Applying Blockchain Technology to Secure Dataset Used for Data Analytics

Chi Keong Lo
School of Computing and Technology
Asia Pacific University of Technology & Innovation
Malaysia
chikeonglo@gmail.com

Nowshath K Batcha
School of Computing and Technology
Asia Pacific University of Technology & Innovation
Malaysia
nowshath.kb@apu.edu.my

Raheem Mafas
School of Computing and Technology
Asia Pacific University of Technology & Innovation
Malaysia
raheem@staffemail.apu.edu.my

***Abstract*— In order for an organization to gain an edge over their competitor, they must be able to make the right decision. Using machine learning models such as XGBoost and Artificial Neural Network during data analysis presents a way to obtain the information necessary for making decision. However, there is a need to secure both the data and the results from analysis to maintain data integrity whilst ensuring that malicious actors will not be able to access. The solution is to merge data analytics together with a private blockchain by storing the data and the result on the blockchain. Using smart contracts for access control, this ensures only users with permission will be able to access whilst maintaining the integrity of the data since it cannot be modified.**

Keywords*— *Data Analytics, Blockchain, Artificial Neural Network.

I. INTRODUCTION

For any organization, it is of upmost importance that they obtain information that can help add value and improve their operations. With the increase data generation, organizations can make use of data analytics to obtain the necessary information that can give them an edge over their competitors. Data analytics as described by authors of [1] is "a combination of processes and tools applied to datasets for gaining invaluable insights to improve firm decision making." Furthermore, it was found that an organization's data analytics competency is the most important factor in determining the decision-making performance, made up of both quality and efficiency. Hence, a way for organizations to improve their data analytics competency is to apply the appropriate models during data analysis. Machine learning models can be applied during data analysis, which involves training a chosen model on a dataset to learn the correct outputs [2]. For example, it can be used for image analysis, object recognition and predictions, and pattern recognition amongst many others. There are two main ways of implementing machine learning models, either through supervised or unsupervised learning. Simply put, supervised learning involves giving the model data to learn how to predict an outcome. On the other hand, unsupervised learning lets the model figure out how to group the data together on its own to find a solution. Both methods serve different purposes, supervised learning is used to identify and predict an output while unsupervised learning is used to understand the patterns underlying the data. Arguably, supervised learning models are easier as it can learn from historical data and predict the likely future outcome by discovering the relationship between the target variable and other variables in the dataset [3]. How well a model performs will be determined by a chosen performance measure and there are two types, classification and regression. Classification models predicts the likely output class of an observation whilst regression models return a likely numerical value of an observation. For example, classification can be used to predict whether a customer will default on their loan while regression can be used to predict the maximum amount a customer can borrow. Once the results are predicted, the data analysis portion could be considered finished. However, beyond data analysis, it is necessary to secure the data and ensure that it would not be manipulated. Blockchain can be used to secure the data and is best known as the technology behind the famous crypto currency, Bitcoin, which allows participants to trade without the need of intermediaries such as banks. When trust in financial institutions was at an all-time low due to the global financial crisis of 2008, Bitcoin was an alternative that appeared as people searched for alternative ways of trading that are more trustworthy. Hence, the first use of blockchain began with the financial sector and had since expanded far beyond as people find innovative ways of integrating blockchain. For example, [4] states that blockchain can be used by governments for identity management of their citizens and achieving an open supply chain for transparency on how a good was produced. The four reasons behind using blockchain is the reduce costs and complexity, shared trusted processes, improve discoverability, and ensure trusted recordkeeping. These reasons are the results of the framework underlying how blockchain works.

In essence, blockchain is a decentralized and distributed ledger where transactions are first verified by a majority of the participants before it is grouped as a block and added to the chain [5]. As it is an append only system, each block added will be chained to the previous block and broadcasted to all the participants, also known as nodes, in the system. This provides the four main features of blockchain which are disintermediation, immutability, redundancy and transparency [6]. Each participant of the blockchain can access to all the transactions that were ever made since each participant had a ledger as there are no intermediaries to handle this. Furthermore,

all the blocks will be broadcasted to all participants and cannot be modified.

An important feature of newer implementation of blockchain is 'smart contract', which is an autonomous program designed to mimic normal contacts. Smart contracts can be created by coding any rules, penalties and conditions necessary into it, which can then be autonomously executed when these conditions are fulfilled [7]. One of the abilities of smart contracts is to control access rules, thereby ensuring data privacy since access is restricted to only users with permission [6]. Furthermore, it is also possible to automate certain processes like reverting firmware or automating payments when certain conditions are met.

II. Problem Statement

An organisation will have many different sets of data that needs to be analysed. However, of the two type of supervised analysis that could be performed, most of the previous research focus on classification problems. The reason being that these types of problems are straightforward because it is easy to understand how well the model does by looking at how many are correctly labelled. On the other hand, regression problems do not have a certainty associated as the outputs gives an estimate of the possible value. The performance measures for regression only states how close or far the estimate is against the actual value. As a result, it is harder to quantify how good a regression model on its own, unlike classification models that states how many are correctly labelled.

In addition, storing data for an organisation can be a problematic issue. They have to identify how to store the data in such a way that it will be secured against malicious actors whilst ensuring integrity of the data when it is retrieved for use. Furthermore, there may be a need to hold many different versions of the same kind of data for different purposes, such as for internal and external reports. In addition, there must be backups to prevent loss of the data. Lastly, there is also a need to manage data access for each personnel, which can become a time-consuming process.

The main contribution of this work is the merging of data analytics with blockchain. Firstly, machine learning methods will be used to obtain a model and the comparison between the different methods could be used as a baseline measure for future works on the same or similar dataset. Secondly, by using blockchain as a security measure, it ensures the integrity of both the model and the results obtained. Furthermore, as all changes to the model and the results will be tracked, it can provide an easy identification of how the data could be used by each subsequent user. Therefore, replicable of the results can be easily performed by any future works to ensure that the results obtained previously was accurate.

III. Literature Review

This section looks at two supervised machine learning methods, XGBoost and Artificial Neural Network (ANN), with the goal of identifying how it had been used in previous works. Also it explores the various works done on blockchain applications.

A. *XGBoost*

XGBoost is a scalable tree boosting system which [8] tested using classification and learning to rank problems. Their focus was on determining the scalability of XGBoost by checking how fast each tree, with maximum depth of eight, was created on average over 500 trees. Classification using base XGBoost was the fastest at 0.68 seconds over other methods such as R.gbm at 1.03 seconds and scikit-learn at 28.51 seconds. Likewise, base XGBoost obtained 0.82 seconds in learning to rank problem, which is the fastest compared to pGBRT at 2.57 seconds. Note that the times improves when XGBoost is further optimised. The most important finding was that XGBoost scales linearly with each additional machine, allowing it to handle 1.7 billion data with only four machines, showing the potential to use minimal resources to handle large amount of data.

Dimitrakopoulos et.al [9] applied XGBoost to biomedical data consisting of gene expression data and pathway data. The goal was to efficiently categories the observations into two groups, young age and old age, by identifying the crucial pathways and sub pathways

B. *Artificial Neural Network*

The type of ANN depends on the network topology, the number of layers and direction information travels (Lantz, 2015). For example, single-layer network refers to where there is only input and outputs, multilayer networks have one or more hidden layers, deep neural networks have more than one hidden layer. Likewise, feedforward network refers to where information travels only from inputs to outputs, backpropagation allows for backwards information travel and recurrent feeds the node output back to itself. ANN naming convention depends on the network topology. The default network is commonly known as the Multilayer Perceptron Neural Network (MLPNN), which is simply a multilayer feed forward network. Ultimately, regardless of the network topology, all ANN modelling occurs by adjusting the connection weights between the inputs and outputs nodes [10].

Hence to summarize when it comes to machine learning techniques, XGBoost and ANN methods are both very popular and very good at accurately predicting the outcome.

C. *Application of Blockchain*

Banerjee et. al. [11] presented a conceptual approach showing how blockchain can be used in dataset sharing, and for compromised firmware detection and self-healing. The conceptual approach appears to take a private blockchain approach with off-chain storage. In both cases, a reference to the dataset and firmware will be stored on-chain and will be used to check against a downloaded dataset or firmware to ensure integrity. This allows for the option to remove the dataset or firmware if necessary. In the case of compromised firmware, the updates can be traced through its history and

reverted. However, as this study was purely conceptual, it is hard to know how well it will fare in actuality. Lu and Xu [12] created originChain, a consortium blockchain with off-chain storage designed to create a traceability platform for goods. This links together the companies, suppliers, retailers and labs together to ensure each good can be traced back to its origin. A consortium blockchain is a private blockchain with multiple owners instead of one. Off-chain storage was rationalized as to prevent all participants from being able to access all the data stored on-chain since sensitive data will be involved. Smart contracts were used to represent the legal agreement and can be updated later with additional or removal of services.

IV. EXPERIMENTATION

The whole process of data analysis was done into three steps, exploration, pre-processing and modelling. The datasets were obtained from Kaggle [13], containing a regression problem to predict the rate of heart disease. These heart disease datasets originated from a competition hosted by Microsoft. The datasets consist of population information "across United States at the county-level". Three sets of data were given and separated into train values, train labels and test values. The goal is to train predictive models using the train values and labels, before proceeding to predict the outcome for each observation using the test values. The train values and labels datasets consist of 3198 observations with 34 and 2 variables respectively. The observations from each set are indicated by their row id which acts as a key to merge the two datasets. Similar to the train values dataset, the test values dataset contains 3080 observations with 34 variables. XGBoost was the first model for comparison and table 1.0 shows each parameter, the description and the tuning that was performed. The last two parameters are the most important as it determines the number of trees needed to build an accurate model. With the 'nrounds' and 'early_stopping_rounds' setting, the model will attempt to improve the RMSE score up to 1000 iterations but if at any point the model does not improve after eight iterations, the process will conclude and returns the total number of trees built. The whole cross validation process was run with 100 iterations, each parameter will be randomised during each iteration mimicking random search. The parameters value in the iteration with the lowest RMSE is noted.

Table 1.0 XGBoost parameters

Parameter	Description	Tuning
max_depth	Maximum depth of tree	Select 1 value from 6 to 10
eta	Learning rate	Random uniform from 0.1 to 0.3
subsample	Subsample ratio of the training instances	Random uniform from 0.6 to 0.9
colsample_bytree	Subsampling of columns	Random uniform from 0.5 to 0.8
min_child_weight	Minimum instance in each node	Select 1value from 1 to 40
max_delta_step	Maximum delta step we allow each leaf output to be	Select 1 value from 1 to 10
nrounds	Number of trees built	1000
early_stopping _rounds	Stop after number of rounds with no improvement	8

The R2 of the model was 0.9857. Using the model on the validation set, table 2.0 shows the RMSE for train and valid set.

Table: 2.0 RMSE for XGBoost model

	Train	Valid
RMSE	29.92	31.57

In ANN, the complexity of the model comes from the different various types of possible parameters to tune for. However, there are several guidelines and rule of thumb available to simplify the tuning. Heaton [14] mention that two layers will often be sufficient unless dealing with complex dataset like time-series. The number of neurons should be between the ranges of inputs up to twice the number of inputs. Brownlee [15] summarizes learning rate and momentum of ANN. Grid search is applied to learning rate from 0.1 to 0.00001 in logarithmic scale. 0.5, 0.9 and 0.99 are common values of momentum. Table 3.0 summarizes the parameter inputs and tuning applied to the model. The ANN model was applied through the mxnet package in R with grid search performed based on the tuning in table 4.6 for the parameters. The model attained R2 of 0.6637 and figure 4.12 shows the ANN output along with the optimal tuned parameters. Table 4.0 shows the RMSE for both the train and valid set.

Table 3.0 ANN parameters

Parameter	Description	Tuning
Layer1	Neuron in first hidden layer	Grid search of values (20, 30, 40)
Layer2	Neuron in second hidden layer	Grid search of values (0, 10, 20)
Layer3	Neuron in third hidden layer	0
Learning rate	Speed at which the model learns	Grid search of values (0.00001, 0.0001, 0.001, 0.01, 0.1, 0.2)
Momentum	Addition of history to weight update	Grid search of values (0.5, 0.7, 0.9, 0.99)
Dropout	Neurons dropped from layer	Grid search of values (0, 0.2, 0.4, 0.6, 0.8, 1)

Activation	Activation function	Rectified linear unit (relu)

Table IV.0 RSME for ANN model

	Train	Valid
RMSE	34.64	38.81

The results from both the models are compiled in table 5.0, with R^2 and RMSE for both train and valid set.

Table 5.0 Summary of results

Model	R2	Train (RMSE)	Valid (RMSE)
XGBoost	0.9857	29.92	31.57
ANN	0.6637	34.64	38.81

The best model was the XGBoost model since it has the highest R2, lowest train RMSE and was basically tied for lowest valid RMSE. It was able to model 98.57 percentage of the variance in the target variable. Furthermore, it likely could be optimized further since parameter tuning was performed through random search with only 100 iterations. Random search essentially assigns the parameter with a random value within a set condition. For example, the learning rate was randomly assigned a value from a uniform distribution between 0.1 and 0.3. With random search, it can be easier to obtain optimal parameter as it covers more of the search space with a smaller number of iterations. However, the number of iterations is also the limitation as due to the random factor, small iterations can completely miss the optimal parameters. For ANN, it performed the worst out of the three models. With the low R2 and higher RMSE for both train and valid set, it is likely that it was not optimized fully. Rather than random search, grid search was applied instead. Grid search works by providing a list of values each parameter can take on and was chosen primarily due to the present of layers and neurons. Technically, the numbers of layers and neurons can be of any values and even with the guidelines, it can be subject to changes depending on the dataset. Furthermore, more layers and neurons can cause over fitting, making it worse at predicting unseen data. Therefore, a grid search was applied to iterate over the search space to obtain optimal model. However, from the model, it can be seen that the grid search was not comprehensive enough to obtain a model that is comparable to the other two models.

In both XGBoost and ANN models, they likely can be optimized further as due to computational limitation, the number of iterations and search space was restricted. However, despite the possibility of further optimization, the computational power and time was already much higher for both XGBoost and ANN models compared to the random forest model. XGBoost was barely better than random forest in term of RMSE whilst ANN performed worse. Therefore, for the purpose of obtaining an efficient predictive model, random forest will be sufficient for this dataset as tuning and optimizing the other two models will take much more computational power and time only for a slight increase in prediction accuracy.

Further the work focus on how a company can make use of blockchain to securely store their data to ensure immutability and integrity of the data. In order to store data onto the chain, the usual method would be to treat each observation as a transaction and store it as such. This can be done by initializing a smart contract structure and reading each transaction into the smart contract. However, this process can be tedious since each observation have to be inputted individually, and expensive since each observation as a transaction. Cost had to be considered since each transaction on the blockchain has a cost associated with it. Whilst this method of storing retain the properties of blockchain, it may not be suitable for some situation such as retrieving the data for analysis.

An alternative was used instead, by storing the data first on InterPlanatary File System (IPFS) which works similar to blockchain but is intended for file storage. IPFS is designed to be a distributed file system where files can be uploaded and stored on it (Protocol Labs). A hash would be returned for the file uploaded, which can be given to others to download the file. However, the obvious flaw would be anyone with the hash would be able to obtain the file. Hence the file will be encrypted first before it is uploaded and the password to decrypt will be given through smart contract. This method will ensure that the owner can restrict the access to the file as it could not be decrypted without the password. Furthermore, the distributed property of IPFS ensures that it will be replicated and stored in multiple nodes.

This method can be seen as a better way of off-chain storage as it retains less of the flaws. The two main problems of storing off-chain is making sure that the data is correctly reflected on the blockchain and the security of the data. By encrypting the data before uploading, it ensures that no one can access without the key. Furthermore, once uploaded, it will be spread to other nodes in the system making it immutable. Therefore, the file cannot be change and will always be reflected correctly in the blockchain. Whilst this proof of concept uses a public version for testing, it is possible to use a private implementation of IPFS for more security. Now that the data has been securely stored on IPFS and access has been decided to be given through smart contract, technically any blockchain that supports smart contracts can be used. However, as another measure of security, a private blockchain will be used so that users can be managed and restricted, such as only to employees of a company. Next a smart contract was added with four functions. Firstly, there is a distinct difference between the users of the smart contract. This work will refer to owner as the user that deploy the smart contract, and user for anyone else that interacts with the smart contract. The first function allows only the owner to add the file hash and password. The second function lets the user view the files that have been added to the system. The third function let the owner approve a user to access of the password. The last function allows a user to retrieve the password for the hash file. Setting the first user as the owner, the smart contract was deployed. Remix was used as the interface for interacting with the smart contract. After the smart contract was deployed, the file hash and password were stored into the smart contract.

This method of blockchain implementation shows how a company can securely store their data or analysis. By using the IPFS as a storage medium in combination with blockchain for

access control, it overcomes blockchain weakness of storage space and ensures that only approved user have access.

For this particular heart disease rate dataset, XGBoost was considered to be the best model at predicting. It obtained the highest R2, highest RMSE for both train and valid set amongst the three models. However, in terms of computational power and time, random forest model was much better as it uses less power and time yet still gave comparable results to XGBoost when predicting unseen data. If computational power and time is not a concern, XGBoost and ANN would likely be much better as it could be optimised further by expanding the tuning space. Another important lesson comes from pre-processing, it is important to test a model on the pre-processed data first before continuing with modelling to check for suitability. For example, the PCA method of pre-processing produced a model that is much poorer at prediction compared to the RFE method.

However, there are many factors beyond data analysis that are often not looked at. A big part is to ensure the data integrity, so that the users can be sure that the data will be accurate and free from manipulation. By using an IPFS, a distributed file system, the data integrity can be preserved. With the addition of a blockchain implementation, it can track who has access to the data. Smart contract can be further implemented to restrict access to the data. All the security measures can be implemented as desired to vary the level of control and access to the data. For data analysis, the users of the data can be certain that everyone else would be using the same dataset for analysis, and the results can be checked for reproducibility.

V. CONCLUSION

For this particular heart disease rate dataset, XGBoost was considered to be the best model at predicting. It obtained the highest R2, highest RMSE for both train and valid set amongst the three models. However, in terms of computational power and time, random forest model was much better as it uses less power and time yet still gave comparable results to XGBoost when predicting unseen data. If computational power and time is not a concern, XGBoost and ANN would likely be much better as it could be optimized further by expanding the tuning space. Another important lesson comes from pre-processing, it is important to test a model on the pre-processed data first before continuing with modelling to check for suitability. For example, the PCA method of pre-processing produced a model that is much poorer at prediction compared to the RFE method.

However, there are many factors beyond data analysis that are often not looked at. A big part is to ensure the data integrity, so that the users can be sure that the data will be accurate and free from manipulation. By using an IPFS, a distributed file system, the data integrity can be preserved. With the addition of a blockchain implementation, it can track who has access to the data. Smart contract can be further implemented to restrict access to the data. All the security measures can be implemented as desired to vary the level of control and access to the data.

REFERENCES

[1] Ghasemaghaei, M., Ebrahimi, S. and Hassanein, K. (2018) Data analytics competency for improving firm decision making performance. Journal of Strategic Information Systems. 27 (1). p. 101-113..

[2] Louridas, P. and Ebert, C. (2016) Machine Learning. IEEE Software. 33(5). p. 110-115.

[3] Lantz, B. (2015) Machine Learning with R. 2nd ed. United Kingdom: Packt Publishing.

[4] Palfreyman, J. (2015) Blockchain for Government? [Online]. Available from:https://www.ibm.com/blogs/insights-on-business/government /blockchain-for-government/ [11/12/2018].T. Poletti, "Fitbit stock tanks as manufacturing woes, slower demand batter holiday sales," Market Watch, California, 2016.

[5] Khan, M.A., and Salah, K. (2018) IoT security: Review, blockchain solutions, and open challenges. Future Generation Computer Systems. 82 (5). p. 395-411.

[6] Savelyev, A. (2018) Copyright in the blockchain era: Promises and challenges. Computer Law and Security Review. 34 (3). p. 550-561.

[7] Lu, Q. and Xu, X. (2017) 'Adaptable Blockchain-Based Systems'. IEEE Software. (December), pp. 21–27.

[8] Chen, T. and Guestrin, C. (2016) XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16). San Francisco, California, USA: ACM. p. 785-794.

[9] Dimitrakopoulos, G.N., Vrahatis, A.G., Plagianakos, V. and Sgarbas, K. (2018). Pathway analysis using XGBoost classification in Biomedical Data. In Proceedings of the 10th Hellenic Conference on Artificial Intelligence (SETN '18). Patras, Greece: ACM. Article No. 46.

[10] Han, J., Kamber, M. and Pei, J. (2012) Data Mining: Concepts and Techniques. 3rd ed. United States of America: Morgan Kaufmann.

[11] Banerjee, M., Lee, J., and Choo, K.-K.R. (2017) A blockchain future to Internet of Things security: A position paper. Digital Communications and Networks. doi: 10.1016/j.dcan.2017.10.006.

[12] Lu, Q. and Xu, X. (2017) 'Adaptable Blockchain-Based Systems'. IEEE Software. (December), pp. 21–27.

[13] Kaggle (2018) Microsoft Data Science Capstone [Online]. Available from: https://www.kaggle.com/nandvard/microsoft-data-science-capstone [20/11/2018].

[14] Heaton, J. (2017) Heaton Research: The Number of Hidden Layers [Online].Availablefrom: https://www.heatonresearch.com/2017/06/01 /hidden-layers.html [13/3/2019].

[15] Brownlee, J. (2019) How to Configure the Learning Rate Hyperparameter When Training Deep Learning Neural Networks [Online]. Available from: https://machinelearningmastery.com/learning-rate-for-deep-learning-neural-networks/ [Accessed: 13/3/ 2018].