function is for those data points waiting to import to the blockchain in advance. In practice, we need some base data for the machine learning model, otherwise it will not run due to the loss of data points.

3.6.2 Get predicted result. This function is for both doctors and patients to use. It accepts a data point, and at the same time, it would download the whole dataset from the blockchain, and uses the dataset to train the machine learning model. Then the machine learning model would get the new input data point and gives the predicted result.

3.6.3 Upload one data point. This function is for doctors to upload a data point to the blockchain. Once having the authoritative result, doctors are able to input it into a text box on the front-end with all other data features. Then this function will deal with the data point and store it in the blockchain as part of a tranaction.

3.6.4 Query function. This function is for user to query a data point from the blockchain. Once a data point is uploaded to the blockchain, the blockchain will display a permanent private password for this user. The user is responsible to remember the code and is able to use the code to retrieve the data they have uploaded to the blockchain.

3.7 Supervised Learning Model

To begin with, we need to find a valid machine learning model. Because "covid res" is a categorical value and is already classed, we need to find a supervised learning method to deal with this classification problem. We tentatively picked a few popular classifiers such as the decision tree, KNN, Support Vector Machine. In the end, we decide to go with support vector machine as our final model. There are a few nice merits of choosing SVM:

1. SVM works best when there is a good separation between classes.
2. Effective with high dimensional data(which medical daya usually is).
3. It is memory efficient compared with KNN, decision tree.
4. It works well when data size is small(which our data is).

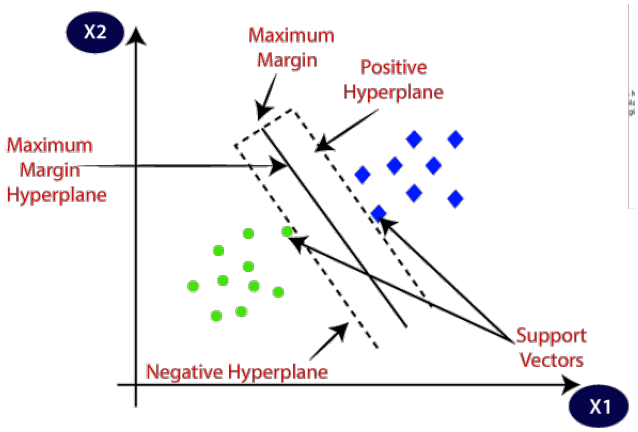


Figure 8. An example of support vector machine for classification

This particular algorithm is easy to implement as it is the base for other more complex algorithms so there is plenty of resource for using this at a small scale. For our purpose, we are only going to have a little over 500 rows to our data so training time is not our concern. The framework at the state we have it pulls the data off the blockchain when it is prompted and then trains the model then and there. While this would work with less data, once many points begin being added to the blockchain, this training time will be increasing a lot and this will become and impractical process. Our hope is that another step of this would be to implement a process

in which the model gets trained in real time and is able to be trained on new data as it is added so we do not train it multiple times. That is beyond the scope of this project for this semester and this will also be mentioned in the future work section of this report.

3.8 Evaluation

To evaluate the effectiveness of our model in predicting, we enacted two main metrics:

1. ROC AUC Score
2. Weighted F-1 Score

3.8.1 ROC AUC Score. ROC Curve is an abbreviation of receiver operating characteristic curve. It is a graph showing the performance of a classification model at all classification thresholds. This curve plots two parameters:
(I) True Positive Rate —— $TPR = \frac{TP}{TP+FN}$;
(II) False Positive Rate —— $FPR = \frac{FP}{FP+TN}$

True Positive Rate is used to measure the percentage of actual positives which are correctly identified. And false positive rate is the proportion of true negatives that are misclassified as positives.

AUC here stands for area under the ROC curve. ROC AUC score simply means magnitude of the area under the ROC curve. AUC provides an aggregate measure of performance across all possible classification thresholds. One way of interpreting AUC is as the probability that the model ranks a random positive example more highly than a random negative example. For example, the best case is that for a random classfier, its ROC AUC score is 1 which is perfect.

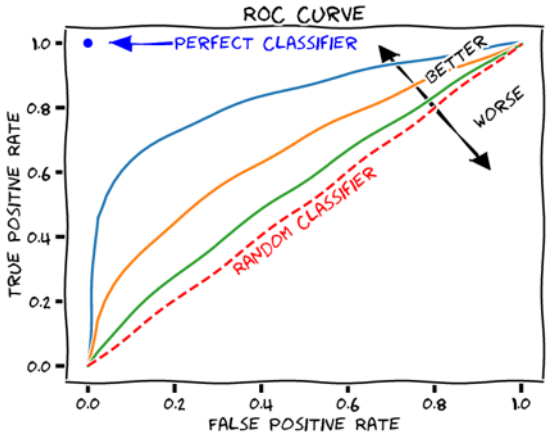


Figure 9. An illustration of ROC AUC curve

3.8.2 Weighted F-1 Score. Firstly, we need to understand the concepts of the basic terms in calculating F-1 score: **Recall and Precision**.

$Recall = \frac{TP}{TP+FN}$
$Precision = \frac{TP}{TP+FP}$

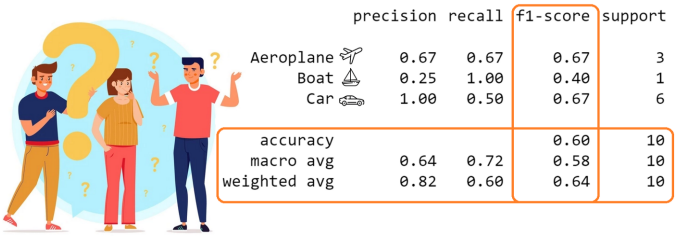
Formula for F-1 Score:

F-1 Score = $\frac{2*Precision*Recall}{Precision+Recall}$

Definition: Harmonic mean of precision and recall for a more balanced summarization of model performance.

To evaluate model performance comprehensively, we should examine both precision and recall. The F1 score serves as a helpful metric that considers both of them. However, because our dataset is still imbalanced even after data processing, we consider weighted average F-1 score to be the one to go.

The weighted-averaged F1 score is calculated by taking the mean of all per-class F1 scores while considering each class's support.Support refers to the number of actual occurrences of the class in the dataset. With weighted averaging, the output average would have accounted for the contribution of each class as weighted by the number of examples of that given class.



	precision	recall	f1-score	support
Aeroplane	0.67	0.67	0.67	3
Boat	0.25	1.00	0.40	1
Car	1.00	0.50	0.67	6
accuracy			0.60	10
macro avg	0.64	0.72	0.58	10
weighted avg	0.82	0.60	0.64	10

Figure 10. An illustration of weighted average F-1 score

3.9 Result

After 5 fold cross validation, we obtained these outputs for our SVM model:

```
ROC AUC score for the combined sampling method: 0.905
```

Figure 11. The ROC AUC score for SVM

For ROC AUC curve, our svm model achieved a decent score of 90.5%. Although we believe there is space of improvement, due to the nature of our imbalanced data, we still consider this as a good score.

For Weighted Average F-1 score, our model obtained a score of 94%. In our another attempt of using random forest algorithm, we achieved a F-1 score of 91%. So SVM is still a better option in this case compared to random forest.

	precision	recall	f1-score	support
0	0.97	0.97	0.97	95
1	0.75	0.75	0.75	12
accuracy			0.94	107
macro avg	0.86	0.86	0.86	107
weighted avg	0.94	0.94	0.94	107

Figure 12. The weighted average Score for SVM

4 Future Work

While we feel that we have done a rather comprehensive job of laying out this framework, there is still more work that can be done. The good thing about this particular work is that it will be always changing and there will always be able to be new adaptations of this framework and models that can be used with it.

One of the first things that we think should be done in the future is making it a full scale system. Is to launch it on a full scale blockchain that is being contributed to by multiple nodes in an actual network. Moreover, there are two other aspects of what we meant by "Full scale":

1. it needs to be effortless or costless
2. it needs multiple health institute to upload more data

It does not make sense if working with our system is going to be more costy than a centralized system, because our goal is to make this system distributed so that everyone with authorization has access and can basically log into without geographical limitations, and also cheap to work with: the cost should be almost zero in ideal cases except for the cost for using Ethereum. In our system which ran in a VM environment, we cost about 20 Ethers to make our system fully functional. And the current price of a unit ether is around 2500 dollars in real life. And due to lots of limitations, we failed to imitate the process of uploading the data or use the tool pretending as a health official. Add for this project, because of the sensitivity of the data, we do not have enough real life data to work with. If we can feed more data to our model, we believe we can definitely achieve a better classification. In the future research of continuing this project, I believe we can explore more about these options.

Also, for system design and implementation, there is still something that is able to improve. Currently, our application has a basic web page, and this web page could be elaborated; more and more decorative tools could be utilized. What's more, for the real-world using, APMRS is supposed to deploy in a server with a public URL to be used by users.

APMRS offers a pattern for people to do further change, too. APMPS has some flexibility to revise so that it can apply to a different background with a different dataset. However, some parts of the implementation of APMRS are not very friendly to change. Future researchers may be possible to refactor the code to make the coupling of the code looser. Some patterns may be extracted from the code and make those patterns revisable. It will make APMRS fit more general use.

5 Final Conclusion

In this report, we have given a comprehensive review of what we have accomplished throughout this semester. We have been able to successfully create a framework that is able to ensure a safer transfer of covid-19 data between doctors and medical insitutions. This is a great piece of work that has a good place within the medical field. We are able to take advantage of the properties of data that is on the blockchain. The main property is that we are ensuring the integrity and the accuracy of the data. Another property that we really like is that there now is not the flaw of the single point of failure that tends to plague most centralized systems.

The system overall is a very novel idea. The parts of the system that are helping us to make it work are not necessarily novel since we are using some already existing smart contracts and the ethereum blockchain and some machine learning algorithms that are already implemented for us in python. The overall system architecure with exploits the communication of the blockchain and the machine learning algorithm is the novel idea that we think has serious potential.

Another thing that we conclude is that this overall system framework can be expanded into many different scenarios. There may be different fields that could use this system other than just the medical field. If something like the banking industry needs this system, then all that needs too happen is the blockchain needs to be modified a little to accommodate different data and a new machine learning algorithm may need to be picked depending on what needs to be predicted.

It is also important to note that in the example that we are presenting, we are predicting the result of the covid-19 test. One could argue that this is not a valuable thing to predict at this particular point in time but that is not the point that this use case attempts to make. We are well aware of the fact that there is not much need for us to be predicting whether someone has covid or not. However, as long as we can show that we can predict something in general with this framework, we can claim that there is no limit to what can be predicted with more accurate and integral data using this framework. An example of this could be if humans were to face another pandemic. There was a time during the covid-19 pandemic that testing was the most important thing but we lacked the resources to be able to get tested. That is a scenario where this technology could have been helpful because it could have given someone a mildly accurate prediction of their condition so they can take action accordingly.

Overall we are very pleased with this system architecture. We believe that this is a great semesters work and we are

leaving a great place for future researchers to pick up where we left off. We are proud of the work we have done and have learned a lot throughout the duration of this course. We really appreciate the opportunity to have learned the things we have learned in this course and to work hands on with this technology.

6 Contribution

Weiting Li:
- Related Work research about Covid-19
- Leading in communicating with Prof & TA
- Exploring Data
- Data pre-processing
- Data model selection and programming
- Evaluation
- Writting report for these parts above
- Powerpoint Presentation

Jiayue Zhou (60%):
- Related work research and writing
- APMRS system architecture design
- APMRS system workflow design
- APMRS algorithms design
- The source coding of APMRS
- The web page to display APMRS
- Writting report for these parts above
- Demo presentation

Drew Klaubert:
- Report writing an organization
- Taking the completed work and reporting it in the final report
- Assisting in communicating with Prof & TA
- Assisting with model selection
- Leading in creating presentation
- Sharing presenting duties
- Creating backup framework in only python in case ethereum version of APMRS did not work properly enough for demo

References

[1] Kai Fan, Shangyang Wang, Yanhui Ren, Hui Li, and Yintang Yang. 2018. MedBlock: Efficient and Secure Medical Data Sharing Via Blockchain. *Journal of Medical Systems* (2018).

[2] Xueping Liang, Juan Zhao, , Sachin Shetty, Jihong Liu, and Danyi. 2017. Integrating Blockchain for Data Sharing and Collaboration in Mobile Healthcare Applications. (2017).

[3] Dounia Marbouh, Tayaba Abbasi, Fatema Maasmi, Ilhaam A. Omar, Mazin S. Debe, Khaled Salah, Raja Jayaraman, and Samer Ellahham. 2020. Blockchain for COVID-19: Review, Opportunities, and a Trusted Tracking System. *Arabian Journal for Science and Engineering* (2020).

[4] Shangping Wang, Yinglong Zhang, and Yalin Zhang. 2019. A Blockchain-Based Framework for Data Sharing With Fine-Grained Access Control in Decentralized Storage Systems. (2019).