Constructive Learning of Deep Neural Networks for Bigdata Analysis

Soha Abd El-Moamen Mohamed
Information System Department
Faculty of Computer and
Information Assiut University
Egypt

Marghany Hassan Mohamed
Computer Science Department
Faculty of Computer and
Information Assiut University
Egypt

Mohammed F. Farghally
Information System
Department
Faculty of Computer and
Information Assiut University
Egypt

Abstract. The need for tracking and evaluation of patients in real-time has contributed to an increase in knowing people's actions to enhance care facilities. Deep learning is good at both a rapid pace in collecting frameworks of big data healthcare and good predictions for detection the lung cancer early. In this paper, we proposed a constructive deep neural network with Apache Spark to classify images and levels of lung cancer. We developed a binary classification model using threshold technique classifying nodules to benign or malignant. At the proposed framework, the neural network models training, defined using the Keras API, is performed using BigDL in a distributed Spark clusters. The proposed algorithm has metrics AUC-0.9810, a misclassifying rate from which it has been shown that our suggested classifiers perform better than other classifiers.

Keywords: Health care, Deep learning, Constructive Deep learning, Diagnosis systems, Big data analysis

1 INTRODUCTION

Medical applications can predetermine the parts of lungs damaged in increasing levels and this is important at stages of lung cancer. We can collect images of damaged parts and diagnose the affected parts early. Medical data has three types: structured, semi-structured, and unstructured data. [4] explains an overview of unstructured data that considered as the foundation of predictive analysis. Here's the power and importance of unstructured data for predictive analysis. So, the organizations shouldn't neglect unstructured data. Structured data is the data which has a fixed and normalized shape or format. Structured or unstructured datasets can be handled effectively according to the perspective of a big data framework [18]. Machine learning and deep learning methodologies thus offer alternate

approaches to these problems and an explicit connection to big, high-dimensional datasets [12].
It might be reasonable to choose Neural Network (NN) architecture through manual design if there are qualified human experts with sufficient advanced knowledge of the problem to be solved. However, this is clearly not the case for some real-world situations where a number of advanced information is not available. NN deploys a fixed architecture in the back-propagation algorithm, while a constructive algorithm takes on dynamic NN architectures. Constructive algorithm is a supervised learning algorithm and its dynamic models are commonly used to solve real-world problems. Constructive algorithm starts a NN with a limited design, i.e. with a limited hid- den layers, neurons and links. First, start searching for a simple NN approach, then attach hidden nodes and weights gradually until an optimal methodology is identified. This paper analyzes a more proficient and massive data processing framework, Apache Spark. Apache-spark allows to solve iterative Machine Learning (ML) problems by using a distributed deep learning module called Spark BigDL framework which is a Apache Spark framework that depends on distributed deep learning [3]. Deep learning is useful in acceleration at data processing even if this data is structured or unstructured [18]. For managing high-dimensional datasets, we used a constructive deep-learning with Spark. Identifying in healthcare systems, [8] multi- disease simulations implemented.
The contribution of this work is to perform a distributed Convolution Neural Network (CNN) using constructive deep neural network algorithm called cascade-correlation growing Deep Learning Neural Network Algorithm (CCG-DLNN) by using Spark BigDl framework.
The paper is set out as following. Section 2 presents a related works. Section 3 includes a comprehensive overview of the proposed Algorithm. Section 4 is about results and experimental analysis and a conclusion is at Section 5.

2 RELATED WORKS

One of the most common cancer worldwide is the lung cancer. We generally apply different strategies for the detection of lung cancer. The detection of pulmonary nodules is using deep learning algorithms based on the CT (Computed Tomography) images exhibit. A deep neural network can be used to identify hidden details within morphological features and merging the learned representations with the initial morphological features as in [13] which used Stacked Denoising Auto Encoder (SDAE) for deep learning functionality. Extraction of CT images is by using ROI segmentation which saved using a compression technique [11]. [13] proposed a technique using the size of the nodule which has results better than im- age segmentation using ROI for detecting cells with cancers [11]. [21] performed the image preprocessing using mean and median filters, then it segmented the lung of CT image by Otsu's threshold and a segmentation approach called marker-controlled Watershed. The physical dimensional measure and the gray-level co-occurrence matrix (GLCM) method are performed for the feature extraction step. Finally, it detected the cancer nodules from these resulted features. The best methods to classify images to detect cancer are GLCM and SVM. [19] make sputum color images using the CAD (computer-aided design) system, and this is different from NC ratio and circularity which are types of feature extraction derives properties of models and estimate cells as cancerous using a threshold.
[9] used deep learning applications for exploring non-small-cell lung cancer. extracting quantitative imaging features Prognostic signatures during deep learning networks, quantitative imaging features prognostic signatures and patient stratification scrutinized physiological imaging artifacts. [2] used nondelta features resulted from clinical tomography images for analyzing the features of delta radiomics to improve result from these images. Also, they used a ROC curve to improve the mechanism performance at lung cancer prediction with a multiple number of features. Moreover, the paper improved performance by utilizing the radiomics conventional features.

[20] extracted local features at video sequence by using the elastic solutions on the distributed environment based on the Spark paradigm and the bag of visual words (BoV) model state of the art. Then the paper computed feature maps of CNN by adopting the BigDL library.

Our focus in this paper is to design a distributed CNN model using a constructive deep neural network that effectively models the CT lung images. We used TensorFlow and Spark to force the parallelism of data and Spark scheduling, to enable tensor communication directly using parameter servers and TensorFlow executors. Direct Process-to-process communication enables TensorFlow program to scale without effort.

3 THE PROPOSED APPROACH

A detailed framework that is mentioned in this paper will be discussed in this section. The proposed approach combines the benefits of frameworks that used for big data processing (i.e. Spark) with the benefits of a constructive deep neural network algorithm called cascade-correlation growing Deep Learning Neural Network Algorithm (CCG-DLNN).

3.1 Cascade-Correlation Growing Deep Learning Neural Network Algorithm (CCG-DLNN)

Constructive neural networks (CoNN) are a set of algorithms that changes the design of the network, automatically generating a network with an acceptable dimension. The algorithms used for the CoNN learning method are called constructive algorithms. Constructive algorithm (As in Fig. 1) begins with a small network design and adding ayers, nodes, and connections as appropriate during training. The architecture adaptation process continues until the training
algorithm gets a near optimal solution for the problem.

The CCG-DLNN algorithm uses the same strategy like Cascade Correlation Neural Network algorithm (CCNN) [5] but by adding more than one hidden layer - that has more than one neuron - between the input and output layers. So, at each iteration we add a new hidden layer with one neuron or add a new neuron at the last added hidden layer (l = NHL). At the beginning, the learning algorithm begins with a simple network that has only the input and out- put layers and does not have any hidden layers. Due to the lack of hidden neurons, this network can be learned by a simple gradient descent algorithm ap- plied individually to each output neuron. New neurons are connected to the network one by one during the learning process. Each of them is put in a new hidden layer and linked to all of the previous hid- den neurons at the previous hidden layer. When the neurons are eventually connected to the network (activated), their input connections will freeze and no longer change.

Adding new neuron can be divided into two sections. First, we start with a candidate neuron and receiving input connections from all pre-existing hidden units at last hidden layer (l = NHL). The candidate unit's output isn't yet added to the current network. We do a sequence of passes over the examples set for training, updating the weights of the candidate unit for each pass at input side. Second, the candidate is linked to the output neurons (activated) and then all output connections are trained. The entire process is repeated multiple times until the desired accuracy of the network is achieved.

let C is introduced to measure changes in loss function when a new neuron is added to the current hid- den layer and τ is the threshold to add a new hidden layer. As shown in equation 1 and equation 2, If The loss function L (φ_{t-1}) at iteration (t − 1) is degraded compared to L(φˆt) at iteration (t) before adding the new neuron, then increment C by one. Otherwise, C will be zero. While C less than the threshold τ, add one new neuron to the last hidden layer. If C reaches the threshold value τ then adding a new hidden layer that has one neuron as in Fig. 2.

$$\begin{cases} C = 0 & if L(\varphi_{t-1}) - L(\hat{\varphi}_t) > \epsilon \\ C = C + 1 & otherwise \end{cases} \tag{1}$$

$$\begin{cases} l = l + 1 & if C \geq \tau \\ N_{ul} = N_{ul} + 1 & otherwise \end{cases} \tag{2}$$

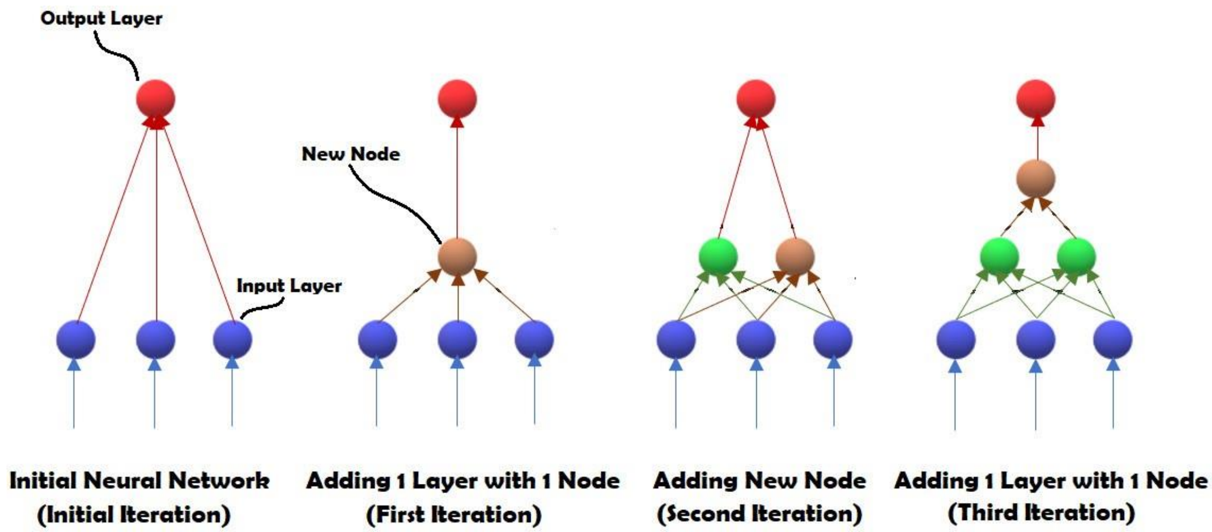


Figure 1: Constructive Deep Neural Network

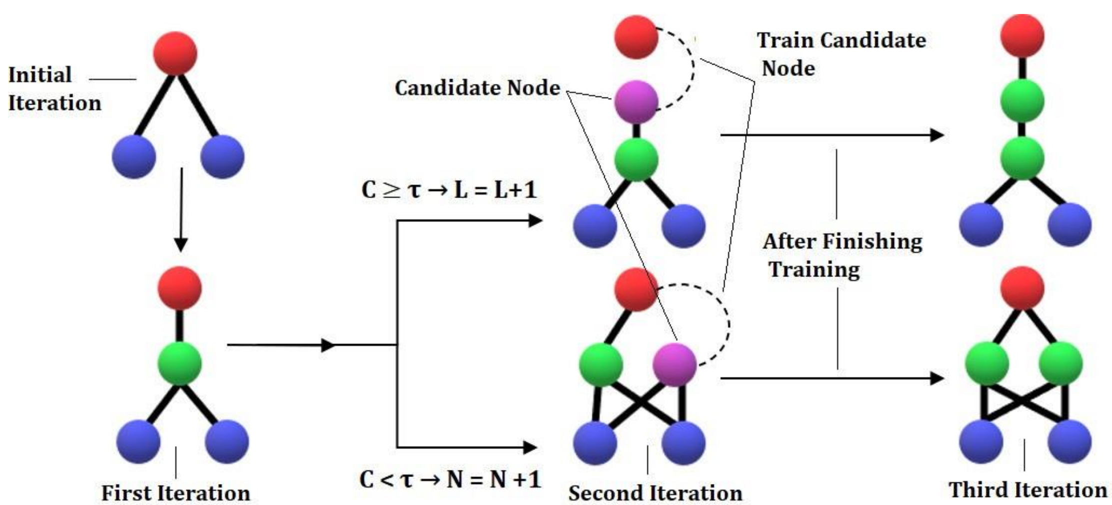


Figure 2: CCG-DLNN Algorithm

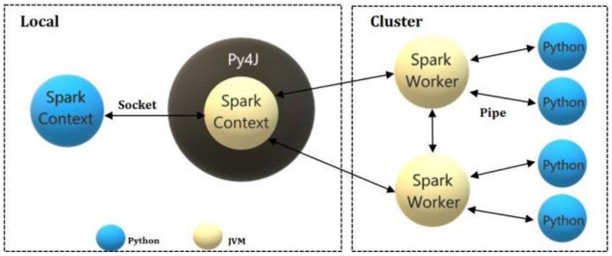


Figure 3: PySpark Framework

3.1 Framework

To solve real-world data problems like the classification of lung cancer images, we propose a big data analytic framework with constructive deep neural net- work. Addressing such problems is difficult because of time and space constraints. Because of the huge data and the lack of devices with high computing power, supportive learning frameworks are needed that can handle this data using only the stipulated sources.

To overcome these challenges, this paper combines big data analysis, machine learning and constructive deep learning. The basic structural framework presented in this section forms the core of our research and experience.

3.1.1 Apache Spark

Apache Spark can analyze faster than Apache Hadoop. It is considered as a general-purpose computing system and provides an open-source, distributed cluster system. Scala programming is what Spark is based on. Like java, Scala compiles firstly at bytecode using JVM to process the big data using Spark. PySpark is released by the Apache Spark com- munity for supporting python at Spark. For Spark programming, Pyspark API is developed in python and for developing spark applications at Python as shown at Fig. 3.

In this paper, we combine TensorFlow and Spark [1] to perform the parallelism for the data processing and Spark scheduling to enable tensor communication directly using executors of TensorFlow and parameter servers.

3.1.2 Distributed Convolutional Deep Neural Network in big data analytics

There are many hierarchical layers at the Convolutional Neural Network (CNN). These layers divided into feature maps and classification layers. CNN accepts data from the input layer and sends it to the convolutional layer as shown in Fig. 4. The convolutional layer does convolution operations by having the same size filter maps. Then, the output of the convolutional layer is directed to the sampling layer to reduce layer size. There are many numbers of deep learning methods which are locally connected to the CNN [10]. The limited shared memory is considered as a challenge in big data analytics. Researchers com- bine the layers of convolution and sampling at only one step [17]. So, the activities and error values are stored at a single step when applying backpropagation.

At the proposed framework, a distributed convolution deep learning is used in big data analysis as shown in Fig. 4 to learn features and classify nodules in CT Lung images into malignant or benign nodules.

3.1.3 The Steps of a proposed framework

1. Image pre-processing:
   The first step at the proposed framework is the image pre-processing which has two steps:

   (a) The first step at the image pre-processing is image smoothing. Median filters are applied to input image which helps in reducing image noises. Median filters remove all elements that have a high frequency from input images for providing smoothed and accurate intensity surface image as an output [16].

   (b) Then the dual-tree complex wavelet trans- form (DTCWT) algorithm that was pro- posed by [14] is performed. The DTCWT is important in solving shift variance problems and lowness at directional selectivity in two or more dimensions pictures. It is based on a discrete wavelet transform (DWT). It calculates the mean energies of the real and imaginary parts of separated

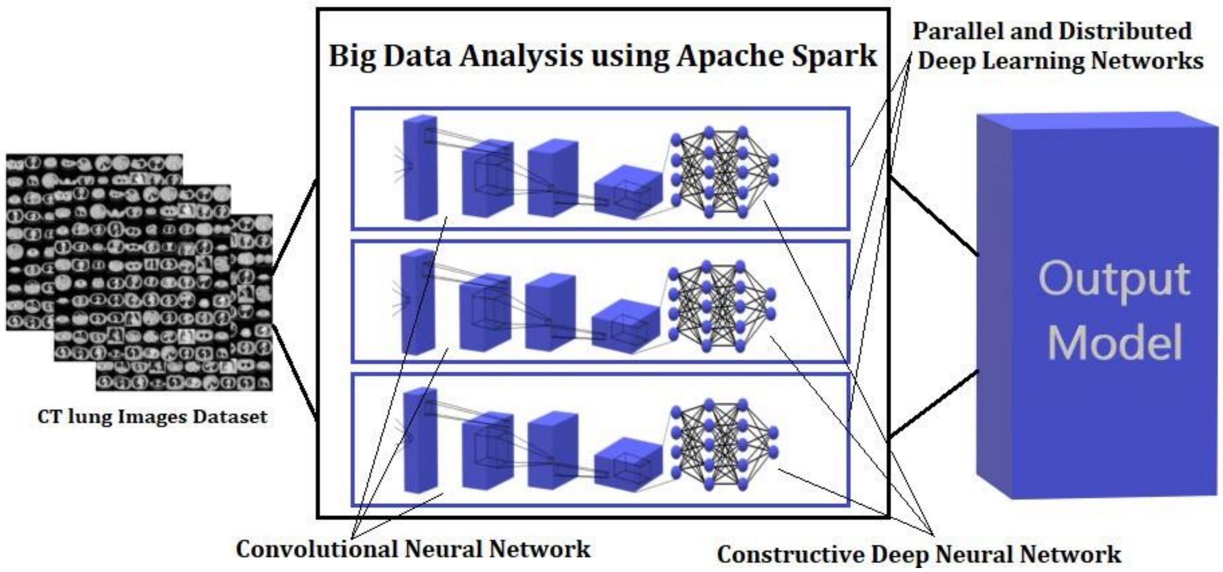


Figure 4: The proposed Model

complex wavelet coefficients. Then, these energies identify the image effective features for defect detection. DTCWT includes dual-tree shift-invariance and selective orientation for surveying the wavelets which are 2D or 3D. DTCWT helps in denoising at volume and image.

2. Feature Extraction:

In our framework, we use Gray Level Co-occurrence Matrix (GLCM) that proposed by [7] for extraction the features from the MRI lungs images. The extraction of the feature is a dimensional reduction process. It transforms data into feature sets.

A Gray Level Co-Occurrence Matrix (GLCM) has information of pixels' positions at the image which have the same values of gray levels. The GLCM entropy feature is used in this paper. Entropy is used for measuring the missing information or message in the transmission process and also measures the information into image (complexity of an image).

The entropy equation is shown at Equation 3 [7].

if the probability values P (i, j) is allocated uniformly along the matrix of GLCM, then the highest Entropy value is found. This occurs when the image hasn't pairs of grey level, with particular preference over others.

$$f = -\sum_i \sum_j p(i,j) log(p(i,j)) \quad (3)$$

where i, j is the cells number of GLCM matrix.

3. Data parallelism:

In data parallelism, the data is partitioned into small subsets. Each data partition enters as an input to each executor.

At each executor, there is a copy of CNN and a constructive deep neural network which take the subset data as input and parallelize the processing of gradient descent.

The server receives gradient delta from each executor and then synchronizes the model parameters between executors. Then, the output of each executor is combined as shown at Fig. 4.

For task scheduling and partitioning data, we

used Spark at the proposed framework. Each executor has a Spark wrapper of the Tensorflow application.
Each executor has one node that handles the synchronization of the parameter and the remaining nodes run independently the application of Tensorflow. After batch elements processing and receive the last parameter from the server, the executor sends the delta of its gradient to the parameter server.
At the same time, Spark core sends the partition of the data to each executor.
The count of data partitions is based on the count of epochs the size of the dataset. Spark driver handles tasks and replicates the TensorFlow model to each cluster.
Tasks are created and sent to all executors for each stage and partition. The driver generates new tasks for the new stage and sends these tasks to executors, after complete all tasks of one stage. This task repeats up to the last stage and sends the results to the driver [15].

4. CNN and Constructive deep neural network:
As shown at Fig. 4, each executor has a Convolutional neural network (CNN) model which based on a constructive deep neural network.
As shown at Fig. 5, CNN consists of two main layers which are the layer of feature extraction and the fully-connected layer. The layer of the feature extraction consists of (a convolution, a nonlinear, a pooling, and an overlay layer). The fully connected layer is responsible for taking the output of the pooling layer and classifying the image into a label [22]. At our proposed framework, we replace the traditional fully connected layer at CNN model with a constructive deep neural network algorithm called CCG-DLNN algorithm (see 3.1). The CCG-DLNN algorithm has two benefits: Firstly, the cascade architecture where adding the hidden node to the neural network each time and doesn't modify after inserting it. Secondly, it is a learning algorithm that creates and adds a new hidden neuron,

4 EXPERIMENTAL RESULTS

4.1 Dataset

We used the The Lung Image Database Consortium image collection (LIDC-IDRI) [6].
The LIDC-IDRI dataset is an available set which has 1018 lung CT scans from many different organizations and universities. There are 3 structures types at this dataset:

1. The largest diameter of lung nodules which is wider than 3 mm.
2. The largest diameter of lung nodules which is narrower than 3 mm.
3. The largest diameter of Nonnodule structures which is wider than 3 mm.

We divided the Motor The LIDC-IDRI dataset into two subsets of 734 training and the remaining 284 cases for validating.

For performing our framework, the preparation steps include pre-processing (Median filters for image smoothing Fig. 6 and, The dual-tree complex wavelet transform for solving shift variance problems and lowness at directional selectivity) is performed firstly. Feature extraction is performed after pre- processing step by using e Gray Level Co-occurrence Matrix (GLCM) for extraction the features from the MRI lungs image.

4.2 Cloud platform

The proposed model is deployed on Google Cloud Dataproc which is a managed service for big data processing. Our model is based on BigDL library. BigDL is a library which based on distributed deep learning and developed for bringing deep learning that is supported to Apache Spark. The pre-processed CT-lung image data was stored in the cloud through Google Storage Bucket. We use Google Dataproc Cluster with the following versions (Debian=9,

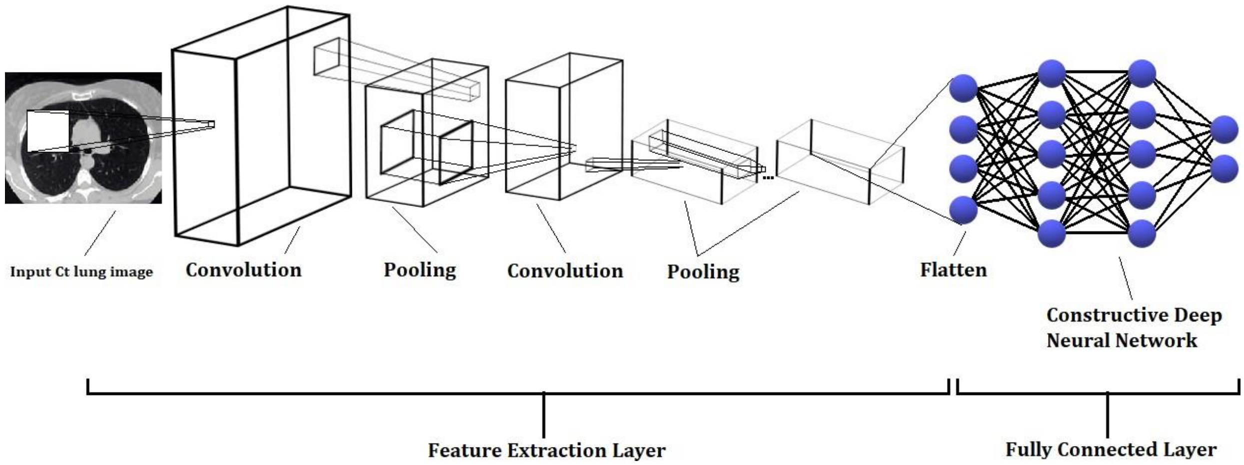


Figure 5: CNN and Constructive deep neural network

Hadoop=2.9, Apache Spark=2.3.4, Cloud Dataproc image version=1.3, TensorFlow=2.2.0, Keras=2.4.3 and python=3). The cluster has Master node and many Worker nodes. Master node contains the YARN Resource Manager, HDFS Name Node, and The pre-processed CT-lung image data was stored in the cloud through Google Storage Bucket. We use Google Dataproc Cluster with the following versions (Debian=9, Hadoop=2.9, Apache Spark=2.3.4, Cloud Dataproc image version=1.3, TensorFlow=2.2.0, Keras=2.4.3 and python=3). The cluster has Master node and many Worker nodes. Master node contains the YARN Resource Manager, HDFS Name Node, and all job drivers. Master node has 4 cores CPU, 15 Gb memory and, 500 Gb disk size. Each worker node contains a YARN Node Manager, a HDFS Data Node and The HDFS replication factor is 2. And also, each worker node has 4 cores CPU, 15 Gb memory and, 500 Gb disk size.

To measure the performance, we used the Area under ROC curve (AUC) for comparing the proposed classifier which uses CNN with constructive deep neural network algorithm (CCG-DLNN) with the traditional CNN model. Fig.7 shows the ROC curve for the proposed architecture that used a CCG-DLNN algorithm with the CNN model is AUC=0.9810. How- ever, the ROC curve for the normal CNN model has AUC=0.9666.

Fig.8 shows the training speed for the proposed classifier. The highest throughput is when executing 10 executors and with 3 or 5 cores per executor with 9300 and 28030 respectively.

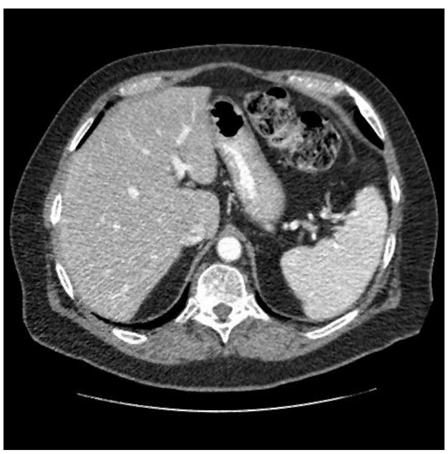
Fig.9 and Table 1 show that the training loss of the proposed framework which add the constructive algorithm CCG-DLNN to the CNN model, less than the traditional CNN model at all iterations.

Fig.10 shows the results of the training speedup of the proposed model for a different nodes number and cores at CPU. The results show that when the number of cores increased (from 10 to 100), the training time decreased.

5 CONCLUSION AND FUTURE WORK

This paper uses Apache Spark processing framework to analyze CT-lung image data. Apache-spark allows to solve iterative Machine Learning problems by using a distributed constructive deep learning module called Spark BigDL. The proposed approach com- bines the benefits of frameworks that used for big data processing (i.e. Spark) with the benefits of a constructive deep neural network algorithm called cascade-correlation growing Deep Learning Neural

(a)



(b)

Figure 6: (a) The Original CT Lung Image (b) CT Lung Image after performing median filter× with 3 3 window size.

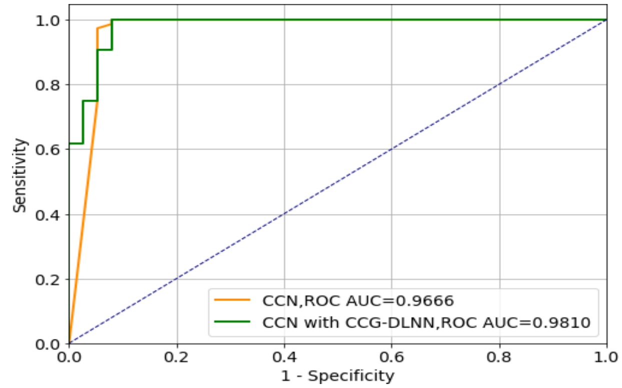


Figure 7: ROC curve for proposed architecture

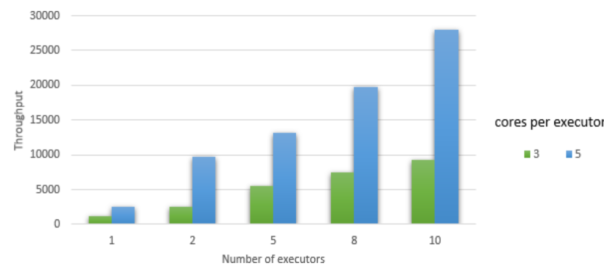


Figure 8: Throughput when changing the executor's number and number of cores at each executor for the proposed architecture

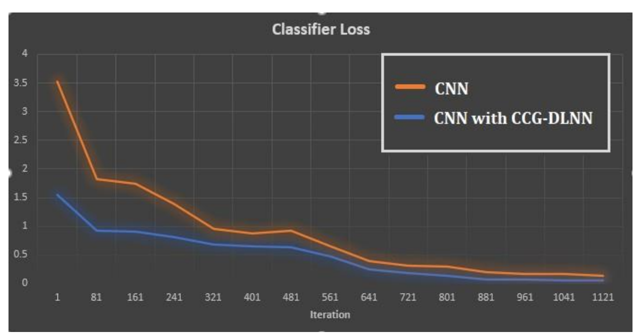


Figure 9: The comparison at loss rate between the traditional CNN model and the proposed framework

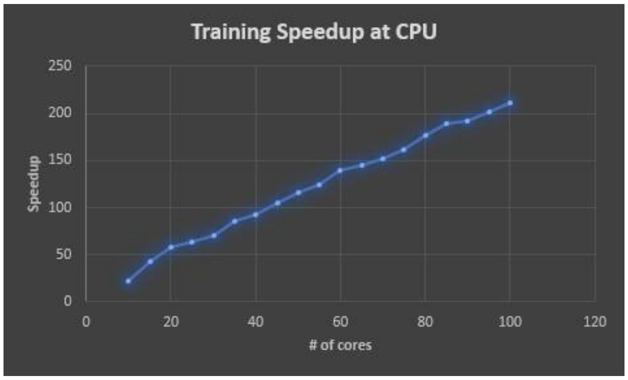


Figure 10: Training speed-up versus the numbers of CPU cores

Table 1: The comparison at loss rate between the traditional CNN model and the proposed framework

Iterations	CNN	CNN with CCG-DLNN
1	3.5	1.6
81	1.8	0.8
161	1.6	0.6
241	0.9	0.2
321	0.7	0.04
401	0.9	0.02
481	0.4	0.01
561	0.3	0.01
641	0.3	0.01
721	0.2	0.01
801	0.1	0.01
881	0.04	0.01
961	0.02	0.01
1041	0.01	0.01
1121	0.01	0.01

Network Algorithm (CCG-DLNN). The data is partitioned into small subsets. Each data partition enters as an input to each executor. At each executor, there is a copy of CNN and a constructive deep neural net- work which take the subset data as input and parallelize the processing of gradient descent. The server receives gradient delta from each executor and then synchronizes the model parameters between executors. Then, the output of each executor is combined. Our results showed that our model can increase to many of CPU cores.

The results show that the proposed architecture that used a CCG-DLNN algorithm with the CNN model is better than the traditional CNN model as the ROC curve for our proposed framework is AUC=0.9810. However, the ROC curve for the normal CNN has AUC=0.9666.

In the future, we expect that many of the tools will develop gradually. We can regard that the development will make improvements and will gain from the knowledge learned by a many number of developers.

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest.

REFERENCES

[1] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., et al.: Tensor- flow: Large-scale machine learning on hetero- geneous distributed systems. arXiv preprint arXiv:1603.04467 (2016)

[2] Alahmari, S.S., Cherezov, D., Goldgof, D.B., Hall, L.O., Gillies, R.J., Schabath, M.B.: Delta radiomics improves pulmonary nodule malig- nancy prediction in lung cancer screening. IEEE Access **6**, 77,796–77,806 (2018)

[3] Dai, J.J., Wang, Y., Qiu, X., Ding, D., Zhang, Y., Wang, Y., Jia, X., Zhang, C.L., Wan, Y., Li, Z., et al.: Bigdl: A distributed deep learning

framework for big data. In: Proceedings of the ACM Symposium on Cloud Computing, pp. 50– 60 (2019)

[4] Eberendu, A.C.: Unstructured data: an overview of the data of big data. International Journal of Computer Trends and Technology **38**(1), 46–50 (2016)

[5] Fahlman, S.E., Lebiere, C.: The cascade-correlation learning architecture. In: Advances in neural information processing systems, pp. 524–532 (1990)

[6] Hancock, M.C., Magnan, J.F.: Lung nodule malignancy classification using only radiologist- quantified image features as inputs to statisti- cal learning algorithms: probing the lung image database consortium dataset with two statistical learning methods. Journal of Medical Imaging **3**(4), 044,504 (2016)

[7] Haralick, R.M., Shanmugam, K., Dinstein, I.H.: Textural features for image classification. IEEE Transactions on systems, man, and cybernetics (6), 610–621 (1973)

[8] Harimoorthy, K., Thangavelu, M.: Multi-disease prediction model using improved svm-radial bias technique in healthcare monitoring system. Journal of Ambient Intelligence and Humanized Computing pp. 1–9 (2020)

[9] Hosny, A., Parmar, C., Coroller, T.P., Gross- mann, P., Zeleznik, R., Kumar, A., Bussink, J., Gillies, R.J., Mak, R.H., Aerts, H.J.: Deep learning for lung cancer prognostication: A ret- rospective multi-cohort radiomics study. PLoS medicine **15**(11), e1002,711 (2018)

[10] Jan, B., Farman, H., Khan, M., Imran, M., Is- lam, I.U., Ahmad, A., Ali, S., Jeon, G.: Deep learning in big data analytics: A comparative study. Computers & Electrical Engineering **75**, 275–287 (2019)

[11] Kaucha, D.P., Prasad, P., Alsadoon, A., El- chouemi, A., Sreedharan, S.: Early detection of lung cancer using svm classifier in biomedical image processing. In: 2017 IEEE International Conference on Power, Control, Signals and In- strumentation Engineering (ICPCSI), pp. 3143– 3148. IEEE (2017)

[12] Khan, M.A., Karim, M., Kim, Y., et al.: A two-stage big data analytics framework with real world applications using spark machine learning and long short-term memory network. Symme- try **10**(10), 485 (2018)

[13] Kim, B.C., Sung, Y.S., Suk, H.I.: Deep feature learning for pulmonary nodule classification in a lung ct. In: 2016 4th International Winter Conference on Brain-Computer Interface (BCI), pp. 1–3. IEEE (2016)

[14] Kingsbury, N.G.: The dual-tree complex wavelet transform: a new technique for shift invariance and directional filters. In: IEEE digital signal processing workshop, vol. 86, pp. 120–131. Cite- seer (1998)

[15] Makkie, M., Huang, H., Zhao, Y., Vasilakos, A.V., Liu, T.: Fast and scalable distributed deep convolutional autoencoder for fmri big data an- alytics. Neurocomputing **325**, 20–30 (2019)

[16] Sazzad, T.S., Ahmmed, K.T., Hoque, M.U., Rahman, M.: Development of automated brain tumor identification using mri images. In: 2019 International Conference on Electrical, Com- puter and Communication Engineering (ECCE), pp. 1–4. IEEE (2019)

[17] Scherer, D., Müller, A., Behnke, S.: Evaluation of pooling operations in convolutional architec- tures for object recognition. In: International conference on artificial neural networks, pp. 92– 101. Springer (2010)

[18] Sujitha, R., Seenivasagam, V.: Classification of lung cancer stages with machine learning over big data healthcare framework (2020)

[19] Taher, F., Werghi, N., Al-Ahmad, H.: Rule based classification of sputum images for early

lung cancer detection. In: 2015 IEEE International Conference on Electronics, Circuits, and Systems (ICECS), pp. 29–32. IEEE (2015)

[20] Tu, N.A., Huynh-The, T., Wong, K.S., Bui, D.M., Lee, Y.K.: Distributed feature extraction on apache spark for human action recognition. In: 2020 14th International Conference on Ubiq- uitous Information Management and Communi- cation (IMCOM), pp. 1–6. IEEE (2020)

[21] Tun, K.M.M., Khaing, A.S.: Implementation of lung cancer nodule feature extraction using dig- ital image processing (2014)

[22] Zhang, Y., Wang, S., Zhao, H., Guo, Z., Sun, D.: Ct image classification based on convolutional neural network. NEURAL COMPUTING & APPLICATIONS (2020)

Prediction of Fleet Demand Needs Using Backpropagation Artificial Neural Networks and Fuzzy Time Series in Sea Release Transport System

Nur Hijriah Zubaedah Narang
Department of Electrical Engineering
University of Brawijaya
Malang, East Java, Indonesia

Rahmadwati
Department of Electrical Engineering
University of Brawijaya
Malang, East Java, Indonesia

Erni Yudaningtyas
Department of Electrical Engineering
University of Brawijaya
Malang, East Java, Indonesia

Abstract: The number of service users or consumers in choosing the marine ship mode as a means of transportation between Kupang City and Rote Ndao Regency can help service users related to inter-island trading activities, serving shipments of commodities and manufactured goods and sectors. The ability to predict quickly, precisely and accurately the demand for fleet demand is very important for service providers or ship service users. As a case in point, errors in predicting demand needs can result in planning allocations for both additional fleets and planning needs for scheduling operations for marine transport traffic.

This study discusses the prediction of fleet demand in the ferry system by taking time series research data. The methods used include Backpropagation ANN and Fuzzy Times Series. The results obtained show that the performance of the backpropagation method which is formed from training data and validated on the testing data provides a fairly good level of prediction accuracy with a mean square error (MSE) value of 0.016, while with the FTS method the MSE value is 0.55.

Keywords: Backpropagation ANN, Fuzzy Times Series, Mean Square Error, Demand, Forecasting

1. INTRODUCTION

Times series data can be studied as a basis for predicting fleet demand by looking for trend patterns in the form of graphs that go up or down over a long period of time. One way to look for trend patterns is by using the Artificial Neural Network (ANN) Backpropagation technique. Backpropagation technique makes predictions based on the learning process of a set of input output patterns. In addition to the ANN method, the prediction of fleet demand needs with time series research data can also use the Fuzzy Time Series (FTS) method to compare with the Backpropagation method. The FTS concept uses the relationship between historical data in the past and data that is actual. And by using the FTS (forecasting) prediction calculation on the data, the prediction results are obtained. Previous research in overcoming forecasting problems using the Backpropagation ANN method has been widely carried out. One of them is to predict the number of newspaper requests by Nabilla Putri Sakinah, et al. In 2018. Based on the test results, the best number of iterations is 200, and the best learning rate is 0.6, and testing training data and test data produces the best training data values. that is 100 and the test data 10. So that the smallest error rate is 0.0162. In addition, previous research using the FTS method to solve forecasting problems was M Angga Prasetya A in 2018 to predict the inflation rate based on expenditure groups with an average time based fuzzy time series algorithm. The test results obtained were the average MSE value of 0.486 was recorded for month 15, MSE 0.335 was recorded in year 3 and finally MSE 0.336 was assessed as a divisor of 1.9 for consecutive month data categories and MSE 0.336 was assessed as divider 2 for data categories for consecutive years. The two studies that have been done above use different times series data so that the prediction results obtained cannot be used as a basis for comparing the performance of the two prediction methods used. Therefore it is necessary to compare the two methods to see which method has the best performance. While research on the problem of predicting the demand for fleet demand has been carried out as one example, namely the study of demand for crossing transportation based on demand in Bau - Bau and Wa Ara Regencies using the regression analysis method by Ria Rahmarie Cangara in 2012. The results of the study In the Bau-Bau - Wa Ara route, the production of shiploads becomes variable Y (not free) and variable x (free) is the population and economic growth, the R-square is 0.843 and significant F is 0.027 In addition, other research from Adris A. Putra in 2013 is about forecasting the balance of the number of public transport fleets based on passenger demand. The results of this study using a quantitative load factor analysis method (the ratio between the number of passengers and the seating capacity at a certain time unit) is the number of public transport fleets that should be available according to the amount of passenger demand in the city of Makassar as many as 2.283 units of vehicles or 51.5% of the total number of vehicles. 4,511 existing vehicles. This condition shows that the number of fleets is not balanced with the large demand for passengers (over supply). The second example of the forecasting case study above using the linear regression method and the load quantitative analysis still has weaknesses. The linear regression method can only be used for linear analysis while the load factor method is less flexible because it can only be used to determine the availability of fleet load capacity. To overcome the weaknesses of the regression analysis method and the quantitative load factor, the Backpropagation and FTS methods are used which can analyze non-linearly and are flexible to complex functions.

2. RELATED WORK

Backpropagation is a part of ANN which is a gradient descent method to minimize the squared output error. The characteristic of Backpropagation is to minimize errors in the output generated by the network. In the forecasting process with the concept of the FTS model, the value used is the value of the fuzzy set based on real numbers. The stages in the FTS forecasting algorithm are as follows:

1. Calculate the value of the set universe and the value of the length of the interval
2. Define the upper and lower limits
   $u_i = [(D_{min} + ((i-1) * range), (D_{min} + (i * range))]$
3. Form a fuzzy set
4. Fuzzification process
5. FLR dan FLRG
6. Defuzzification

Then calculate the Mean Squared Error (MSE) of the two systems created using the following equation:

$$MSE = \sum \frac{(t - y_k)^2}{N}$$

3. METHOD

3.1 Research Data

In the early stages the data will be normalized first by finding the maximum and minimum values from the dataset. Furthermore, the dataset is divided into two data, namely 80% training data (training data) and 20% test data (testing data). Then the training data is carried out by the training process to get the model pattern and then the model pattern is tested using test data with the Backpropagation and FTS methods to get the predictive results of the two methods. The results of this testing process are then evaluated using the Mean Square Error (MSE) method with the parameter of the level of accuracy of the prediction results of the two methods.

3.2 The Research Stage

The research stage is a process of a sequence of steps that are interrelated with one another to obtain the results being researched.

4. Normalization Data

The data must first be normalized with the aim that each data has a value between 0 and 1 so that it is more balanced in determining the value of information contained in a data.

5. Variables and Data Analysis

The research begins with a literature study related to the application of artificial neural networks and fuzzy time series to predict fleet demand requirements. Based on the data obtained, data analysis was carried out using the Backpropagation and FTS methods to determine the relationship between input and output.

a. Feedforward Process
   The process in ANN begins by adding up the multiplication between the input and the existing weight and then calculating the activation value which is then used as input by the layer above it is called the feedforward process.
b. Backpropagation Process
   The process for calculating the error information for each neuron in each layer starts from errors in the output layer to the hidden layer.
c. FTS Algorithm Process
   Forecasting steps using the FTS method are as follows:

   Input the demand data for the fleet of expenditure categories, then create a universe set and interval, then identify the sub-universe set. The result will describe the fuzzy membership function. The next step is Fuzzification with the resulting fuzzy set results then Determine the FLR of the fuzzification results, FLRG then FLRG Defuzzification which in the end will get the forecast results. MSE will calculate the error value from the forecasting results as the output of the FTS learning algorithm process.

3.3 Manual Calculation

1. ANN Backpropagation calculation

In this manual calculation, industrial data from 2008 to 2019 will be used. Training data and testing data ragged.

Table 1. Training Data for Fleet Demand Needs

Years									
2008	2009	2010	2011	2012	2013	2014	2015	2016	2017
7900	7911	7987	7947	7836	8206	8052	8072	7906	7990

Table 2. Testing Data for Fleet Demand Requirements

Years	
2018	2019
7962	7922

Before the training process is carried out, the training data used must be normalized first. The min and max values of the training data are 6407 and 44520, respectively. Using the following equation:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Table 3. Results of Training Data Normalization

Years									
2008	2009	2010	2011	2012	2013	2014	2015	2016	2017
0.39 10	0.0 396	0.0 416	0.0 405	0.0 376	0.0 473	0.0 433	0.0 438	0.0 394	0.04 16

In this study, the network architecture used is 4-2-1, namely 4 input layer neurons, 2 hidden layer neurons and 1 output layer neuron. By means of a random value between -0.5 and 0.5 which represents the weight of the artificial neural network.

Table 4. Weight values at the Output Layer

Results of Nguyen Widrow Initialization		W_1	W_2
	Z_1	-0.4199	0.1806
	Z_2	-0.4558	0.1450
	Z_3	-0.4919	0.1094
	Z_4	-0.5281	0.0741

Table 5. Feedforward Output Layer Results

Iterasi	Data	$v_{kj} * y_j$		y_{net_k}	y_k
1	1	- 0.2017	0.0861	0.8843	0.7077
….	….	….	….	….	….
4	10	0.6417	-0.2307	-1.2312	0.2259

After doing the feedforward process, the next step is the backpropagatio nprocess. In the final stage of backpropagation is to calculate new weights on the hidden layer using the previously obtained weight change values:

Table 6. Calculation Results *wkj*

iterasi	Data	Calculation Results w_{11} (baru)		
Iterasi 1	Data 1	w_{11}	w_{12}	bias
		-0.4558	0.1450	0.9254
….	….	…..	….	…..
Iterasi 4	Data 10	w_{11}	w_{12}	bias
		-1.1319	-0.4760	-0.3761

Furthermore, MSE calculations are carried out to determine the error level of network learning.

Table 7. Calculation of MSE

Iterasi	MSE
1	0.3123
2	0.2852
3	0.1050
4	0.0465

2. Fuzzy Times Series calculations

Fragment of demographic data on demand for demand from 2008 to 2019.

Table 8. Population Data

No	Years	Total population
1	2008	100251
2	2009	103521
3	2010	106553
4	2011	108283
5	2012	108731
6	2013	114160
7	2014	116586
8	2015	119381
9	2016	120887
10	2017	124014
11	2018	127341
12	2019	132570

The FTS process begins by defining the set of universes U using the difference in the time series data

$$U = [\,-V1, V_{max} + V2\,]$$

Next, divide the universe U into partitions or equal parts

$$U = \{u_1, u_2, \ldots, u_n\}$$

$$B = \frac{U_{max} - u_{min}}{n}$$

$$u_i = [u_{ib}, u_{ia}]$$

$$u_{ia} = U_{ib} + b$$

$$u_{(i+1)b} = u_{ia}$$

Next, determine Fuzzy Logical Relationships (FLR) by connecting the fuzzification results

Table 9. Fuzzy Logical Relationships

$A_1 \to A_5$	$A_3 \to A_2$
$A_2 \to A_2$	$A_3 \to A_3$
$A_2 \to A_3$	$A_4 \to A_3$
$A_3 \to A_1$	$A_5 \to A_4$
$A_1 \to A_5$	$A_3 \to A_2$
$A_2 \to A_2$	$A_3 \to A_3$
$A_2 \to A_3$	$A_4 \to A_3$
$A_3 \to A_1$	$A_5 \to A_4$

Then form the FLRG (Fuzzy Logical Relation Groups) from the previously obtained FLR by grouping the same left side into one.

Table 10. Fuzzy Logical Relationships Group

$A_1 \to A_5$
$A_2 \to A_2,\ A_3$
$A_3 \to A_1, A_2, A_3$
$A_4 \to A_3$
$A_5 \to A_4$

predicting the difference between t and t-1 by defusing the fuzzy output. he centroid method multiplies each membership in the FLRG that has been unified by the midpoint on the partitions or parts that correspond to the midpoint then the results are added, then the sum is divided by the sum of the membership values of the FLRG membership.

Table 11. Midpoint Value of Each Partition

Interval	Upper limit	Lower limit	Midpoint
u_1	-6000	-5800	-5900
u_2	-5800	-5600	-5700
u_3	-5600	-5400	-5500
u_4	-5400	-5200	-5300
u_5	-5200	-5000	-5100

For example, the prediction of the difference in 2017 that will be sought. See the fuzzy set (Ai) and the FLRG from the actual difference in 2016. The actual difference in 2016 is included in the A3 fuzzy set with the FLRG can be seen in Table 10, namely A3 -> A1, A2, A3 then the fuzzy set- these are:

$$A_1 = \{1/u_1 + 0.5/u_2 + 0/u_3 + 0/u_4 + 0/u_5\}$$

$$A_2 = \{0.5/u_1 + 1/u_2 + 0.5/u_3 + 0/u_4 + 0/u_5\}$$

$$A_3 = \{0/u_1 + 0.5/u_2 + 1/u_3 + 0.5/u_4 + 0/u_5\}$$

Then combined:

$$A_1 \cup A_2 \cup A_3 = \{1/u_1 + 1/u_2 + 1/u_3 + 0.5/u_4 + 0/u_5\}$$

Predicted difference in 2017:

$$V_{2017} = \frac{1*(-5900) + 1*(-5700) + 1*(-5500) + 0.5*(-5300) + 0*(-5100)}{1+1+1+0.5+0}$$

$$V_{2017} = \frac{-22340}{3.5}$$

$$V_{2017} = -6400$$

So the predicted value of demand in 2017 is

120887 + (-6400) = 114487

The MAPE equation calculates the error rate in percentage terms. Here is the formula:

$$MAPE = \left(\frac{1}{n}\sum \frac{|x-y|}{x}\right) * 100\%$$

Therefore, the values that are entered into the equation are the actual and predicted values in 2010 to 2019. So that the MAPE value is 0.063%.

4. RESULT AND DISCUSSION

The maximum iteration test is used to determine the maximum number of iterations. The graph of the results of testing the maximum number of literacy against the MSE value is shown in Figure 1 as follows:

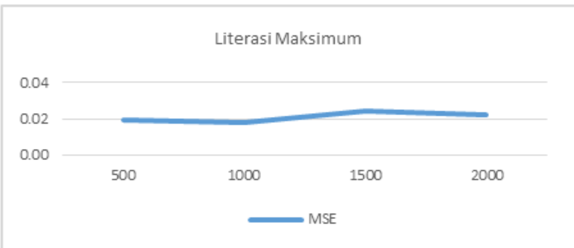


Figure 1. Testing Graph of Maximum Number of Iterations Against MSE Value

Testing is done by changing the learning rate (alpha) value from 0.1 to 0.9. Each learning rate value is tested with differentinitial weights. The graph of the results of the Learning rate test against MSE is shown in the Figure below:

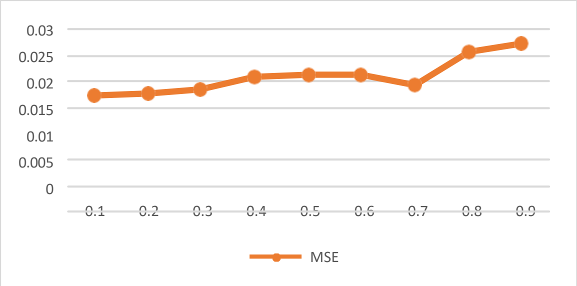


Figure 2. Graph of Learning Rate Testing Results

After the above tests were carried out, the results were that the best number of iterations and learning rates were 500 and 0.1. The next test is testing the amount of training data 20, 40, 60, 80, and 100. While the number of test data used is 90, 70, 50, 30, and 10. The graph of the test results on the number of Training Data and Test Data on MSE is shown in the Figure below:

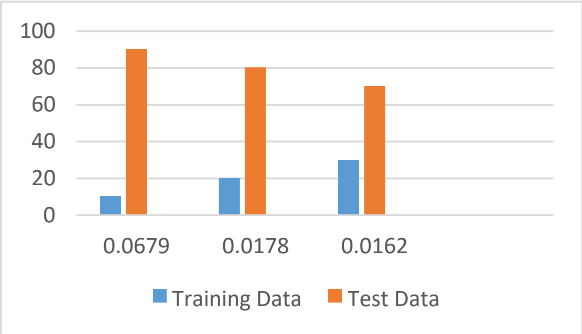


Figure 3. Graph Testing the Amount of Training Data and Test Data

Basically, from Dmin and D_{max} we can get the definition for the set of universe U. Then defining each fuzzy set Ai with $1 \leq i \leq 7$. Furthermore, testing the accuracy of the prediction system aims to find out the accuracy of the Fuzzy Time Series system for prediction cases of fleet demand requirements.

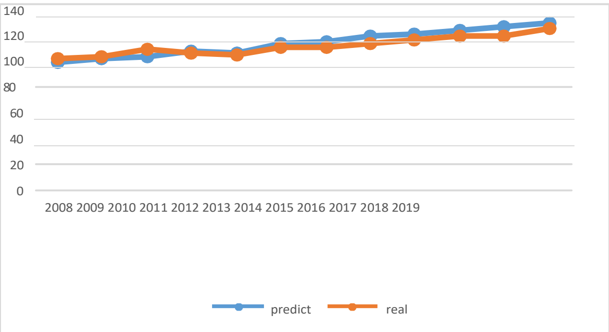


Figure 4. Comparison Graph of FTS Real Value and Prediction

5. CONCLUSION

Based on the analysis of the test results of the fleet demand prediction system using the Backpropagation and Fuzzy Time Series methods, it can be concluded:

1. Forecasting using the Bckpropagation method results in a smaller MSE value when compared to FTS
2. The performance of the back propagation method which is formed from training data and validated onthe testing data provides a fairly good level of prediction accuracy with a mean square error (MSE)) .016 While the FTS method obtained an MSE value of 0.55
3. The network architecture used in this study cannot be said to be the best result considering that there are still many possible combinations used, both data input patterns, the number of neurons in the hidden layer, the maximum number of epochs and the target MSE value. This is because the selection of these parameters must be done by trial and error. Likewise with the FTS algorithm, the forecasting process needs to be tested with different sample numbers of calendar year inflation.experts who have contributed towards development of the template.

6. REFERENCES

[1] Angga Dwi Apria Rifandi.,2017. Optimasi Interval Fuzzy Time Series Menggunakan Particle Swarm Optimization Pada Peramalan Permintaan Darah: Studi Kasus Unit Transfusi Darah Cabang-PMI Kota Malang. Fakultas Ilmu Komputer.Universitas Brawijaya.Malang

[2] A V. Anufrenko,dkk.2017.Diagnostik rute transportasi jaringan data menggunakan

[3] Aziz Rudi, Asrul, Risman.,2016. Model Pemilihan Moda Transportasi Laut Pada Wilayah Geografis Kepulauan;Studi Kasus Rute Kota Kendari Kabupaten Konawe Kepulauan Provinsi Sulawesi Tenggara: Jurnal Perencanaan Wilayah Dan Kota Vol.27,No.2,PP.151-172

[4] Cangara. R., 2012. Studi Kebutuhan angkutan penyeberangan berdasarpermintaandiprovinsiSulawesitenggara:Makasar: UNHAS

[5] Chen,S.M., 1996. Forecasting Enrollments with Fuzzy Time Series. Fuzzy sets and System 81,pp.311-319

[6] Dwi Prisita Anggriningrum.,2013. Perbandingan Prediksi Harga Saham Dengan Menggunakan Jaringan Syaraf Tiruan Backpropagation Dan Arima. UJM 2 (2) (2013)Unnes Journal of Mathematics

[7] Fausett, L.V., 1994. Fundamentals Of Neural Network: Architectures, Algorithms and Applications, New Jersey: Prentice Hall

[8] Guoyuan Li,dkk.2017.Pemodelan dan analisis berbasis jaringan saraf untuk memprediksi waktu pergerakan kapal.Norwegian University of Science and Technology

[9] Haviludin.,2016. Prediksi Kedatangan Turis Asing ke Indonesia Menggunakan Backpropagation Neural Networks. Jurnal Teknologi dan Sistem Komputer, 4(4), 2016, 485-490

[10] Iraklis Lazakis, dkk.2017.Prediksi kondisi sistem mesin kapal melalui metode analitis dan jaringan saraf tiruan.University of Strathclyde

[11] M Angga Prasetya A.,2018. Implementasi Algoritme Average Time Based Fuzzy Time Series Untuk Peramalan Tingkat Inflasi Berdasarkan Kelomk

International Journal of Computer Applications Technology and Research
Volume 9–Issue 12, 327-331, 2020, ISSN:-2319–8656

Optimization of Missing Value Data Imputation Automatic Dependent Surveillance Broadcasting (ADS-B) Based on K-Nearest Neighbor and Genetic Algorithm

Didik Hariyanto
Electrical Engineering Department,
Faculty of Engineering,
Brawijaya University.
Malang, East Java, Indonesia

Sholeh Hadi Pramono
Electrical Engineering Department,
Faculty of Engineering,
Brawijaya University
Malang, East Java, Indonesia

Erni Yudaningtyas
Electrical Engineering Department,
Faculty of Engineering,
Brawijaya University
Malang, East Java, Indonesia

Abstract: The flight navigation equipments technology use still conventional, namely using radar, now slowly starting to switch to Automatic Dependent Surveillance-Broadcast (ADS-B [6]. In this study, using RTL-SDR to detect aircraft and carry out tests through the Monte Carlo alltitude method, latitude, and longitude only [3]. However, in this system there is a problem regarding the missing value in the preprocessed data results / ADS-B flow data. In handling missing values, the KNN method is the most popular, but the weakness in the KNN method, can reduce the performance[9]. So a Genetic Algorithm (GA) is proposed to optimize the k value in the KNN method. The results of this study obtained a better MSE value in the imputation process. Altitude k = 3, with MSE 128668.96, Speed k = 6, with the MSE value = 457.5201, while the k value in the Heading variable k = 61 with MSE = 752.1429. For Lattitude and Longitude, the value of k = 3, MSE 9.16E-05 and k = 2 and MSE 1.68E-05.

Keywords: ADS-B, Missing Value, Imputation, K-Nearest Neighbor, Genetic Algotirhm

1. INTRODUCTION

The Air transportation safety is an important and major factor in the operation of flight services including flight navigation services. Meanwhile, flight navigation services can be provided maximally by airport operators when supported by good airport facilities. Along with technological developments that are increasingly sophisticated day after day, supporting facilities for flight navigation services are growing rapidly. The use of aviation navigation equipment technology, which was initially still conventional, namely radar, is now slowly starting to switch to Automatic Dependent Surveillance-Broadcast (ADS-B) [1].

With the development of communication technology, one way to get ADS-B data is use a Software-Defined Radio (SDR), namely the Mini USB RTL-SDR receiver using a new IC tuner, the R820T2 made by Rafael Micro. SDR technology was first introduced in 1991 by Joseph Mitola. RTL-SDR has a wide frequency range, with frequencies ranging from 25 Mhz - 1750Mhz. So that you can listen to all radio activities in that range and in the form of other data [2].

By integrating RTL-SDR, and dump 1090 as decoder software on the Rasbery Pi3, the ADS-B receiver is relatively more efficient and inexpensive[5]. The Research conducted by Akshay, N et al in 2017, which produced a fairly complete ADS-B flight data, was carried out using the RTL-SDR. The results of this study are information on altitude, position, speed, direction, and other information to ground stations and other aircraft. In this study, using RTL-SDR only to detect aircraft and carry out tests through the Monte Carlo method of latitude, longitude and latitude [3]. However, in the system, there is a problem about missing value in the results of the preprocessing data / ADS-B flow data.[6]

Imputation is filling in the missing value (empty data) with a certain value. [7]The rule of imputation is to get the predicted value as close as possible to the missing value, in other words imputation tries to minimize the value between the missing value and the predicted value of the missing value [4].

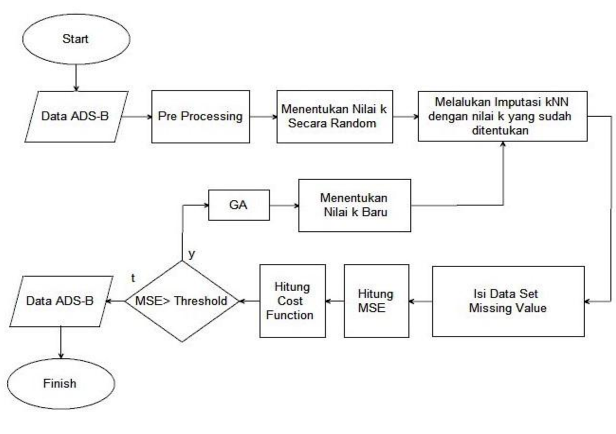
In handling missing data, KNN is the easiest and most popular method. However, this method has several drawbacks, one of which is that incorrect selection of *k* values can reduce classification performance [9]. So that a genetic algorithm is proposed to optimize the k value in the KNN so that it can produce a good estimated value with the smallest possible MSE[8]. This, the classification results will be obtained with high accuracy. This study aims to deal with missing data with imputation techniques using a combination of the KNN and GA (KNN-GA) algorithms.

Based on the previous studies, in this study an alternative solution that can be given in dealing with ADS-B data which includes heading, speed, longitude, altitude, latitude where missing values are found is by means of imputation. Meanwhile, to overcome the weaknesses of the k-NN method, namely by increasing the value of k using Genetic Algorithm (GA). By conducting research on the imputation of missing values, it is hoped that the data will be more accurate and correct, so that the information that will be provided to ATC in carrying out its duties as air traffic guides has the integrity of the information.

2. METHOD

The concept of the method in this study is shown in Figure 1 below.

Figure 1. Research Method



2.1 Data Collection

The data collection mechanism uses a model like Figure 2 below.

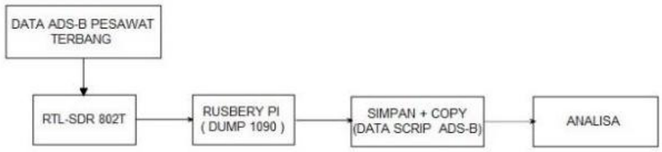


Figure 2. ADS-B Data Retrieval

1) Aircraft ADS-B data is taken by making a groundstation consisting of colinear antennas, RTL-SDR and Rasberry Pi 3 which are integrated with Linux OS and dump1090 decoder whose function is to convert analog data into binary form.
2) After the data is collected and stored in the .csv file, then save the ADS-B data in the memory embedded in the Rasberry Pi 3
3) Copy the ADS-B data file to a PC / laptop for data analysis.
4) Convert ADS-B data to normalize data.

2.2 *k-Nearest Neighbor (*K-NN) Method

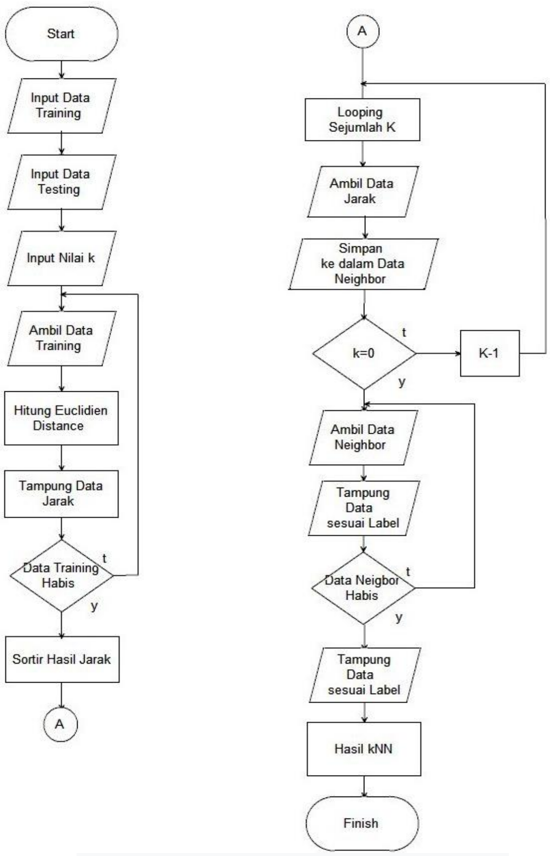


Figure 3. Flow diagram of the KNN Method

From Figure 3 it can be explained that the K-Nearest Neighbor Method Flowchart is as follows:

1) Take the missing value data from the .csv file, and sort the valuable data into training data
2) Take blank data to be used as testing data from the .csv file
3) Determine the initial K value for the KNN algorithm
4) Of the many training data, take the training data one by one to be compared with the relevant testing data
5) Calculating the distance from training data and testing data with the Euclidean Distance formula
6) The results from this distance are accommodated in certain variables so that they can be processed
7) If the training data is still not finished, take new training data and recalculate the value of Euclidean Distance
8) Of these data. If the training data has been taken out, then proceed to the previous process.
9) Sorting distance storage variables from the lowest value to the farthest distance, this is used because basically KNN looks for the data closest to the training data
10) The results of the distance are not taken all but a number of K values are taken. To retrieve this data, it is necessary to loop a predetermined number of K values.
11) Retrieving distance data one by one in sequence.
12) Save the retrieved data into neighbor variables.
13) If the value of k is still not used up, then reduce the value of K by 1 and return the looping value to k. If the value of K has run out then continue to the next process
14) Taking on the neighbors data one by one.
15) Hold the neighbors data according to the data label. Data label is a classification of data or output data.
16) If the neighbors' data has not run out, then take the neighbors data continuously, if it runs out then continue to the next process.
17) Hold labeled data into variables.
18) The results of the KNN are in the first serial number data with a predefined label.

2.3 Genetic Algorithm (GA) Method

The flow diagram work steps of the Genetic Algorithm (GA) are as follows in Figure 4:

1) Determining the GA input value, the GA input is the data before the missing value is found and the data after
2) Missing value. the data value which will later also be a benchmark for fitness values
3) Determine the number of population. The number of population is obtained from the solution per population (chromosome) and the number of input data (gene)
4) Because at the beginning of the iteration the best value is still unknown, the value of the population is a random value with predetermined limits.
5) Determine the number of generations or the number of iterations. The number of early iterations is used for how long the algorithm is executed, the more generations the algorithm will run, but the resulting data can be better.
6) Calculating the fitness value or cost function. This function is used to find out how suitable / good the data is in the population
7) The results of the fitness value are accommodated in a variable. This variable will be filtered again to get the most optimal value
8) From the fitness data, take the best data to be used as a parent (parent). Parent data is taken more than 2 data
9) Parent data is combined with a crossover function. This crossover function divides the parent data in half. The first half of the parent data is taken to be combined with the second half of the parent data.
10) Mutation process is used to prevent the same data on the population. From the crossover data, add the random value.
11) The results of the mutation data and parent data enter the new population
12) Looping to step 5 if the generation is not yet 0
13) Recalculate the fitness value of the population formed from iterations.
14) The data that has the highest fitness value will be taken for the results of the GA algorithm

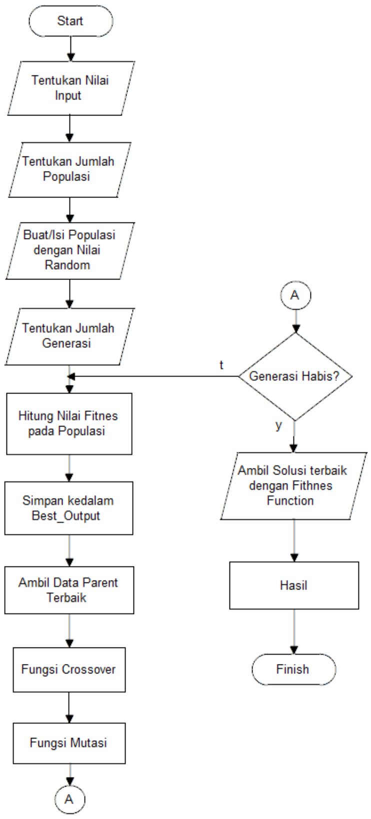
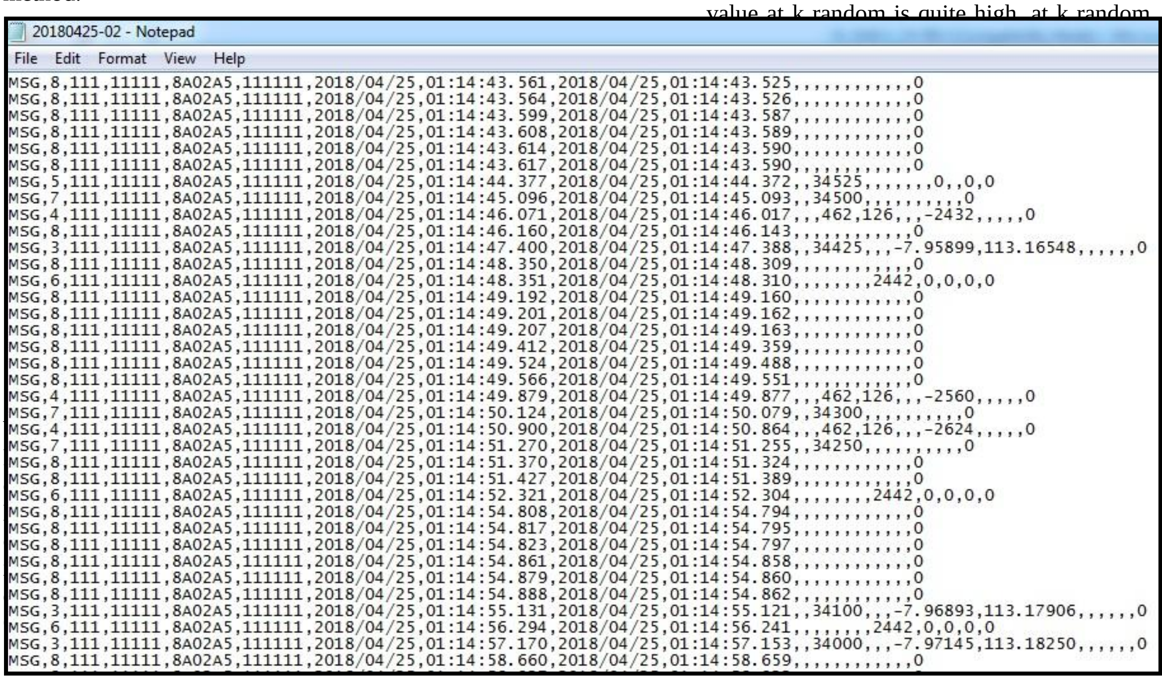


Figure 4. Flowchart of the Genetic Algorithm (GA) Method

3. RESULT AND DISCUSSION

3.1 Data Collection Results

In conducting the research, the researchers collected ADS-B data on civil aircraft at Abdulrahman saleh airport which was taken by making a ground station consisting of collinear antennas, RTL-SDR and Rasberry Pi 3 which have been integrated with Linux OS and dump1090 decoder whose function is to convert analog data into binary form. After the data is collected and stored in the form of a .csv file, then storing the ADS-B data in the memory embedded in the Rasberry Pi 3. Copying the ADS-B data file to a laptop for data analysis. Convert ADS-B data to normalize data. Tried on ADSB data. The following is the data acquisition result in Figure 7. then in this research a method for filling in the blank data will be presented by trying to use the KNN method and optimizing using the GA method.


20180425-02 - Notepad
File Edit Format View Help

```
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:43.561,2018/04/25,01:14:43.525,,,,,,,,,,,,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:43.564,2018/04/25,01:14:43.526,,,,,,,,,,,,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:43.599,2018/04/25,01:14:43.587,,,,,,,,,,,,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:43.608,2018/04/25,01:14:43.589,,,,,,,,,,,,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:43.614,2018/04/25,01:14:43.590,,,,,,,,,,,,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:43.617,2018/04/25,01:14:43.590,,,,,,,,,,,,0
MSG,5,111,11111,8A02A5,111111,2018/04/25,01:14:44.377,2018/04/25,01:14:44.372,,34525,,,,,,0,,0,0
MSG,7,111,11111,8A02A5,111111,2018/04/25,01:14:45.096,2018/04/25,01:14:45.093,,34500,,,,,,,,,,0
MSG,4,111,11111,8A02A5,111111,2018/04/25,01:14:46.071,2018/04/25,01:14:46.017,,,462,126,,,-2432,,,,,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:46.160,2018/04/25,01:14:46.143,,,,,,,,,,,,0
MSG,3,111,11111,8A02A5,111111,2018/04/25,01:14:47.400,2018/04/25,01:14:47.388,,34425,,,-7.95899,113.16548,,,,,,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:48.350,2018/04/25,01:14:48.309,,,,,,,,,,,,0
MSG,6,111,11111,8A02A5,111111,2018/04/25,01:14:48.351,2018/04/25,01:14:48.310,,,,,,,,2442,0,0,0,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:49.192,2018/04/25,01:14:49.160,,,,,,,,,,,,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:49.201,2018/04/25,01:14:49.162,,,,,,,,,,,,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:49.207,2018/04/25,01:14:49.163,,,,,,,,,,,,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:49.412,2018/04/25,01:14:49.359,,,,,,,,,,,,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:49.524,2018/04/25,01:14:49.488,,,,,,,,,,,,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:49.566,2018/04/25,01:14:49.551,,,,,,,,,,,,0
MSG,4,111,11111,8A02A5,111111,2018/04/25,01:14:49.879,2018/04/25,01:14:49.877,,,462,126,,,-2560,,,,,0
MSG,7,111,11111,8A02A5,111111,2018/04/25,01:14:50.124,2018/04/25,01:14:50.079,,34300,,,,,,,,,,0
MSG,4,111,11111,8A02A5,111111,2018/04/25,01:14:50.900,2018/04/25,01:14:50.864,,,462,126,,,-2624,,,,,0
MSG,7,111,11111,8A02A5,111111,2018/04/25,01:14:51.270,2018/04/25,01:14:51.255,,34250,,,,,,,,,,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:51.370,2018/04/25,01:14:51.324,,,,,,,,,,,,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:51.427,2018/04/25,01:14:51.389,,,,,,,,,,,,0
MSG,6,111,11111,8A02A5,111111,2018/04/25,01:14:52.321,2018/04/25,01:14:52.304,,,,,,,,2442,0,0,0,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:54.808,2018/04/25,01:14:54.794,,,,,,,,,,,,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:54.817,2018/04/25,01:14:54.795,,,,,,,,,,,,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:54.823,2018/04/25,01:14:54.797,,,,,,,,,,,,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:54.861,2018/04/25,01:14:54.858,,,,,,,,,,,,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:54.879,2018/04/25,01:14:54.860,,,,,,,,,,,,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:54.888,2018/04/25,01:14:54.862,,,,,,,,,,,,0
MSG,3,111,11111,8A02A5,111111,2018/04/25,01:14:55.131,2018/04/25,01:14:55.121,,34100,,,-7.96893,113.17906,,,,,,0
MSG,6,111,11111,8A02A5,111111,2018/04/25,01:14:56.294,2018/04/25,01:14:56.241,,,,,,,,2442,0,0,0,0
MSG,3,111,11111,8A02A5,111111,2018/04/25,01:14:57.170,2018/04/25,01:14:57.153,,34000,,,-7.97145,113.18250,,,,,,0
MSG,8,111,11111,8A02A5,111111,2018/04/25,01:14:58.660,2018/04/25,01:14:58.659,,,,,,,,,,,,0
```

Table 1. Missing Value on ADSB data for AbdulrahmanSaleh Airport.

Time	*Fligth*	*Variabel*				
		Altitude	*Speed*	*Heading*	*Latitude*	*Longitude*
Missing value						
12:25:10 AM	Sjy3131	34975			-7.80826	112.96259
12:25:12 AM	Sjy3131	?	?	?	?	?
12:25:12 AM	Sjy3131	35000	?	?	?	?
12:25:13 AM	Sjy3131	?				
12:25:15 AM	Sjy3131	?	459	126		
12:25:16 AM	Sjy3131	34975			-7.8157	112.97274
12:25:16 AM	Sjy3131		459	126		
12:25:17 AM	Sjy3131		?	?	?	?
12:25:17 AM	Sjy3131	35000	?	?	?	?
12:25:18 AM	Sjy3131		458	126		
12:25:19 AM	Sjy3131	35000			-7.81947	112.97791
12:25:20 AM	Sjy3131	35000			-7.82073	112.97961
12:25:20 AM	Sjy3131		458	126		

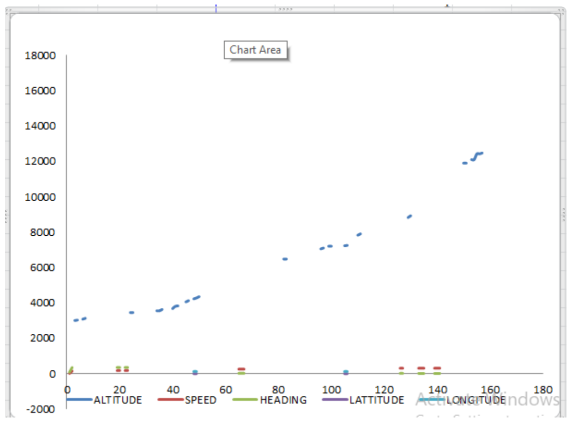


Figure. 6. Graphic display of Sriwijaya Air aircraft data which is still missing value

3.2 Imputation Results with the KNN Method

In Table 2 shows the MSE value of imputation results with the KNN method using the Euclidean distance at k = 2. Experiments were carried out on five variables with different percentages of missing. From this table, it can be seen that for each k random given the resulting MSE value tends to increase. The Altitude variable at the value of k = 2 results in a small MSE. Meanwhile, the best MSE heading variable is obtained at k = 2 with MSE 772.1429469. From the results of the table above, there is an increase in the MSE value at k random is quite high, at k random. An increase in
ed at all the k
also seemed to
given. In each
lue, the higher

itude	Longitude
7486	6.36099E-05
3283	6.35987E-05
2224	8.69229E-05
2803	0.000143168
26593	0.000145379
48593	0.000185453

Table 3 is the best k search result with the GA method. In the Altitude variable, 10 iterations were carried out and the value of k = 3 was obtained, with an MSE value of 128668.96. At Variable Speed, 10 iterations were also carried out so that the value of k = 9 was obtained with MSE 457.52. In the Heading variable, 70 iterations were carried out resulting in k = 61 with the best MSE value of 752.1429. Meanwhile, for the Lattitude 10 and Longitude variables, 50 iterations were carried out to produce an value of k = 3 and MSE at Lattitude = 9.16E-05 and MSE Longitude 1.68E-05.

Table 3. Table of best k search results with the GA method

Variabel	*Altitude*	*Speed*	*Heading*	*Lattitude*	*Longitude*
iterasi	10	10	70	10	50
k - best	***k:3***	***k:6***	***k:61***	***k:3***	***k:2***
MSE	128668.95	457.52	752.1429	9.16E-05	1.68E-05

3.4 Results of imputation using the KKN-GA method

After completing the experiment with the KNN algorithm and the results have been obtained, then the experiment is continued with optimization with the Genetic Algorithm (GA). In this experiment, the results were quite good when filling in data that experienced missing value (mv). The experiment was carried out on the Sriwijaya Air aircraft type by taking data from 10-20 seconds.

The comparison of the MSE values generated by the two methods using the Euclidein distance measure can be seen in table 4. The MSE value generated by the KNN-GA method in each missing group is entirely better than the MSE value generated by the KNN method. With the KNN algorithm the best MSE results are small. It is the same with the KNN-GA where the best MSE value is obtained with a small one. From the comparison of parameter combinations that give the best results for both methods, it can be seen that there is a fluctuation pattern of the same MSE value increase and the best MSE value is produced by a small MSE value.

For a clearer comparison of the performance of the KNN and the KNN-GA, it can be seen in Figure 7. Comparison of the MSE produced by the KNN method (blue line) with the optimization result (red line) is presented in the figure. For the same missing k values, the GA optimization results all succeeded in giving better values than the experiment with the KNN method.

Table 4 Comparison Table of KNN and KNN-GA

Variable	*K- best*	*KNN*	*k -best*	*KNN-GA*
		MSE		*MSE*
Altitude	2	561219.7664	3	128668,96
Speed	5	392.6019214	6	457.5201.
Heading	2	772.1429469	61	752.1429.
Lattitude	10	0.000126593	3	9.16E-05
Longitude	12	1.68E-05	2	1.68E-05

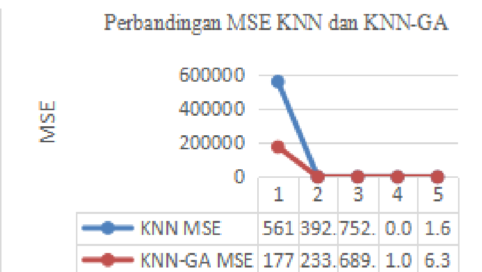


Figure 7. Comparison Graph of MSE KNN and KNN GA.

5.4. Results of Aircraft Data Imputation at Abdurrahman Saleh Airport

From the results of experiments on aircraft data at Abdurahman Saleh Airport, the KNN-GA, GA method has a good performance. This is shown by its ability to minimize MSE. If the imputation value for each observation is considered, the imputation result for each variable in the missing observation is relatively different from the actual value. There is a big difference and some are close to the actual value.

Table 5 presents some of the results of the data imputation of medium aircraft at 1 hour of data collection. It can be seen from the table that the imputation value in some experiments is different from the actual value in the complete data (the part with a question mark "?"). For example, the variable Lattitude in the second experiment. The resulting imputation value is -7.91262 while the actual value of the variable is -7.8157. However, there are imputation results that are relatively close to the actual data value such as the value of the Speed variable in the 10341th experiment. The resulting imputation value was 455, not much different from the actual value of 454. The full imputation results can be seen in the Appendix. Table 5 Results of the Abdurrahman Saleh Airport Aircraft Data Imputation.

Table 5. Comparison Table of actual data and Imputasion

No	Aktual Data					Imputation Results				
	Altitude	*Speed*	*Heading*	*Lattitude*	*Longitude*	Altitude	Speed	Heading	Lattitude	Longitude
......	......	......	......	......	......	......	......	......	......	......
20	34975	?	?	-7.80826	112.96259	34975	458	126	-7.80826	112.96259
21	?	?	?	?	?	35000	458	126	-7.91262	112.96327
22	35000	?	?	?	?	35000	458	126	-7.91262	112.96327
23	?	?	?	?	?	35000	458	126	-7.91262	112.96327
24	?	459	126	?	?	35000	459	126	-7.91262	112.96327
25	34975			-7.8157	112.97274	34975	459	126	-7.91262	112.96327
26	?	459	126	?	?	35000	459	126	-7.91262	112.96327
27	?	?	?	?	?	34975	459	126	-7.81519	112.97203
28	35000	?	?	?	?	35000	459	126	-7.8157	112.97274
29	?	458	126	?	?	34975	459	126	-7.8157	112.97274
30	35000	?	?	-7.81947	112.97791	35000	459	126	-7.81668	112.97274
31	35000	?	?	-7.82073	112.97961	35000	459	126	-7.81668	112.97274
......	......	......	......	......	......	......	......	......	......	......
......	......	......	......	......	......	......	......	......	......	......
10335	31750			-7.94261	113.0975	31750	455	126	-7.94261	113.0975
10336		455	124			31725	455	124	-7.9438	113.09922
10337	31750			-7.94307	113.0982	31750	455	126	-7.94307	113.0982
10338	31725			-7.9438	113.09922	31725	455	126	-7.9438	113.09922
10339		455	124			31700	455	124	-7.9447	113.09922
10340		454	125			31700	454	125	-7.9447	113.09922
10341	31725					31725	455	344	-7.9447	113.09922
10342	31725			-7.94493	113.10084	31725	455	344	-7.94493	113.10084
10343		454	125			31700	454	125	-7.94717	113.10378
10344		454	125			31700	454	125	-7.94717	113.10378

4. CONCLUSION

A. Conclusion

From the research that has been done, several conclusions can be drawn.

1. The KNN method in this study is used to fill in data that has missing value. As with the properties of K-NN, this method will increase in accuracy if the training data or old case patterns that are owned are increasingly varied. In addition, this method has strong consistency, by looking for cases by calculating the closeness between new cases and old cases based on their k values. In accordance with the experimental results of k values randomly at k = 2, 5, 2, 10, and 2. For the better In the finding process for k on the KNN, optimization was carried out using the GA method.
2. Optimization using the GA method is superior, this is indicated by the smaller MSE value compared to the KNN. GA is better from the two methods.

3. Through GA Optimization, get variable information from the method through MSE so that the imputation of the missing value data for aircraft is better.

B Advice

Several things that can be developed for further research in the same scope include:

1. Assessing the application of the KNN-GA to different data structures through data collection and other types of data
2. Do a combination of other methods from the data cases that experience missing value.
3. Use a distance measure other than Euclidean Distance.
4. Applying optimization methods other than the GA method and making comparisons with other imputation methods

6. REFERENCES

[1] ICAO. Guidance Material on Issues to be Considered in ATC Multi-Sensor Fusion Processing Including the Integration of ADS-B Data. 2008.

[2] Eko Marpanji, Kadarisman, Gamantyo H, Aplikasi platform komputasi software defined radio (SDR) untuk digital spektrum analizer .ITS. ISSN 2337-3539.2010.

[3] Akshay N et, Shruthi R, Sushmitha K N, Vanitha R, Dr. Rekha K R .*"Live detection with Mode-S Trasponder Using RTL-SDR"*.2017. Karnataka, India

[4] Little, Roderick, J. A & Rubin, Donal B. "Statistical Analysis With Missing Data". California. 1987.

[5] C. F. Suharno, M. A. Fauzi, and R. S. Perdana,"Klasifikasi Teks Bahasa Indonesia Pada Dokumen Pengaduan Sambat Online Menggunakan Metode K-Nearest Neighbors (K-NN) dan Chi-Square," *J. Pengemb. Teknol. Inf. Dan Ilmu Komput. Vol 1 No 10 2017*, Jul. 2017.

[6] Aprilio Adrianus Yoza, Ginardi Hari & Purwananto Yudhi. . Manajemen Kualitas Aliran Data A *utomatic Dependent Surveillance-Broadcast* (ADS-B) Banyak Titik Dengan Pohon Keputusan, Jurnal Teknik Pomits, Vol. 4, No. 2. 2014

[7] Hiroshi De Silva and A. Shehan Perera, "Missing Data Imputation using Evolutionary K-Nearest Neighbor Algorithm for Gene Expression Data", Proceedings of 6 th International Conference on Advances in ICT for Emerging Regions, 2017.

[8] Pang Nin Tan, Michael Steinbach, & Vipin Kumar. . *"Introduction to Data Mining 5th Ed. Boston: Pearson Education Inc.*" 2006

[9] Steffi Pauli & fazat Nur Azizah. "Imputation of missing value using dynamic Bayesian network for multivariate time series data. IEEE. 2017

.

International Journal of Computer Applications Technology and Research
Volume 9–Issue 12, 332-335, 2020, ISSN:-2319–8656

Design of Remote Network Audio and Video Communication System

Chen Lin
School of Communication Engineering
Chengdu University of Information Technology
Chengdu, China

Tianbao Wang
School of Communication Engineering
Chengdu University of Information Technology
Chengdu, China

Chenxi Yang
School of Communication Engineering
Chengdu University of Information Technology
Chengdu, China

Abstract: With the continuous development of information technology, real-time audio and video communication is becoming more and more important in people's daily life. At present, there are a lot of software with audio and video communication function in the market, such as Tencent QQ software, wechat, nailing, etc. these software are developed with C / S architecture, and can not achieve cross platform and cross terminal. Users need to download the software to carry out audio and video communication. In view of this, this paper uses webrtc technology based on ice framework to realize the design of remote audio and video communication system, completes the internal network penetration, establishes the communication process between clients, designs room server and signaling server for conversation, and completes the design of remote network audio and video communication system.

Keywords:Intranet penetrating, webrtc, remote audio and video

1. INTRODUCTION

At present, most of the audio and video real-time communication software in the market is developed by independent application program based on C / S architecture, which requires users to install software on the device, and the real-time communication of audio and video between software of different platforms is not allowed. Audio and video data coding and decoding methods, echo cancellation, communication protocol establishment and other issues are also the difficulties of traditional audio and video communication software. To solve this problem, this paper introduces and adopts webrtc real-time communication technology based on browser launched by Google, which provides audio and video acquisition, coding and decoding, audio / video synchronization, flow control (RTP / RTCP), network transmission and other functions [1]. It also supports cross platform communication based on IP network, which can realize real-time audio and video communication between communication terminals without platform control. The ice protocol stack, which integrates stun and turn advantages, breaks through the limitation of NAT / firewall and realizes intranet penetration. Using webrtc technology and Intranet penetration ice, users can realize audio and video communication through browser in heterogeneous network environment, reduce the load of client equipment, and meet the needs of cross platform and long-distance instant communication.

2. INTRODUCTION TO SYSTEM TECHNOLOGY

2.1 Webrtc Technology

Webrtc technology is a real-time multimedia communication technology opened by Google in 2011. Developers can complete the development of audio and video communication by calling the API interface provided by browser itself, and can realize real-time audio and video communication function based on browser without any third-party plug-in. As shown in the webrtc architecture diagram in Figure 1, the overall architecture of webrtc is divided into three layers. The first layer of your web app is a web application developed by developers through a browser that provides webrtc support. The second layer mainly provides JavaScript API for developers to develop independently, For example: getusermedia API provides webrtc with the function of acquiring video and audio data from the camera and microphone of the device; rtcpeerconnection API transfers stream data between browsers; rtcdata channel API establishes data channel for webrtc. The third layer is the core layer of webrtc technology, including voice engine, video engine and transport engine.[2-3]The audio engine mainly encodes the audio data collected by the microphone for audio coding, noise suppression, echo cancellation, etc., and then transmits it to other clients through the network; the video engine encodes and processes the video data collected by the camera, enhances the color of the video, and improves the user experience, and then transmits it to other clients through the network If the intranet penetration is realized, the audio and video data are encrypted and transmitted to different LAN clients.[4]

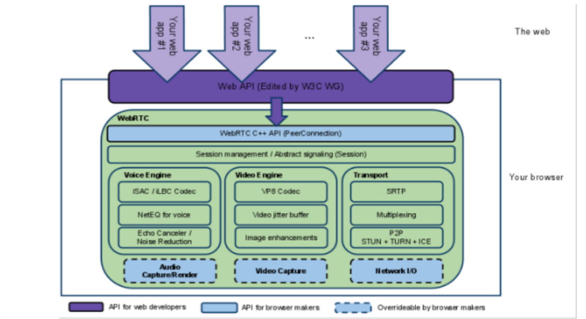


Figure.1 Webrtc architecture

2.2 ICE

At present, many schemes have been applied to solve NAT problems, such as stun (session transverse utilities for NAT), turn (full name of turn is transverse using relays around NAT, mainly adding relay function), algs, symmetric RTP, RSIP, etc. However, when these technologies are applied to different network topologies, they have obvious advantages and disadvantages, so that different solutions can only be applied according to different access methods. Therefore, the problems of all NAT and efficiency can not be well solved, and many complex and fragile factors will be introduced into the system.

In this paper, ice interactive connectivity is used.[5]It is a solution of traversing NAT / firewall proposed by IETF. It is a NAT transmission protocol for offer / answer model. A medium based on UDP is established on this model. The SDP of offer and answer contains a variety of IP addresses and ports. The IP addresses in the local SDP and the remote SDP are paired, and then the connectivity test is carried out through P2P connectivity check. If the test is passed, then the test is conducted And indicates that the transmission address can establish a connection, It is suitable for use in heterogeneous network environment. As shown in Figure. 2 Ice framework, ice effectively integrates the advantages of stun and turn, and makes it work under the most suitable situation by comprehensively using stun and turn, so as to make up for the defects of using one of them alone, and provides the optimal solution for the system to penetrate NAT / firewall in various situations.

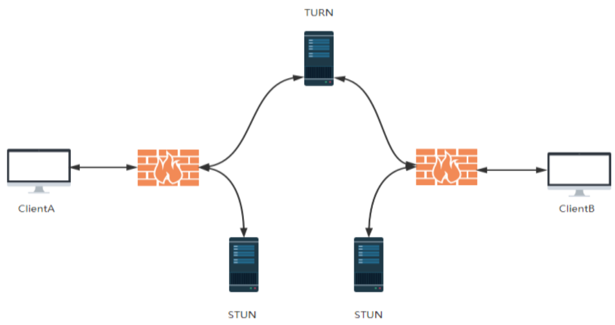


Figure.2 Ice framework

3. SYSTEM ARCHITECTURE

The system architecture of remote network audio and video communication system is shown in Figure 3. The system is composed of web communication client, stun and turn server based on ice framework, and signal server. Because the intranet IP can't communicate directly in the public network, it is necessary to borrow the public IP to realize the communication between clients not in the same LAN. Therefore, local and remote, the web communication client of remote network audio and video communication system, first obtains its communication candidate address by ice protocol including stun and turn protocol, that is, the browser sends test data package to stun server to obtain its public IP address (NAT address). In some cases, if stun server can not achieve traversal, it must In the public network, turn server is used to implement relay transfer, and then the candidate address obtained is communicated to the other party through the signaling server exchange. The signaling server is also responsible for the interaction of other signaling information, such as SDP message. After the signaling information exchange is completed, the transmission of media stream data can be realized[6].In the process of media stream data transmission, the web communication client using webrtc technology will construct audio and video media controls in the web page, bind the audio and video data collected by local and remote cameras and microphones to the media controls for output, so as to realize remote network audio and video communication.[7]

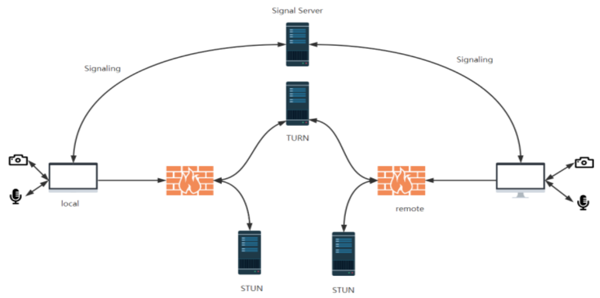


Figure.3 System architecture

4. TRANSMISSION MODE

The transmission mode of remote network audio and video communication system adopts ice framework, stun and turn are comprehensively used, so that client A and client B in heterogeneous network environment can realize communication under Wan, and meet the network transmission requirements of remote network audio and video communication system. As shown in Figure 4, symmetric NAT / firewall network topology based on iceframework.

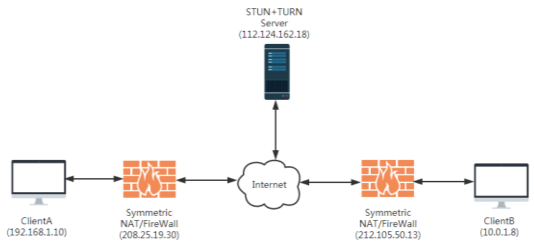


Figure.4 Symmetric NAT / firewall network topology

client A and client B realize the exchange process through NAT / firewall through ice framework[9]as follows:

(1) client A collects IP address and finds out the traffic address that can be received from stun server and turn server;

(2) client A sends an address list to stun server, and then sends start information to client B according to the sorted address list. The purpose is to realize the communication between client A and client B;

(3) Client B sends a stun request to each address in the startup message;

(4) client A sends the reply message of the first stun request received to client B;

(5) After receiving stun's reply, client B finds out the addresses that can realize communication between client A and client B;

(6)The highest address in the list is used to further complete the communication between client B and client A.

5. AUDIO AND VIDEO COMMUNICATION DESIGN

5.1 Communication process establishment

The design requirements of remote network audio and video communication system: 1. Be able to collect real-time audio and video data from local client A and remote client B; 2. The collected images and audio have certain clarity, which will not affect the identification between communication personnel; 3. Send audio and video packets in the same step; 4. Video window can receive the transmitted video stream and display real-time pictures; 5. Local area client A and remote client B enter the same room server and can exit the room server at any time. According to the design requirements, webrtc technology is used to establish the communication process between local and remote personnel, the room server for audio and video conversation between local client A and remote client B, and the signaling server for signaling exchange between local client A and remote client B.

The communication flow between local client A and remote client B is established as shown in Figure 5.

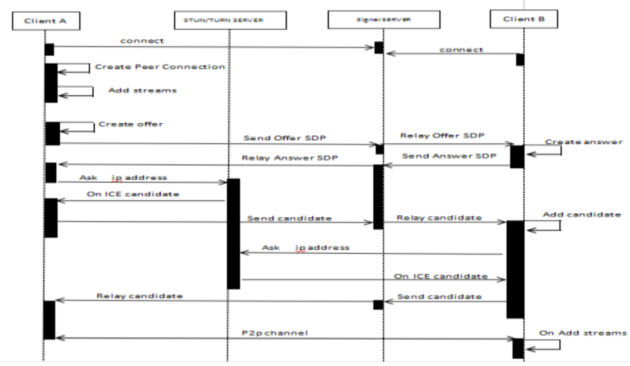


Figure.5 Communication flow chart

(1) Local client A first creates a peerconnection object, then opens the local audio and video device, encapsulates the audio and video data as mediastream and adds it to peerconnection.

(2) Local client A calls the createoffer method of peerconnection to create an SDP object for offer. The SDP object stores the relevant parameters of the current audio and video. Local client A saves the SDP object through the setlocaldescription method of peerconnection, and sends it to remote client B through signal server.

(3) Remote client B receives an offer sent by local client A The SDP object is saved by the setremotedescription method of peerconnection, and a response SDP object is created by calling the createanswer method of peerconnection. The SDP object is saved by setlocaldescription method of peerconnection and sent to local client A through signal server.

(4) Local client A receives the response SDP object sent by remote client B and saves it through the setremotedescription method of peerconnection.

(5) In the offer / answer process of SDP information, local client A and remote client B have created corresponding audio channels and video channels according to SDP information, and started the collection of candidate data. Candidate data can be simply interpreted into IP address information of communication terminal (local IP address, public IP address, address assigned by relay server).

(6) When local client A collects candidate information, peerconnection will send notification to local client A through onicecandidate interface. Local client A will send the received candidate information to remote client B through signal server, and remote client B will save it through addicecandidate method of peerconnection. Do the same for remote client B and local client A again.

(7) In this way, the local client A and the remote client B have established a P2P channel for audio and video transmission. After receiving the audio and video stream from local client A, remote client B will return a mediastream object identifying the audio and video stream of local client A through the onaddstream callback interface of peerconnection, which can be rendered in remote client B. The same operation also adapts to the transmission of audio and video stream from remote client B to local client A.

5.2 Building room server

The room server of audio and video conversation between remote client B and local client a adopts loose coupling mode [10], which makes local client a and remote client B directly enter the room for interactive audio and video communication. The loosely coupled room server is connected with the server of signaling server through websocket through WS module in nodejs. The algorithm flow of the whole staff joining the room is as follows:

```
(1)this.rooms = {};// initialize room
(2) socket.send(JSON.stringify({
(3)      "eventName": "join",
(4)      "data": {
(5)         "room":"roomname"
(6)      }}), //send join signaling containing the name of the room and join the room
(7) current.send(JSON.stringify({
(8)      "eventName": "new_people",
(9)      "data": {
(10)        "socketId":socket.id
(11)     }}),//send the information of new employees to all the people in the current room
(12) socket.send(JSON.stringify({
(13)     "eventName": "others",
(14)     "data": {
(15)        "connections": ids,
(16)        "you": socket.id
(17)     }}),//send information about other people in the room to the new person for establishing a connection with each other
(18) room[i].send(JSON.stringify({
(19)     "eventName":"removeroom",
(20)     "data": {
(21)        "socketId"socket.id
```

(22) }}),//send exit room signaling to inform other personnel

(23) that.removeSocket(socket);//personnel exit the room

5.3 Establishment of signaling server

Before the connection between remote client B and local client a is established, data cannot be transmitted between the two ends. Therefore, we need to realize the signaling data exchange between the two ends through the signaling server, such as the exchange of media description information SDP [11], the exchange of connection address, the exchange of connection message to control the opening or closing of communication, and the message exchange informing each other when an error occurs Establish a point-to-point connection at both ends. Nodejs is used to design the signaling server this time. At present, nodejs is a very mature web server. It uses V8 engine on the server side to parse the written JavaScript application. After the application is parsed by V8, the C / C + + API at the bottom of nodejs is called to start the server. First load the client of the server, that is, the room server where remote client B interacts with local client a for audio and video interaction. The server listens to port 3000 to process different messages. For example, when the server receives a message message, it will broadcast directly, and all clients connected to the server will receive the broadcast message; the server receives "create or" If there is no one in the room, it will send a "created" message; if there is a person in the room, it will send a "join" message; if you exit the room, it will send a "removerroom" message, and then through the node server.js The command starts the signaling server.

6. CONCLUDING REMARKS

This audio and video part uses webrtc technology to establish the local and remote communication process, completes the design of the local and remote room server and signaling server used to exchange signaling information, and completes the design of remote audio and video communication system under heterogeneous network environment combined with ice framework, which is conducive to users' real-time long-distance audio and video communication on cross platform web pages, and reduces the cost of communication software Load, reduce the hardware load. On the basis of this design, more functions can be developed and designed, which is conducive to more research on remote network audio and video system in the future.

7. REFERENCES

[1] Li Yuxuan. Design and implementation of instant messaging video system based on webrtc [D]. 2016

[2] Hu Xun, Zhou Yuanping. Wireless video transmission onAndroid mobile terminal based on webrtc [J]. Microcomputer and applications, 2015 (19): 83-85 DOI:10.3969/j.issn.1674-7720.2015.

[3] Lin Hong, Wang song, Yang Xin, et al. Application andplatform technology development and design based on webrtc technology [J] . Telecommunication science, 2013, 29 (9): 20-25

[4] Cao Shenhui. Research and implementation of NAT traversal technology [D]. Nanjing: Nanjing University of Posts and telecommunications, 2013

[5] Du Derong, Liu Jianming, Li Hongzhou, et al. Research on SIP traversing NAT in mobile Internet environment [J]. Microelectronics and computer, 2013,30 (7): 11 - 113

[6] Zhu Guang, Zhang Yunhua, Lu Juan. Research on the scheme of VoIP traversing NAT based on ice 00. Computer applications and software, 2011,28 (10): 223-224 NAT and firewall traversal with STUN/TURN/ICE[Z/OL]. Perreault S. http://viagenie.qc.ca/publications/2008-09-24-astricon-stun-turn-ice.pdf .

[7] Peng Yongchao, Fei Jinghao. Interactive live video system based on webrtc and PWA [J] . Computer programming skills and maintenance, 2017 (2): 74-76

[8] NAT and firewall traversal with STUN/TURN/ICE[Z/OL]. Perreault S. http://viagenie.qc.ca/publications/2008-09-24-astricon-stun-turn-ice.pdf .

[9] media server technology for convergent WWW/mobile real— timemultimedia communications supporting WebRTC［C］/ / World of Wireless，Mobile & Multimedia Networks． IEEE，2013．

[10] Liang Yan. Analysis of webrtc technology based on HTML5 ［ J ］. Information and communication technology, 2014 (2): 52- 56