ISSN 1816-6075 (Print), 1818-0523 (Online)
Journal of System and Management Sciences
Vol. 13 (2023) No. 1, pp. 344-361
DOI:10.33168/JSMS.2023.0119

Root Cause Analysis for IT Incident using Artificial Neural Network (ANN)

Revan Hadi, Abba Suganda Girsang

Computer Science Department, BINUS Graduate Program – Master of Computer Science, Bina Nusantara University, Jakarta, 11480, Indonesia
revan.hadi@binus.ac.id; agirsang@binus.edu

Abstract. Root Cause Analysis (RCA) is a problem-solving method to identify the root cause of a fault or problem. In the context of IT operations, RCA is one of the methods needed to track the root cause of an IT incident. The process of RCA for IT Incident involves gathering as much as possible information about application logs, database logs, network logs, server logs, etc. and predicting main root cause of IT incident. To solve that problem, Artificial Neural Network algorithm will be adopted to predict the root cause of incident. ANN algorithm is used to classify the dataset and then get the training model that can be used for RCA. 1000 dummy data from the past documented incidents are used as a dataset. To evaluate how good the results are, categorical cross entropy and k-fold cross are calculated to compare root cause generated by system and reference root cause. This study also set up comparisons to other methods such as Logistic-Regression, KNN, Decision Tree, Random Forest, AdaBoost and XGBoost. Evaluation results for RCA using ANN, k-fold cross evaluation metric is 81,90% with an accuracy of 85% compared to the other methods, not the best but more stable with an 87% precision score.

Keywords: Artificial Neural Network, IT Incident, IT Operation, MTTR, Root Cause Analysis, System Logs

1. Introduction

Information and Technology (IT) incidents such as service interruption, performance degradation, outages, and system anomalies are inevitable in operating IT services within enterprise companies. IT incidents have been a topic of research and innovation for many years. Despite extensive research in this area, IT incident rates are tremendous. Multiple industry surveys show that there are significant losses due to IT incidents such as network downtime, database corruption, application bugs, hardware malfunction, and so on. There are two approaches in handling IT incidents: reactive approach and proactive approach (Tan and Gu, 2010). Corrective action is taken after the failure occurs in the reactive approach. In this approach, quick action taken to find the cause of the error and promptly handle the failures can result in downtime. The reactive approach includes anomaly detection and Root Cause Analysis. Whereas, in a proactive approach, proactive actions are taken before failure occurs to avoid it; thus, it prevents downtimes and associated losses. The proactive approach works is based on forecasting system failures so that corrective action can be taken to avoid the failure. This approach offers better reliability by preventing downtime. IT operation monitoring has become a challenging task due to increased complexities in IT infrastructure (i.e., network, server, database, etc.) and its utilization. Any failure for a small amount of time leads to significant losses to an organization. Thus, it is foremost essential to avoid such failure conditions.

In recent years, Artificial Intelligence for IT Operations (AIOps) has played a critical role in IT incident detection and prediction by leveraging historical system logs and machine learning algorithms. AIOps is a discipline that combines Big Data and Machine Learning to automate IT operational processes, including event correlation, anomaly detection, and root cause analysis. AIOps uses machine learning and data science to provide IT operations teams with a real-time understanding of any issues that could impact the availability and performance of IT services. Root Cause Analysis (RCA) is one of the use cases that can be explored with AIOps. RCA is the approach for defining, understanding, and resolving the fault in the system (Bhanage et al. 2021). RCA is necessary to find the underlying cause of the problem in order to identify specific, focused, and appropriate solutions to minimize MTTR. RCA is an iterative and interrogative process used to explore the cause-and-effect relationships of the underlying problem. There are corresponding symptoms when a problem occurs. These symptoms are then mapped to the root cause through a series of questions and exploratory tests. This is similar to how doctors diagnose symptoms, such as fever, until the root cause, for example a viral infection, is identified.

In the context of IT Operations, the operation team receives a number of service incidents from users on a daily basis. Incidents are usually followed by the symptoms observed by the user. The operational engineer team can further analyze

the problem to identify additional symptoms before proceeding with the analysis process to narrow down the symptoms to the root cause. For example, a user may encounter a symptom such as, "We were unable to save your changes." "Please contact the administrator". The root cause could be a lack of disk space in the database. The time required to narrow down the root cause is an important factor in reducing MTTR. This requires expert assistance from software developers or vendors supplying the product. AIOps technology can help by identifying symptoms, predicting root causes, and assisting the IT operations team in quickly resolving root causes.

Root Cause Analysis (RCA) for IT incidents will involve data from various domains. For example, when there is a delay in access to a database application, the domain that must be investigated is not limited to the application area (system) but also infrastructure aspects such as servers, disk storage, network, and security. System logs provide the information about each component's status and record the system operational changes such as starting or stopping services, software configuration modifications, software execution errors and hardware faults, and so on. Therefore, telemetry data generated from various application systems and infrastructure devices such as log files, server metrics, SNMP, Syslog, usage stats events, IT service tickets, and Known Error Database (KEDB) can be optimized as strategic assets in predicting the root cause of an IT incident. Saha and Hoi (2022) focus on a rich data-source of past documented incident investigation reports, generally termed as Problem Review Board (PRB) data, which constitute a core component in all major IT incident management pipelines. The manual RCA investigation consists of 5 steps: **1) Incident Detection,** which typically relies on Analysis of Key Performance Indices. 2**) Symptom Detection,** which detect the primary effect of the service disruption on performance factor. **3) Investigation,** which requires intensive communication between teams of Site Reliability Engineer in order to understand the nature of the incident and decide a target team who can undertake the RCA Investigation. **4) Immediate Resolution,** based on the conclusion of investigation, action taken is workaround to mitigate problem temporarily. **5) Root Cause Investigation,** the RCA target team can finally find the true root cause.

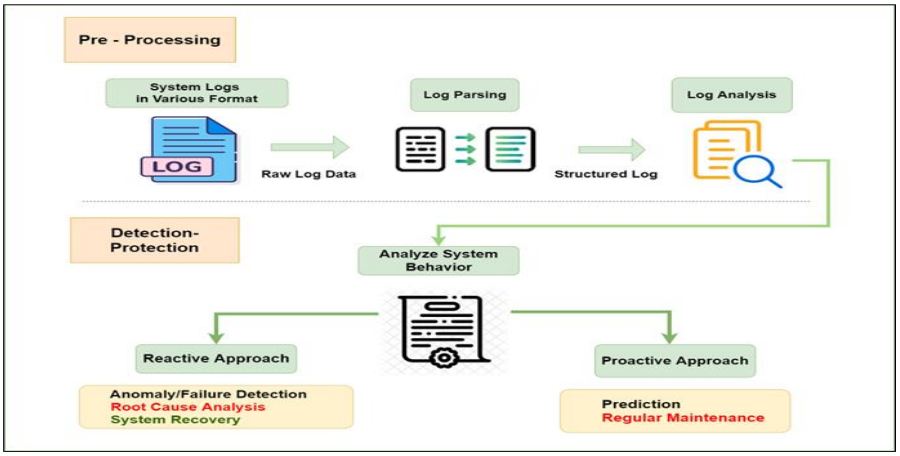

Fig. 1: Flow Diagram of System Logs Processing

Figure 1 shows the flow of system logs processing. The first step is to collect system logs from various systems with various formats. The second step is to process, convert, and standardize all collected logs from unstructured to structured form. In addition, it also performs data abstraction by taking the similarity and relevance of information in log data into account. The third step is to analyze the log to make it more readable and understandable before continuing to analyze system behavior. Finally, there are two approaches: reactive or proactive. In the reactive approach, it focuses on anomaly detection, root cause analysis and system recovery. Meanwhile, in a proactive approach, it focuses on preventing future incident by predicting it.

The field of root-cause analysis (RCA) has been growing for the last couple of years and will continue to attract research attention in the future. Most of them use deep learning to approach the RCA of the system. One research by Kai Qin et al. (2022), implemented binary extreme gradient boosting (Bi-Xgboost) to discover RCA of industrial faults, Bi-Xgboost is proposed for variable contribution analysis. Jiang Wenhao et al. (2022) proposed an alarm propagation graph neural network for fault detection and RCA. This journal identifies the root alarm based on the Bayesian network. Neural network is used to extract features and learn the mapping from alarm propagation graph to the true fault. Later in 2016, Ahmed Jawwad et al. (2016) implemented a random forest method for predicting SLA violations of cluster-based service running in data centers. This approach follows the in-network analytics philosophy where the data analysis is performed at the data source. Most of the research focus on the analysis of the RCA and predicts the output that is new and not in the dataset before. The accuracy of the research is quite high but unstable.

In this research, our work uses ANN to analyze RCA and predict the result. The accuracy of our method is quite high, 0.85 with more stable precision score of 0.87. This accuracy can be improved by using an auto healing algorithm to take out the root cause and heal the system. For accuracy it can be improved with combination

of ANN with another algorithm using the ensemble method. It means there is still room for a researcher to improve this RCA research.

Therefore, the objective of this paper was to propose Root Cause Analysis for IT incident that used an Artificial Neural Network (ANN) to identify the problem domain and resolve the incident as soon as possible. The proposed solution has several benefits. Firstly, it will reduce Mean Time to Resolution (MTTR) IT incident in organizations. Secondly, it will improve IT operational quality and customer experience.

The rest of this paper is structured as follows. Section 2 explains the related works. Section 3, explain the proposed method. Section 4 is about the results and discussion. Finally, the conclusions are presented in Section 5.

2. Related Work

Root Cause Analysis has become a popular research topic in the past few years. Research on RCA has been widely carried out in several industrial areas, including RCA in manufacturing, power plants, transportation, and IT by utilizing Big Data and Machine Learning technology.

In the research conducted by Sarkar et al. (2020), a Root Cause Analysis (RCA) was conducted in the Indian steel industry regarding the high incidence of work accidents. Researchers clustered several incident scenarios to determine the root cause of certain workplace accidents to prevent similar incidents from happening in the future. This research used 2 Machine Learning algorithms, namely Random Forest (RF) and Support Vector Machine (SVM).

In this research, incident data was collected based on categories, numeric values and text. Despite the fact that many studies have been conducted using categorical and numerical data, the text has not been optimized. Text document clustering can be used to extract potential root causes of an accident. Thus, the keywords generated from this clustering will be useful for revealing the root cause and location of work accidents as a leading indicator for the company. The methodology used in this research consists of Data Preprocessing and Root Cause Analysis by Text-Document Clustering.

Meanwhile, Bhanage et al. (2021) explain that in IT infrastructure monitoring, system logs are a must to detect failures, Root Cause Analysis, and troubleshooting. The researcher raised the topic of Systematic Literature Review (SLR) which focuses on detailed analysis based on various qualitative datasets, technical approaches, and automation tools.

This SLR ensures that Machine Learning and Deep Learning with a classification approach will improve performance in comparison to traditional rule-based and method-based approaches. Various logs (RAS logs, health logs, event

logs, event logs, activity logs, transactional and operational logs) are parsed with pattern mining, clustering, and natural language processing (NLP), then modeled with machine learning models (SVM, Naïve Bayes, Random Forest) and deep learning models (RNN, CNN, LSTM, Bi-LSTM) to detect and predict anomalies or failures on various types of IT Infrastructure.

RCA is an approach to define, understand, and resolve errors in the system. RCA is needed to find the root cause of a problem so that appropriate solutions are determined and can prevent future failures. Lu et al. (2020) designed a model to find the root cause of delay in the Spark System by using weighted factors to determine the probability of the root cause. CPU, Memory, Network and Disk are the four components that are included in finding the root cause of the abnormality as shown in Figure 3.

Weng et al. (2020) developed a solution to assist cloud administrators in localizing the root causes of anomalies and do so in both the application and infrastructure areas. Then a graph-based framework was developed by Brandon et al. (2020) to find the root cause of service-oriented and micro-service architectures.

Molan and Molan (2020) conducted a Root Cause Analysis (RCA) of accidents in sea, air, rail, and highway transportation traffic. Molan and Molan (2020) explained that RCA is a method developed to identify the main causes of transportation accidents and prevent recurrence in the future. An effective RCA requires all important accident-related data. One approach to collecting data is the Flanagan technique which consists of 5W (What, Where, When, Who, Why). The entire data collection is then analyzed by graph theory. The graph connects 5 separate things: (1) critical human behavior patterns; (2) mechanism of human behavior; (3) root cause of inadequate behavioral patterns; (4) psychological basis of behavioral patterns; (5) preventive accident preventions.

Thorstenson et al. (2021) conducted a Root Cause Analysis (RCA) for quality problem solving in the manufacturing industry to improve product quality and reduce risk. In this study, an RCA framework based on Big Data is introduced which includes 3 models, namely Problem Identification (PI), Root Cause Identification (RCI) and Permanent Corrective Action (PCA).

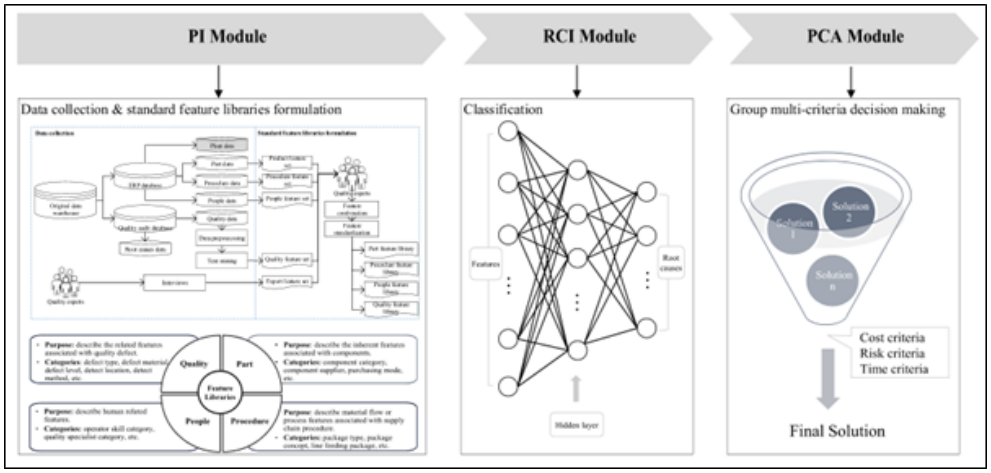

Fig. 2: Framework of the Big-Data driven RCA System

As described in the PI module of Figure 2, relevant data is collected and features are extracted from the data to formulate standard features so that quality problems can be explained in a systematic way. The PI module extracts 4 main data from the ERP database, including plant data, part data, procedure data, people data (4P).

In this RCI module, supervised ML implementation is carried out, namely the classification of K-Nearest Neighbour (KNN) and Neural Network (NN) to predict the root cause. The dataset used is divided into 3 subsets to implement the classification model.

- Training set: used to train the classification model
- Validation set: a small part of the training set used to estimate the performance of the classification model
- Test set: used to evaluate the performance of the final model

Prior to training on the model, Boehmke et al. (2019) suggested pre-processing the data, which includes activities such as (i) filtering zero or near zero variance data; (ii) performing imputation if necessary; (iii) normalizing skewness; (iv) standardizing numerical features (centre and scale); (v) performing dimensional reduction on numerical features; (vi) using dummy encoding for category features.

The K-Nearest Neighbours Model (KNN) algorithm classifies new observations by identifying the classes of K-Nearest Neighbours. Thorstenson et al. (2021) used KD-trees classifier and Fast KNN classifier. KD-trees are very effective in providing space-partitioning data structures, while Fast KNN is very competitive in computing for very large datasets.

In the Neural Network Model algorithm, there are several model developments including Feed Forward Neural Network (FFNN), Convolutional Neural Network

(CNN), and Recurrent Neural Network (RNN). Specifically for this study, Thorstenson et al. (2021) used multi-layer. Perception (MLP) is a class of FFNN and a model that is commonly used in many applications.

The results of the research show that by applying Machine Learning to the RCI module, it can efficiently support RCA both on individual and multi-quality problems. Empirical data shows the root cause of 11,000 quality problems in a predictable 1 second.

Further development of this research can be applied by implementing the Support Vector Machine (SVM) and Random Forest (RF) algorithms on the RCI module. In addition, this RCA application can be carried out in a more complex manufacturing industry and produces a variety of products ranging from luxury, mid-to-high-end to low-end categories.

In the research conducted by Velasquez and Lara (2021), a Root Cause Analysis (RCA) was used to diagnose and classify failures in power transformers in power plants to improve reliability and quality with minimum interruptions. RCA is one way to initiate an understanding of a problem by using causal theory and constraints. This research uses Artificial Neural Network (ANN).

The ANN algorithm requires very large training data. The Support Vector Machine (SVM) approach requires the selection of the right parameters and is very sensitive. This study uses a genetic classification algorithm that progressively increases its accuracy through learning. The methodology used in this RCA is as follows:

- Choose a case: Mapping problems, collecting data and evidence.
- Case analysis: Identification, classification, and prioritization of positive contributions.
- Root Cause identification
- Generate solutions
- Implement solutions and disseminate result

3. Proposed Method

The proposed method of ANN for Root Cause Analysis consists of some steps, as shown in Figure 3. The number of data used were 1000 dummy data from historical IT incidents. This Root Cause Analysis was equipped with Feature Encoding using Encoding label.

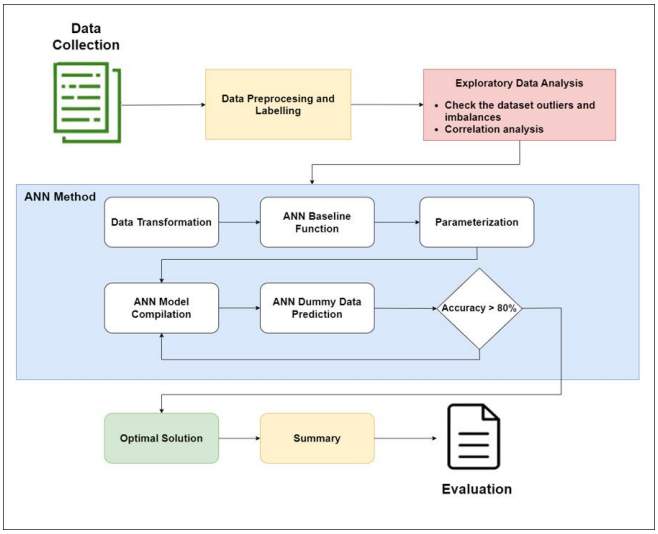

Fig. 3: Proposed Method

3.1. Featuring Data

Dataset of incident was retrieved from historical IT incident that was saved in Known Error Database (KEDB). Dataset in this research was dummy data which contains 10 Featuring Data, has 3 output targets and 1000 rows of data. This dataset has 10 inputs that represent the type of error category that was calculated to determine the Root Cause Analysis. Featuring Data and Targets are shown in Table 1.

Table 1: Sample of Featuring Data and Target

	Feature										Target
Id	CPU	Disk	Delay	Error_1000	Error_1001	Error_1002	Error_1003	Error_1004	Error_1005	Error_1006	Root_Cause
1	0	0	0	0	1	0	1	0	1	0	CPU_ISSUE
2	0	0	0	0	0	0	1	0	1	0	CPU_ISSUE
3	0	1	1	0	0	1	1	0	0	0	CPU_ISSUE
4	0	1	0	1	1	0	1	0	0	0	CPU_ISSUE
5	1	1	0	1	0	1	0	0	1	1	Network_Issue

3.2. Preprocessing and Labelling

Featuring Data has 10 inputs and is in the form of data 1 or 0 which determines what symptom occurs and the combination of each symptom determines the target result or output. Target label consists of CPU_ISSUE, NETWORK_ISSUE, and DATABASE_ISSUE because the machine learning is only able to work with numeric values, therefore the three outputs must be changed to become a label. The changes are commonly known as feature encoding that can be seen in Table 2.

Table 2: Feature Encoding on Target

Root_Cause	Root_Cause_Enc
NETWORK_ISSUE	2
DATABASE_ISSUE	1
CPU_ISSUE	0

3.3. Exploratory Data Analysis

Checking the dataset to detect an anomaly such as outliers and imbalances is necessary, but after comparing the dataset to three unique targets, it can be concluded that the dataset balance is in accordance with the following results:

value_unique_target: ['CPU_ISSUE' 'NETWORK_ISSUE' 'DATABASE_ISSUE']

number_of_value_unique_MEMORY: 323

number_of_value_unique_NETWORK_DELAY: 337

number_of_value_unique_DATABASE_ISSUE: 340

1000 datasets were compared with the number of targets show that the target is in a state of balance with a comparison percentage of 32.3% : 33.7% : 34%.

Correlation analysis is the most frequently applied technique in machine learning and data mining. By looking at the correlation of the heatmap with a certain set of features, it shows the correlation between the feature and the target. Figure 4 below shows correlation of the heatmap between feature and target.

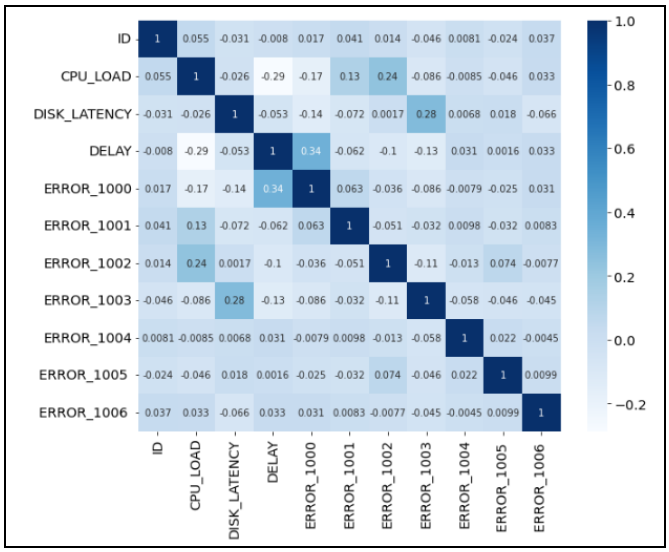

Fig. 4: Correlation Heatmap

3.4. Root Cause Analysis using Artificial Neural Network

The ANN model was conducted to determine the root cause of IT incident. Artificial Neural Networks are a computational system inspired by the structure and

learning characteristics of biological neuron cells and very strong classifiers in pattern recognition.

The next step was creating a baseline function after analyzing the dataset with a correlation heatmap. The models contained two layers. The first layer had 20 neurons and used the ReLu activation function while the second layer used 3 neurons and the sigmoid activation function. The model had a batch size of 10 with 200 epochs. The network topology of these two layers neural network can be summarized as follows: 10 inputs → [20 hidden nodes] → 3 outputs.

The stages of Artificial Neural Network process are given below:

- Begin Data Processing

  To begin the dataset, load to dataframe using pandas was needed. The output variable contained three different string values. When modeling multi-class classification problems using neural networks, output attribute needs to be transformed from a vector that contains values for each class value to a matrix with a Boolean type using one hot encoded.

- Create ANN Baseline Function

  Create a function to generate a baseline neural network for classification problem. The function creates a simple, fully connected network with one hidden layer that contains eight neurons. The hidden layer uses a rectifier activation function. The output layer must create three output values (one for each class) because the use of a one-hot encoding for the dataset.

- Set Parameter

  The model has a batch size of 10 with 200 epochs to train the dataset.

- Compile ANN Model

  All function baselines were called after setting the parameter then the result of accuracy was checked from the dataset.

- Perform Predict with Dummy Data

  Create set list of dummy dataframe to predict. Random data was used in this case.

- End Root Cause Output

  An estimator variable to predict used to validate the prediction result. Using the previous dummy dataset, the value result was 1 (‘Database Issue’) from target data that has already been transformed to one hot encoding.

3.5. Evaluation Method

The proposed method in this study was evaluated by two standard evaluation metrics, which were categorical cross entropy and k-fold cross. Categorical Cross entropy metric is a standard evaluation metric to test how well the output a probability over the classes for each input. Categorical Cross entropy metric is used

for multi-class classification and designed to quantify the difference between two discrete probability distributions. The parameters analyzed were softmax and cross entropy loss as in Eq (1) and Eq (2).

$$\hat{y}_i = \frac{e^{s_i}}{\sum_j^C e^{s_j}} \tag{1}$$

$$CE = -\sum_i^c t_i log\, \hat{y}_i \tag{2}$$

Where $\hat{y}_i$ is the *i*-th scalar value in the model output, *e* is Euler number, *CE* is cross entropy losses, *s* is class that was used in the method (matrices form), t_i is the probability that event *i*-th occured and *i/c* is the number of scalar values in the model output. This loss function is a reasonably good measure of how distinguishable two discrete probability distributions are from each other. $\hat{y}_i$ is the probability that event I occured and the sum of all $\hat{y}_i$ was 1. The output of model needs to be positive so that the logarithm of every output value $\hat{y}_i$ exists. The softmax activation rescales the model output so that it has the right properties.

Another evaluation metric is k-fold cross validation, which creates a process where every sample in the data is included in the set at some test. k represents a number of folds, usually in ranges of 3 to 10. The data was split into k equal folds and their deviations for each running were analyzed.

4. Result and Discussions

4.1. Dataset and Setup

The data used in the research was a dataset of IT incidents that occurred and were recorded in the IT incident portal, then engineered, collected, separated and labeled. The data used were dummy data. Broadly speaking, all root causes of incidents have symptoms which we will then define as attributes. In this dataset only 3 root cause domains were defined as shown in Table 3.

Table 3: Domain of Root Cause

Domain of Root Cause	Notes
Database Issue	Root cause in Database area
Network Issue	Root cause in Network area
CPU Issue	Root cause in CPU area

Another evaluation metric is k-fold cross validation, which creates a process where every sample in the data is included in the dataset in the set at some test. *K* represents a number of folds, usually in ranges of 3 to 10. The data was split into k equal folds and their deviations for each running were analyzed.

Table 4: Error Attributes

Attribute Label	Notes
Disk_Latency	Error related to Disk Latency
Delay	Error related to Network Delay
Error_1000	Error “Conenction_Invalid”
Error_1001	Error “Network_Unreachable”
Error_1002	Error “Trust_Failure”
Error_1003	Error “MP_Processor_Mismatch”
Error_1004	Error “Bad_Unit”
Error_1005	Error “Direct_access_Handle”
Error_1006	Error “Disk_Operation_Failed”

Extracting data from the IT incident portal and labelling the flags “0” and”1” was started after the domain and root cause attributes were defined and labelled. The flag “0” means the error symptom appears and the flag “1” means the error symptom does not appear. 1000 rows of data were prepared with the distribution of the data as described in Table 5 below.

Table 5: Sample 20 of 1000 Rows of Data for IT Incident

Id	CPU_Load	Disk_Latency	Delay	Error_1000	Error_1001	Error_1002	Error_1003	Error_1004	Error_1005	Error_1006	Root_Cause
1	0	0	0	0	1	0	1	0	1	0	CPU_Issue
2	0	0	0	0	0	0	1	0	1	0	CPU_Issue
3	0	1	1	0	0	1	1	0	0	0	CPU_Issue
4	0	1	0	1	1	0	1	0	0	0	CPU_Issue
5	1	1	0	1	0	1	0	0	1	1	Network_Issue
6	0	0	1	1	0	0	0	1	1	1	Network_Issue
7	1	0	0	1	1	0	0	1	0	0	Network_Issue
8	0	0	0	1	1	0	1	0	1	1	Database_Issue
9	0	1	0	0	1	0	1	0	1	1	CPU_Issue
10	0	0	0	1	1	0	1	1	0	1	Network_Issue
11	1	0	0	0	0	0	1	1	0	1	CPU_Issue
12	1	0	1	0	0	0	0	0	1	0	Database_Issue
13	0	1	0	0	0	1	1	0	1	1	CPU_Issue
14	0	1	1	0	0	1	0	0	0	1	Database_Issue
15	1	0	1	0	1	0	0	1	1	0	Network_Issue
16	1	0	0	0	1	0	1	1	0	0	CPU_Issue
17	0	1	0	0	0	1	1	0	1	0	CPU_Issue
18	0	1	0	1	0	0	1	1	0	0	CPU_Issue
19	0	1	1	0	0	1	0	0	1	0	Network_Issue
20	0	1	0	0	0	0	0	1	1	0	CPU_Issue

Table 6 shows that the data distribution is relatively balanced where the composition for each root cause is in the range of 32-34%. The dataset was then divided for training and testing requirement. The composition between the training and testing dataset was 70% : 30% in best practice.

Table 6: Data Distribution

Root Cause	Number of Data	Distribution (%)
Database Issue	340	34%
Network Issue	337	33,7%
CPU Issue	323	32,3%

4.2. Accuracy and Loss

The evaluation results of the proposed method are presented in Table 7. The three test scenarios for the proposed method were accuracy, precision, and recall. The accuracy scenario involved evaluating the set of labels predicted for a sample that must exactly match the corresponding set of labels in the dataset. On the other hand, the precision scenario was used to evaluate the intuitiveness of the classifier in not labelling a negative sample as positive. The recall scenario was used to assess the intuitiveness of the classifier to find all positive samples. As can be seen in Table 7, the accuracy train set of ANN Multiclass in all scenarios was stable with a score of 0.87, whereas the test set is 0.85 for accuracy and 0.84 for both precision and recall. The scenario test results show that the higher the scenario score, the more accurate the prediction.

Table 7: The Scenario Test Result

ANN Multiclass	
Test Scenario	Score
Accuracy (Test Set)	0.85
Accuracy (Train Set)	0.87
Precision (Test Set)	0.84
Precision (Train Set)	0.87
Recall (Test Set)	0.84
Recall (Train Set)	0.87

Figure 5 shows the accuracy and loss of ANN where image (x) and image (y) represent the accuracy and loss of the proposed method with the optimum epoch of 400. The accuracy is stagnant in 0.9 after the 400 epoch. Therefore, the optimum epoch is 400.

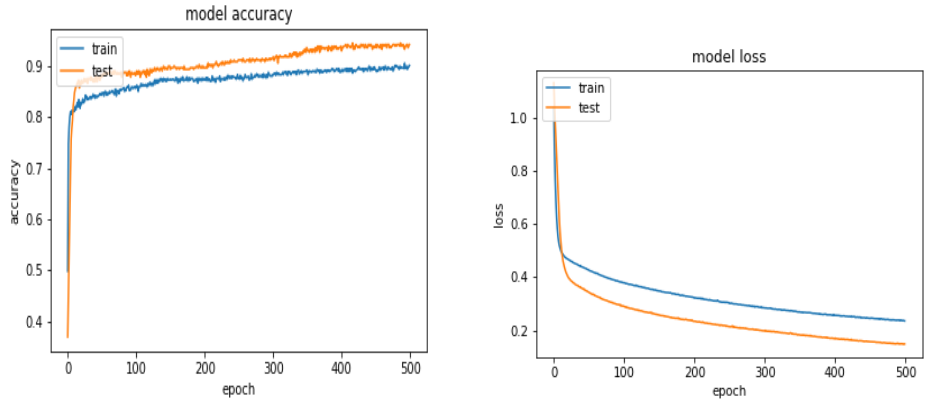

Fig. 5: Accuracy and Loss of ANN

4.3. Comparation with Other Methods

Table 8 compares the proposed method to other methods. The table contains accuracy, precision and recall for training set and testing set using six different algorithms Logistic-Regression, KNN, Decision Tree, Random Forest, ADABoost and XGBoost.

Table 8: Comparison of ANN and Other Methods

Method	Accuracy
Logistic Regression	0.81
KNN	0.81
Decision tree	0.85
Random Forest	0.85
AdaBoost	0.81
XGBoost	0.86
ANN Multiclass	0.85

Table 8 compares the accuracy of six methods to the ANN Multiclass, where all the methods uses dataset with a value of 0.33 for test case. Multiclass ANN Accuracy was 0.85, which shows that the accuracy is not the highest among all methods; however, ANN demonstrates that it has the best accuracy among all proposed methods, with a stability of precision score shown in Table 8. Table 8 indicates that the training set of ANN method is more stable in how the network is supposed to respond to a particular input.

4.4. Discussion

The RCA using the multiclass ANN method has an accuracy of 0.85 and is more stable than the other methods, which have a precision score of 0.87. The use of different methods has the same purpose to predict the upcoming event and prevent failure, such as RCA used in the Indian steel industry conducted by Sarkar et al. (2020) and IT infrastructure monitoring to detect failures conducted by Bhanage et

al. (2021). This study also analyzed the model loss and model accuracy during the epoch. Based on the results of the observation, the optimal epoch was 400, with the most reliable and highest precision. The accuracy of ANN can't be enhanced by increasing the dataset or epoch because it can overshoot the result; this limitation can be improved by future work that combines the method into a single algorithm.

5. Conclusion and Future Work

Root cause analysis in this method used multiclass ANN to predict the output. The dataset size was 1000 with 3 output targets, in particular, network issue, database issue, and CPU issue. 20 hidden nodes and batch size of 10 with 200 epochs for the initial condition was used in this method. When evaluated using the categorical cross entropy and k-fold cross, this method has an accuracy of 0,85, and if 400 epoch was used, the accuracy increased to 0,9. This means the optimum epoch for this method is 400. AdaBoost and XGBoost were used to compare with the current method; AdaBoost has an accuracy of 0,81, while XGBoost is 0,86. The ANN Multiclass outperforms these two methods and has an outstanding precision score of 0,84. It can be concluded that the ANN method is more stable in terms of how the network is supposed to respond to a particular input.

For future work, it can be improved by using more dataset and epoch. However, it has potential to overshoot the result, so it must be performed with caution. After the predicted result is known, it can be responded to by eliminating the root cause using auto healing algorithm to take out the problem. The ANN can also be combined with other methods, such as XGBoost and AdaBoost through the ensemble method. Hyperparameter tuning will be performed to improve the models.

References

Ahmed, J., Johnsson, A., Yanggratoke, R., Ardelius, J., Flinta C., & Stadler, R. (2016). Predicting SLA conformance for cluster-based services using distributed analytics. IEEE/IFIP Network Operations and Management Symposium, 848-852

Bhanage, D.A., Pawar, A. V., & Kotecha, K. (2021). IT Infrastructure Anomaly Detection and Failure Handling: A Systematic Literature Review Focusing on Datasets, Log Preprocessing, Machine & Deep Learning Approaches and Automated Tool. IEEE Access

Brownies, J. (2021). Multi-Class Classification Tutorial with the Keras Deep Learning Library. Retrieved from: https://machinelearningmastery.com/multi-class-classification-tutorial-keras-deep-learning-library/

Cota, D. & Sapp, C. (2020). Introduction Deep Learning Abstraction Methods. Retrieved from: https://www.gartner.com/document/3985472?ref=solrrqp&refval=310153191

Du, M., Li, F., Zheng, G., & Srikumar V. (2017). DeepLog: Anomaly detection and diagnosis from system logs through deep learning. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, 1285–1298.

Elbasani, E. & Kim, J. (2021). LLAD: Life-log anomaly detection based on recurrent neural network LSTM. Journal of Healthcare Engineering, 1–7

Franco, D. G. & Santurro, M. (2020). Machine Learning, Artificial Neural Network and Social Research. Quality & quantity, 55, 1008-1010

Jiang, W. & Bai, Y. (2022). APGNN: Alarm Propagation Graph Neural Network for fault detection and alarm root cause analysis. Computer Networks, 220, 109485

Konno, S. & Defago, X. (2019). Approximate QoS rule derivation based on root cause analysis for cloud computing. In 2019 IEEE 24th Pacific Rim International Symposium on Dependable Computing (PRDC), 33-42

Ma, Q., Li, H., & Thorstenson, A. (2021). A Big Data-Driven Root Cause Analysis System: Application of Machine. Computers & Industrial Engineering, 16

Mauritius, T. & Binsar, F. (2020). Cross-Industry Standard Process for Data Mining (CRISP-DM). Retrieved from: https://mmsi.binus.ac.id/2020/09/18/cross-industry-standard-process-for-data-mining-crisp-dm/

Molan, G. & Molan, M. (2021). Theoretical Model for Accident Prevention Based on Root Cause. Safety and Health at Work, 12, 42-50

Murray, G. (2021). Understanding the Application of AIOps Discipline Within IT Operations. Retrieved from: https://www.gartner.com/document/3995591?ref=solrAll&refval=317064134

Priscillia, S., Schillaci, C., & Lipani, A. (2021). Flood Susceptibility Asessment Using Artificial Neural Networks in Indonesia, Artificial Intelligence in Geosciences, 2, 218-219

Prosise, J. (2021). Multiclass Classification with Neural Network. Retrieved from Atmosera: https://www.atmosera.com/blog/multiclass-classification-with-neural-networks/

Qin, K., Chen, L., Shi, J., Li, Z., & Hao, K. (2022) Root cause analysis of industrial faults based on binary extreme gradient boosting and temporal causal discovery network. Chemometrics and Intelligent Laboratory Systems, 225, 104559

Saha, A. & Hoi, S. (2022). Mining Root Cause Knowledge from Cloud Service Incident Investigation for AIOps, arXiv preprint arXiv:2204, 1-2

Sarkar, S., Ejaz, N., Kumar, M., & Maiti, J. (2021). Root Cause Analysis of Incidents Using Text Clustering and Classification Algorithm, ICETIT, 708

Su, Y., Zhao, Y., Niu, C., Liu, R., Sun, W., & Pei, W. (2019). Robust anomaly detection for multivariate time series through stochastic recurrent neural network. 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2828–2837.

Susac, M. Z., Has, A., & Knezevic, M. (2021). Predicting Energy Cost of Public Building by Artificial Neural Network, CART, and Random Forest, Neurocomputing, 439, 225-227

Tan, Y. & Gu, X. (2010). On predictability of system anomalies in real world. In 2010 IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, 133-140

Velasquez, R. A. & Mejia Lara, J. V. (2020). Root cause analysis improved with machine learning for failure. Engineering Failure Analysis, 115, 104684

Wang, L. & Liu, Z. (2021). Data-driven Product Design Evaluation Method Based on Multi-stage Artificial Neural Network, Applied Soft Computing, 103, 7-8

Weng, J. H., Wang, J., & Yang, Y. (2018). Root cause analysis of anomalies of multitier services in public clouds. IEEE/ACM Transactions on Networking, 26, 1646–1659

Wu, L., Bhogatinovski, J., Nedelkoski, S., Acker, A., Schmidt, F., Wittkopp, T., Becker, S., Cardoso, J., & Kao, O. (2021). Artificial Intelligence for IT Operations (AIOPS) Workshop White Paper. Performance Diagnosis in Cloud Microservices using Deep Learning, 6