Proceedings of the Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)
IEEE Xplore Part Number:CFP20OSV-ART; ISBN: 978-1-7281-5464-0

Quantum Neural Networks for Dynamic Route Identification to avoid traffic

Sumati Boyapati
Sr. Solution Architect
IRIS Software, INC
Edision, NJ
sumati.b80@gmail.com

Srinivasa Rao Swarna
Sr.Data Architect
Tata Consultancy Services
Edison, NJ
swarna.prince@gmail.com

Abhishek Kumar, SMIEEE
Chitkara Institute of Engineering and Technology,
Chitkara University,
Himachal Pradesh, India
abhishek.kumar@chitkara.edu.in

***Abstract*— Computation is the primary task performed for the evaluation of the solution for a specific problem, and in real-time, having better challenges to implementing the solution path with the better computational mechanisms. The concept of quantum computation mechanism using the neural networks is having the highest amount of the success rate in prediction models design and implementation. The idea of a dynamic routing mechanism using quantum computing and neural networks are the main essence. A better prediction model is performed for this specific kind of problem, which needs a particular focus on the latest problem-solving mechanisms. The problem-solving tools like neural networks will dynamically perform with real-time data, but a new add-on is needed to add like big data to implement the live data. The live data can help implement and understand the importance of solving the problem like dynamic routing mechanism. There is a chance of random growth in such a field of computer science. This computational mechanism using quantum computing and the neural network will track the live operations and form the dynamic route changes in the real-time scenario. This real-time scenario worked with a 95% accuracy rate. The accuracy will differ based on the number of connecting nodes are being considered to evaluate the hidden layers of the problem-solving mechanism.**

Keywords—Machine Learning, Quantum computing, Deep Learning, Big Data, Spark, Routing

I. INTRODUCTION

Machine learning mechanisms like classification and regression will solve some problems, and there are many more chances for designing dynamic applications with live data. That kind of data needs to be modified dynamically, and the pre-processing mechanisms will dynamically clean the data. The information which is considering is based on the multiple data frames. The data frame operation can be useful using the programming language like python. The programming languages will help for the better implementation of the problem in the real-time scenario. But there will be a question for the developers to understand which kind of dynamic data to be considered to implement the dynamic routing using quantum computing. Quantum computing will work based on the physical implementation of the concept in the sub-atomic region based on aerospace logic. Random locations are having in the subatomic and needed to consider the particles in the subatomic and plot the variables in the output plane[1-3]. The same concept is used in this research to identify the dynamic implementation of the routing identifications. The routing is an active part of the problem-solving mechanisms. There must be a scenario to understand the modifications of the object's locations in the hypothesis space. The hypothesis space will work on combining similar kinds of items into one type of category. This concept will be considered a significant portion of considering the dynamic components in developing the prediction model. In this prediction model of the dynamic routing using the QNN, it will be challenging to understand how to get the dynamic data. The live data will be taken by the big data concepts like apache spark. There will be a sequence of implementation of gathering the data and implementing the pre-processing mechanism. In this mechanism, two steps are having for the performance, going to know in the lateral part of this article. One thing is needed to understand before going with the concepts like QNN that it is the extension of the machine learning concept, artificial neural networks, and there will be the series of the hidden layers using which are generating the process of giving the accurate output for the specific problem in the real-time. The implementation will be in the form of two patterns. In the first pattern, the users' GPS in a specific location is considered, and there will be a time-bound of the implementation. In this time-bound, whether the user or the mobiles in that particular location are moving the accurate direction or not will be tracked. If the users are not moving in that direction, then there is a chance of identifying the alternative routing of the specific path. Here the comparative study of the past and current implementation of the maps are needed to perform. For instance, consider a company holding the variables that require a more significant number of inputs from experience. There is a possibility of gathering data using the spark SQL or any data manipulation techniques. Two-stage verification and comparison are being performed among the variables considered the necessary level of the feature extraction. In this feature extraction, needed to choose the rule base to generate the logic of gathering the variables beyond the hypothesis. This logic will create the variables which are in the form of both positive and negative. [4]

The lateral part of the article deals with the logic behind using the quantum neural networks; next will explain a few of the existing approaches. The following section will explain the methodologies used for this concept; the next proposed architecture will be described as a continuation of the results and the future scope of the idea.

II. PROCESS OF TWO-WAY IMPLEMENTATION

The reason behind using the two-way mechanism is to understand the process of feature extraction and the implementation of the accurate prediction model for the data chosen. In designing the solution for the specific problem, may get the challenges of understanding the purpose of the problem. Here, the problem identification's primary goal, and the answer for the particular situation, is to make the passenger's dynamic and accurate route from the source to the destination. But the concept here will work only for the single source and single goal in the current approaches. The current system is having the procedure of understanding the machine learning implementation on real-time data, but that will not be the live data. The live data concept and pattern are different from the proposed architecture. The existing methodologies will work on the time-bound, but the same idea is being used in a better way because the current approaches are not with the neural networks. There is a chance of implementing the neural systems in a better way of understanding the live data, and the Spark SQL will help understand the information, which is in the form of live data. The live data will be monitored using the algorithm. The keywords mentioned using the following formula will make you understand the mechanism of implementing the neural networks in the dynamic routing mechanism.

$$y_{ij} = \mu_0 + \mu_1 + \mu_2 + \ldots\ldots\ldots\ldots\ldots + \mu_n \quad \text{----- (1)}$$

Y is the dependent variable, which is holding the values of the model output, and there will be a recursion in the implementation of the (1). The (1) will have the importance of multiple conditions. There is a procedure of using the instance of the data, and that example will consider as the best methodology for the data modification from time to time in the frame. There is a chance of analyzing the information like what is most general time-bound and the least general time-bound, which are mentioned as the MGTB and LGTB. μ will handle the different variables in the other time-bound t. (2) will explain the formation of value for each μ in the form of t.

$$\mu_i = \sum (i + i_k) \text{ ---------- (2)}$$

For every value in k from 0 to n, there is a chance of forming the correct combination of the rules, and there must be a side of using negative instances, which is a concept of thinking beyond the hypothesis space.

III. III. EXISTING APPROACHES

The existing approaches use different machine learning methodologies, which can understand the purpose of advanced methodologies implementation. The implementation will help to understand the inner concept of the problem statement. But the existing approaches are meant to work on the traditional methodologies. That concept will focus on conventional machine learning models like SVM, Random Forest, Decision Trees, KNN, and K-Means. The following literature review will help to understand the purpose of two-stage QNN modelling.

The model [5] is implemented using the image classification and taking the time boundary of the event's occurrence. In this occurrence of the event, quantum physics is needed to study, which can take the object's kinetic motion in the hypothesis space. In this hypothesis space, the item is being considered based on the distance from the regression line. If the point plotted in the hypothesis space is nearer to the regression line, it can be considered the optimal path to traverse.

The different QNN mechanisms that can help understand the medical implementation is discussed [6], and the medical performance will make you understand the complexity in the implementation of any feature in this two-stage verification QNN process. In the first stage, the parts are considered based on the occurrence, and the second stage will make you implement the actual modelling mechanism.

Routing based on the GPS mechanisms is explained [7], and there is a chance of understanding the concept of GPPS, which can make you track the device locations, and there is a procedure need to follow. That is to track the happening of a device in a place for a specific timestamp.

Fig. 1: Device tracking using GPS to find the happening at a location

The device which is being tracked will form the inputs based on it is traversal, and those will be considered as the necessary inputs. Bu t the dynamic implementation of the routing will not happen all the time because it cannot understand the positioning of the device after a timestamp.

Similarly, having the concepts mentioned by [8-10] in which quantum computing using the neural networks playing a vital role in understanding the purpose of every operation of the device in a specific time bound[11-13].

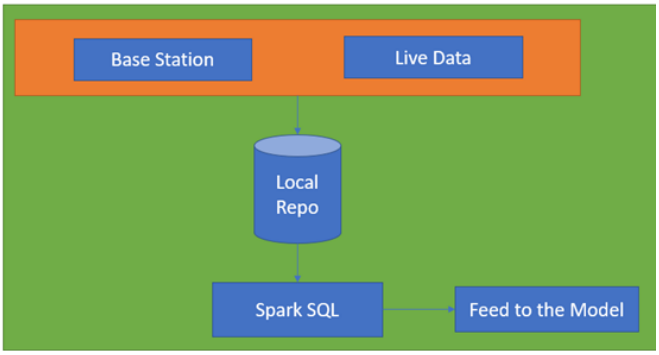

Fig. 2: Stage 1 of the two-level verification

A. *Proposed Architecture*

The proposed architecture will work based on the two-stage verification of the ANN and QNN. This concept will work both in RNN and CNN too. The RNN mechanism will work on understanding the negative and positive feeds of the data. But in the first step, all the stage one data will be handled by the Spark SQL. The following architecture mentioned in figure 2 will explain the concept[8-10].

In figure 2, the live data will be considered from the location, and the Spark SQL will handle that. Using the commands in the Spark SQL, the location will be tracked dynamically. The base station will start tracking all the devices in a particular area, and they will update in the live data stream. Using the live data stream, the data is stored in the local repo's that can be processed using the spark SQL, and then the data are being forwarded to the pre-processing. The pre-processing will clean the data, and that can be submitted to the model for the prediction.

After sending that to the model, then QNN will check for the common points in the data. That means location. If there are many triggers for the same place, it can be used to recognize the dynamic place better. The below pseudocode will be using for the recognition of the active route [14-17].

The proposed methodology is included in two-way verification. In this two-way verification process, the data will be collected first by the GPS landmarks of the specific location. The location landmarks can be gathered by tracking the GPPS of the devices which are moving in the specific location. In the first phase, needed to consider the trigger as zero initially. Because the traffic is null in the initial stage. To identify the traffic in a specific location, needed to move from one location to another location. The main reason to identify the traffic for dynamic routing is to gather the frequently used paths from a certain source to destination. If the traffic is more in a certain location based on the default threshold value, then trigger will be incremented and the people who are travelling in that specific direction will get the alternate direction.

```
Two-way Verification (Location, Trigger, DB):
    Trigger = 0
    Connect_DB
    If(Location = i)
    {
        I = I + 1
        Trigger = I
        Append to DB
    }
Function (people, location, trigger, time):
    Trigger = 0
    People = 0
    Location = 0
    Time = 0
If (people (location) > = 180){
    People += 1
    Location => Saved
    Trigger = Trigger + 1
}
The route will be considered dynamically
Save the route and update it immediately.
```

The data stored will be sent to the QNN. Before going for the QNN, that data is needed to be processed with the decision trees and random forest because the rule is generated for every location identification. If the trigger increases for a specific location, the regulations to be imposed on the data is needed to be increased [18-20].

The two-step verification is not a great complex task to accomplish. It looks complex but can be applied in ease manner. The rule in phase one is to analyse, which are the places or the locations, calling it as having the highest trigger rate. After analysing the trigger, the time-bound of the traffic is calculated in that specific route. If the route R1 is available on day 1 and not using in day 2 then it can be considered as the route has some issues and then the traffic can be maintained dynamically.

The major purpose of the current research work is not for the traffic signalling management or place to place traffic prediction or update. The major focus in on identifying the traffic which may occur in odd times because of not having an idea on not working path so that traffic may occur in those locations. Using QNN, odd time traffics in the path can be dynamically sucked from source to destination.

```
Decision (Location, Trigger, Time):
Trigger = 0
Time = 0
If(Current_loc [location] > = 10){
Trigger = Trigger +1
}
Else if (Current_loc[time] >=180){
```

```
    Triggere = Trigger + 1
    Update the location to the server
    }
    Random_Forest(Location, Trigger, Time):{
    Repeat(Decision())
}
```

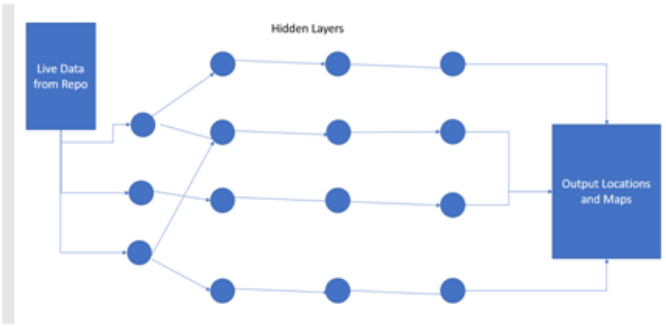

Fig. 3: Identification of different approaches

The following graphs are the results that occurred in analysing the trigger and the location combination. Figures 4 and 5 representations the person and location ratio, location, and trigger ratio.

The algorithm or the pseudocode mentioned in the proposed system will work on understanding the location triggers. The trigger is the component which can track the location by tracking the number of times a specific location is visited. If the trigger increased the threshold value, then it will be considered as the most used path. If the path is not triggered in a time-bound which was used in the previous day, then it must be having some problem. Then it must be dynamically tracked by the neural network.

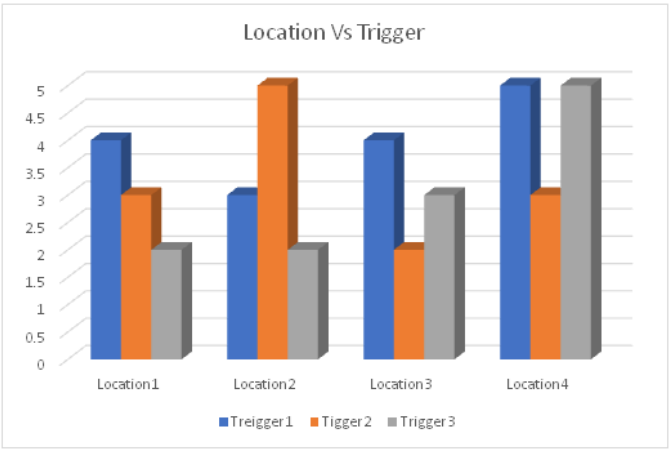

Fig. 4: Location Vs Trigger

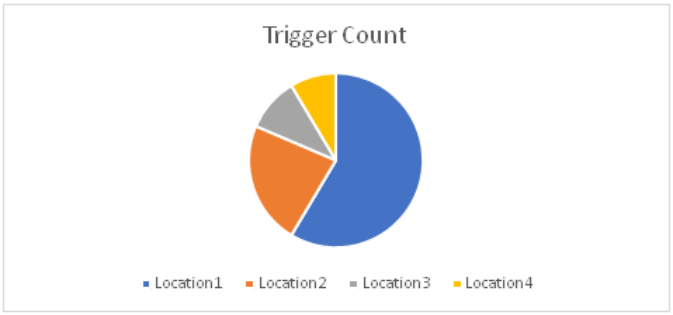

Fig. 5: Trigger Vs Location Count

IV. FUTURE SCOPE

This concept's future scope is a better implementation with more accuracy and making iteration more to understand any other chances of creating more accurate models. The models which are performed in this research will have a respectable accuracy rate. Expecting more accuracy in further research [21-23].

V. CONCLUSION

The dynamic routing implementation deals with the understanding of the models, providing the excellent accuracy of the active route identification. If a person is moving from one source to one destination and unfortunately the route was damaged and cannot move forward, the same thing needs to be notified to the people driving in the same location so that it will be dynamically informed to the other people who are traveling from the same source to destination. QNN and ANN will have the highest scope of developing the current prototype in a better way. The current prototype will work on a single source and a single destination.

References

[1] S. Arunachalam and R. de Wolf, Guest column: A survey of quantum learning theory, ACM SIGACT News, vol. 48, no. 2, pp. 41–67, 2017.

[2] D. Kopczyk, Quantum machine learning for data scientists, arXiv preprint arXiv: 1804.10068, 2018. [3] M. Schuld, I. Sinayskiy, and F. Petruccione, An introduction to quantum machine learning, Contemporary Physics, vol. 56, no. 2, pp. 172–185, 2015.

[3] M. Schuld, I. Sinayskiy, and F. Petruccione, The quest for a quantum neural network, Quantum Information Processing, vol. 13, no. 11, pp. 2567–2586, 2014.

[4] M. A. Nielsen and I. L. Chuang, Quantum Computation and Quantum Information. Cambridge, UK: Cambridge University Press, 2000.

[5] R. Jozsa. Quantum computation prerequisite material, https:// www.semanticscholar.org / paper / QUANTUMCOMPUTATION-PREREQUISITE-MATERIAL-Jozsa/ 6dc618f0041a20d8c722ed2cdd8c4057d6e92a0b, 2013.

[6] A. Eremenko, Spectral theorems for hermitian and unitary matrices, Technical report, Purdue University, USA, 2017.

[7] I. Wegener, Branching Programs and Binary Decision Diagrams: Theory and Applications. Philadelphia, PA, USA: SIAM, 2000.

[8] F. Ablayev, A. Gainutdinova, and M. Karpinski, On computational power of quantum branching rograms, in Fundamentals of Computation Theory. FCT 2001, R. Freivalds, ed. Berlin, Germany: Springer, 2001, pp. 59– 70.

[9] F. M. Ablayev and A. V. Vasilyev, On quantum realisation of Boolean functions by the fingerprinting technique, Discrete Mathematics and Applications, vol. 19, no. 6, pp. 555–572, 2009. [11] F. Ablayev, M. Ablayev, K. Khadiev, and A. Vasiliev, Classical and quantum computations with restricted memory, in Adventures Between Lower Bounds and Higher Altitudes, 2018, pp. 129–155. [12] C. A. Trugenberger, Probabilistic quantum memories, Physical Review Letters, vol. 87, no. 6, p. 067901, 2001.

[10] P. Kaye, Reversible addition circuit using one ancillary bit with application to quantum computing, arXiv preprint arXiv: quant-ph/0408173, 2004.

[11] L. K. Grover, A fast quantum mechanical algorithm for database search, in ACM Symp. on Theory of Computing, Philadelphia, PA, USA, 1996, pp. 212–219.

[12] G. Brassard, P. Høyer, M. Mosca, and A. Tapp, Quantum amplitude amplification and estimation, Contemporary Mathematics, vol. 305, pp. 53–74, 2002.

[13] L. K. Grover and J. Radhakrishnan, Is partial quantum search of a database any easier, in ACM Symp. on Parallelism in Algorithms and Architectures, Las Vegas, NV, USA, 2005, pp. 186–194.

[14] M. Boyer, G. Brassard, P. Høyer, and A. Tapp, Tight bounds on quantum searching, Fortschritte der Physik, vol. 46, nos. 4 &5, pp. 493–505, 1998.