Evaluation of a Bayesian Machine Learning –Based and Regression Analysis -Based Performance Prediction Model for Computer Networks

Akinyemi Bodunde.O., Aladesanmi Temitope.A., Oyebade Adedoyin.I., Aderounmu Ganiyu.A, and Kamagaté Beman.H.

***Abstract*—This study accessed the operation of the purported Bayesian Network machine learning-based prediction model for network performances in the face of security risks. This was with a view to predetermine the effect of network security risk factors on the network Confidentiality, Integrity, and Availability. The performance of the proposed BN prediction model was benchmarked with the existing Regression Analysis (RA) prediction model using Prediction Accuracy, Reliability, and Availability as the evaluation measures of the model performance. The simulation result proved that the prediction accuracy of the Bayesian Network model is higher in all the measures, the reliability is high, but the availability rate is relatively lower. The results showed that the proposed model is able to obtain better effectiveness in optimizing the network performances by gathering information about the inherent network risks to deliver the higher prediction accuracy, higher reliability, and relative availability. This implied that the BN scheme is a robust computational scheme that improves the capabilities properties of the prediction model despite its computational complexity as compared to the RA model. It was concluded that the proposed prediction model measures the security risk quantitatively and predicts network performances using objectives metrics and eventually improves the overall network performance efficiencies.**

***Index Terms*—Bayesian network, computer networks, prediction model, regression analysis, machine learning.**

I. Introduction

The objective of this study has been to develop a framework that automatically performs predictions on network security situations. The study was motivated by the demands of the knowledge of network security risk management by emerging threats, vulnerabilities, and risks in the networks. The task is challenging due to the diversity of the Internet which has not yet been completely understood and well-modeled. [1] discussed the significance of a flexible decision support system for network security managers deciding between interventions, using Bayesian Networks (BNs) models that will ascertain that network services are delivered at the right time, available in the right place, present in the right shape, satisfying quality requirements and obtained at the lowest possible costs.

In [2], the BN prediction model was built using information obtained from experts' knowledge elicitation. Bayesian probability distributions updating using Markov Chain Monte Carlo simulations ensures that the model is not static, but quickly adapts to new input and incorporates it with prior expert opinion in a mathematically tractable manner. The prediction model was applied to predetermine the effect of network security risk factors on network confidentiality, integrity, and availability. The proposed scheme measures the security risk quantitatively and predicts network performances using objectives metrics.

In [2], the proposed prediction model formulated as Bayesian Network model simulation was carried out. Java programming language tools were used to simulate the model formulated. The core of the simulation program was written in Java programming language using Ms-DOS as the execution environment.

There is a need to check whether or not the proposed prediction model is effective and have an impact on security risk management. The three-quality attribute of a prediction model that is required in a network security management system is accuracy, reliability, and availability of the model.

Assessment of the prediction model accuracy is required to trust the data that is collected, develop consensus about the results and consistently predicts values with acceptable accuracy. Assessment of the reliability of a prediction model is required to ensure that predictions are not prone to human errors and retains its functionality over a period of time. It is the degree to which the prediction model can be relied upon to perform its intended function Reliability is defined as the ability of software to maintain a specified level of performance within the specified usage conditions [3]. Assessment of the availability of the prediction model is required to ensure that it does not fail very often and, when it does, it can be quickly returned to service. Availability is the probability that the prediction model is operating according to requirements and will perform a required function without failure under defined conditions for a defined period of time.

The rest of this paper is arranged as follows: Section 2 discusses the related works while Section 3 described the mathematical representations of the performance metrics used while Section 4 described the evaluation results and the conclusions are discussed in Section 5.

Manuscript received June 7, 2019; revised October 28, 2019.

Akinyemi Bodunde.O., Oyebade Adedoyin.I., and Aderounmu Ganiyu.A. are with Department of Computer Science and Engineering, Obafemi Awolowo University, Ile-Ife Nigeria (e-mail: bakinyemi@oauife.edu.ng, sacnet2010@yahoo.com, gaderoun@oauife.edu.ng).

Aladesanmi Temitope.A. is with INTECU, Obafemi Awolowo University, Ile-Ife Nigeria (e-mail: taladesanmi@oauife.edu.ng).

Kamagaté Beman.H. is with Laboratoire LARIT- Cocody Danga Abidjan, Cote D'ivoire (e-mail: beman2017@gmail.com).

doi: 10.18178/ijfcc.2019.8.4.555

II. Related Works

There has been quite a lot of work done in the area of network security risk management. These literature have been presented in [4]. The main issues being addressed recently is the automation of the risk management processes. [5] addressed the problem of automating the network risk management process by presenting an algebraic specification for network security risk management. The study allows reasoning and proving properties about scenarios of attacks using natural algebra. Many-sorted signatures and first-order predicate logic were used to model the risk management process. This allows reasoning on attacks and helps build security decisions. The weakness of this approach is that the risk management decisions from the algebraic point of view are more abstract and very complex specifications. In furtherance, [6] presented a novel method for network security risk management. Also, [7] provided an abstract reduction model for decision making under security risks in a computer network environment. The proposed method relies on a many-sorted algebraic signature and on a rewriting system. The study presented a reduction system that permits to automate the reasoning made by security experts when performing Risk analysis.

Also, [8] presented a prediction model of network security situation based on Regression Analysis. Linear regression was proposed as a method for network security situation evaluation. A prototype system was designed for data collection and regression fitting. The study shows that the Regression Analysis complexity rate is low and less time-consuming. The regression prediction model reflects the physical network's security situation in a certain range of threshold value. The weakness of this method is that it can only work on a small dataset and lack of scalability.

[9] addressed the problem of security risk assessment and mitigation by proposing a dynamic security risk management using Bayesian Attack Graphs (BAG). The Bayesian Attack Graphs (BAGs) is used to model vulnerability exploitations in a test network. It was shown that the attack graphs-based risk management framework using Bayesian networks enables a system administrator to quantify the chances of network compromise at various levels and also help in risk mitigation procedure by identifying the most critical and probable attack path in the network. Conversely, the attack graphs can get complex as the network attacks sequences increases i.e. lack of scalability. It is also a scenario-based approach.

[1] revealed the theoretical background of the performance prediction model for data communication network security risk. This study presented procedures that support dynamic decision-support model that will predetermine the impact of network security risk on the selected network domain is given the causal- effect model.

[2], modeled the proposed performance prediction model for data communication network security risk. the computational prediction model was presented as a Bayesian Network (BN)-based problem. A problem domain was selected, its network security risks causality model was designed using Knowledge Engineering-based approach, and its network structure (the structural model) was obtained from the designed causality model. The structure was then quantified by eliciting experts' opinions on the likelihood of occurrence of the network security risks using questionnaire. The probability distributions were obtained using sampling-based sensitivity analysis method while probabilistic inferences were made using approximate inference algorithm.

[2] demonstrated the use of the proposed model to predict the impact of security risks on the performances of the network so as to assess the future capacity needs and associated recommendations for performance monitoring and analysis of the network system.

This paper attempts to use various security performance assessment methods to validate the functioning of the proposed model. The proposed BN model performances were assessed by benchmarking it with an existing Regression Analysis model using prediction accuracy, reliability, and availability as performance measures. Regression Analysis was selected simply because it is one of the state-of-the-art approaches that can handle uncertainty using subjective data.

III. Evaluation Measures Used

In this study, the performance metrics employed to assess the proposed model are prediction accuracy, reliability, and availability.

A. Prediction Accuracy

Numerical comparison is performed between elicited data i.e. unconditional prior distributions which now serve as the actual or historical and predicted data from observations. The objective is to measure the deviation of prediction from history and report it as a percentage (%). The quality of the model is evaluated using the prediction error as described in [10] and [11]. The lower the error, the higher is the quality of the model. In this study, the prediction accuracy is assessed by measuring the following prediction quality indicators:

1) Mean Prediction Error (MPE): This measures the integral of average deviation of predicted data from the actual data over a period of time. The MPE is a measure of average cumulative error or bias. This denotes that MPE must be as close to zero as possible i.e. minimum bias. It was estimated as follows:

$$MPE=\frac{1}{n}\sum_{t=1}^{n}(D_t - P_t) \quad (1)$$

where
n is the number of time of samples
$t=1,..., n$ are the starting and ending instants of the samples.
D_t is the priors distributions i.e. the actuals data at time t
P_t is the poste`rior distributions i.e. the predicted data at time t

2) Mean Absolute Deviation (MAD): This measures the integral of average absolute values of the deviation of predicted data from the actual data over a period of time. This is also known as absolute error. The equation 4.2 denotes that MAD must be as small as possible.

$$MAD=\frac{1}{n}\sum_{t=1}^{n}|D_t-P_t| \quad (2)$$

where:

n is the number of time of samples.

$t=1,\ldots, n$ are the starting and ending instants of the samples.

D_t is the priors distributions i.e. the actual data at time t.

P_t is the posterior distributions i.e. the predicted data at time t.

Thus, the prediction accuracy of the model is determined by comparing the MADs and MPEs of the model, to ascertain if accuracy is acceptable.

a) If the result yields low MPE and low MAD, this implies that the prediction errors are small and unbiased.

b) If the result yields high MPE and high MAD, this implies that the predictions are inaccurate and biased.

c) If the result yields low MPE and high MAD, this implies that the predictions are on average.

In a nutshell, the result that yields the smallest MAD and has bias close to zero usually gives good accuracy output.

3) Magnitude of Relative Error (MRE): MRE is a normalized measure of the discrepancy between actual values and predicted values, given by:

$$MRE=\frac{1}{n}\sum_{t=1}^{n}\frac{|D_t-P_t|}{D_t} \quad (3)$$

This measures the average relative discrepancy, which is equivalent to the average error relative to the actual effort in the prediction. Thus, for a prediction model to be considered accurate, MRE ≤ 0.25

4) Pred: Pred is a measure of what proportion of the predicted val`ues has MRE less than or equal to a specified value, given by:

$$Pred(q)=\frac{k}{n} \quad (4)$$

where:

q is the specified value

k is the number of cases whose *MRE* is less than or equal to q, and

n is the total number of cases in the dataset.

In order for a prediction model to be considered accurate either Pred(0.25) $\geq$ 0.75 or Pred(0.30) $\geq$ 0.70 is suggested in the literature**.** The higher the value of the Pred, the more accurate the model is.

B. Reliability

In this research, the performance prediction model is considered a repairable computer-based system in which its reliability can be measured using the time between failures model called Mean-Time-Between-Failure (MTBF). MTBF is the mean operating time (up time) between failures during the normal working life or useful life of a specified item of equipment or a system. It is a statistical mean value for error-free operation of a system. The factors influencing the reliability of this model to depend upon the following information:

1) Number of Failures: This denotes the total number of failures observed until execution time from the beginning of model execution. It is the observed trend of cumulative failure count of a program i.e. counting failures in periodic intervals

2) Total Operating Time**:** This is the raw execution time of the system. It is also referred to as the execution exposure that software receives through usage. It is usually measured in the central processing unit (CPU) execution time.

Thus, the probability that a failure will occur in the system is expressed as the failure rate (λ) denoted as

$$\lambda=\frac{Number\ of\ Failures}{Total\ Operating\ Time} \quad (5)$$

The MTBF always refers to the phase with constant failure rate (i.e. without early or wear failures). Thus, MTBF is the inverse of the failure rate in the constant failure rate phase.

$$MTBF=\frac{1}{\lambda} \quad (6)$$

Since reliability is the probability of failure-free operation of the system, MTBF is expressed as a measure of how reliable the model is. The higher the MTBF is, the higher the reliability of the model. MTBF is usually expressed in units of hours.

C. Availability

The availability of a system for a period (0,t) is the probability that the system is available for use at any random time in (0,t). Availability can thus be thought of as the probability that an item or system is up at any instant required.

In this research, availability of the prediction model is considered as the measurement of the proportion of time for which the model is able to perform its function and how likely it is available for use. It is defined as the probability that a program is operating according to requirements at a given point in time and is defined as:

$$Availabilty\left(A(t)\right)=Total\ Execution\ Time-\left(\frac{Total\ Uptime}{Total\ Downtime}\right) \quad (7)$$

where:

Total Uptime is the length of time for a failure discovery or detection.

Total Downtime i.e. length of time for a repair.

In this study, the uptime is obtained based on the failure rate while the downtime is computed as the average time to fix failures encountered in a program until the observable outcome of program execution is the same as the expected outcome.

IV. Model Evaluation and Comparison

A simulation program was developed to provide performance analysis of the prediction model. The performance of the simulation program for the prediction

model that was developed for the proposed model i.e. Bayesian network-based (BN) model was compared with the performance of one of the existing models i.e. the Regression Analysis- based (RA) model. For the purpose of performance comparison, the following performance metrics are used:

1) Prediction Accuracy of the model using the BN model and RA model
2) Reliability of the model using the BN model and RA Model
3) Availability of the model using the BN model and RA model

A. Prediction Accuracy Measures of the Model

A simulation was performed to evaluate and compare the prediction accuracy of the two prediction models under consideration quantitatively using the measures presented in Equations (1) and (2). The values of the prediction accuracy measures achieved by each of the prediction models for the network dataset used for the simulation is as shown in Table 1. The values in this table are the mean of the values obtained from the 15 different risk factors. The actual data are the elicited unconditional probability distributions of the network security risks while the predicted data was simulated for each risk factor following the simulation results presented in [2]. The prediction errors, absolute prediction errors and relative errors of the two models are presented in Table I.

Fig. 1 shows the degree of the prediction trend for both models. It reveals that the trend is seasonal in nature. Fig. 2 shows the degree of prediction errors, while Fig. 3 shows the degree of absolute prediction errors while Fig. 4 shows the degree of relative errors of the two models under consideration. Also, Equations 1, 2, 3, and 4 were simulated to evaluate the MPE, MAD, MRE and Pred values of both models. Table II shows the prediction accuracy values for the two prediction models.

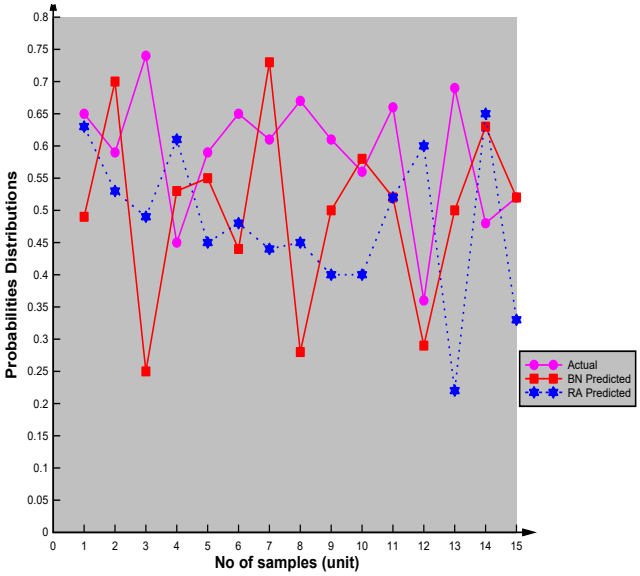

Fig. 1. Prediction accuracy factors of the two models.

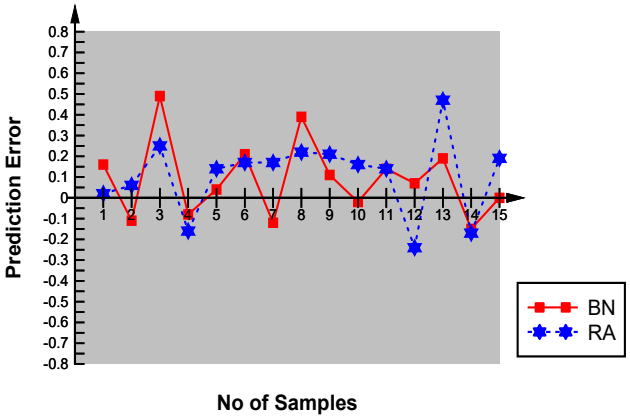

Fig. 2. The degree of prediction errors of the two models.

TABLE I: PREDICTION ERRORS, ABSOLUTE PREDICTION ERRORS AND RELATIVE ERRORS OF THE TWO MODELS

	Actual Data (D_t)	BN Model				RA Model			
		Predicted Data (P_t)	Prediction Error	Absolute Error	Relative Error	Predicted Data (P_t)	Prediction Error	Absolute Error	Relative Error
1	0.65	0.49	0.16	0.16	0.25	0.63	0.02	0.02	0.03
2	0.59	0.70	-0.11	0.11	0.18	0.53	0.06	0.06	0.11
3	0.74	0.25	0.49	0.49	0.66	0.49	0.25	0.25	0.34
4	0.45	0.53	-0.08	0.08	0.19	0.61	-0.16	0.16	0.37
5	0.59	0.55	0.04	0.04	0.06	0.45	0.14	0.14	0.23
6	0.65	0.44	0.21	0.21	0.33	0.48	0.17	0.17	0.27
7	0.61	0.73	-0.12	0.12	0.20	0.44	0.17	0.17	0.27
8	0.67	0.28	0.39	0.39	0.58	0.45	0.22	0.22	0.33
9	0.61	0.50	0.11	0.11	0.18	0.40	0.21	0.21	0.34
10	0.56	0.58	-0.02	0.02	0.03	0.40	0.16	0.16	0.29
11	0.66	0.52	0.14	0.14	0.22	0.52	0.14	0.14	0.22
12	0.36	0.29	0.07	0.07	0.20	0.60	-0.24	0.24	0.66
13	0.69	0.50	0.19	0.19	0.27	0.22	0.47	0.47	0.68
14	0.48	0.63	-0.15	0.15	0.32	0.65	-0.17	0.17	0.36
15	0.52	0.52	0.00	0.00	0.01	0.33	0.19	0.19	0.37

Table II confirms strong evidence that the differences of the Bayesian network model from the RA model are significant. BN model's MPE value is significantly lower than the RA model, this makes the BN model less bias (more neutral) and more accurate. BN model's MAD value is significantly lower than the RA model, this implies good performance. BN model's MRE ≤ 0.25, Pred(0.30) ≥ 0.70, this makes BN model more accurate. The simulation result

shows that the prediction accuracy of the Bayesian network model is better in all the measures.

Thus, a measure of the deviation of predictions from history for BN model is 9% resulting in 91% accuracy, while a measure of the deviation of predictions from history for RA model is 11% resulting in 89% accuracy. This means that the performance comparison of the prediction accuracy of the BN model gives an increase of 2% over the RA model.

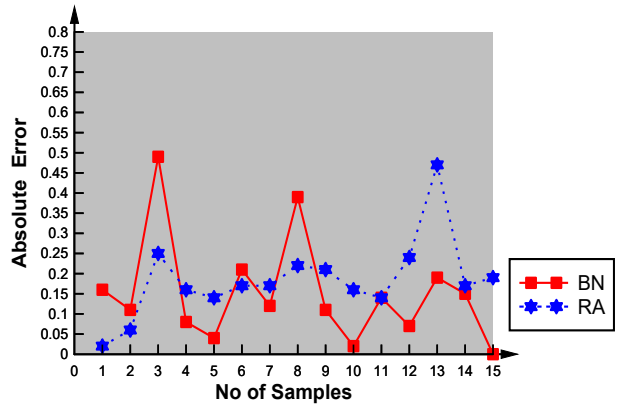

Fig. 3. The degree of absolute prediction errors of the two models.

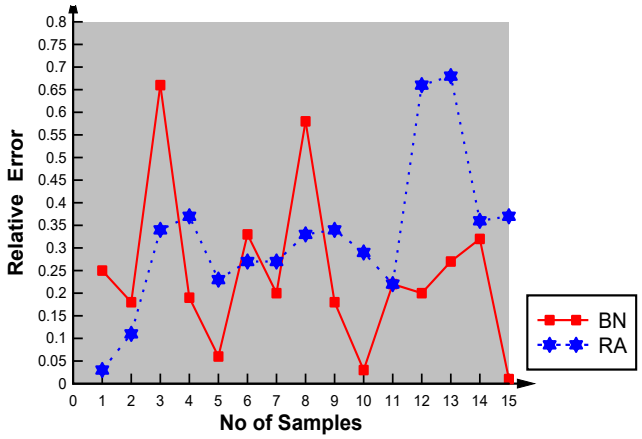

Fig. 4. The degree of relative errors of the two models.

TABLE II: PREDICTION ACCURACY MEASURES OF THE TWO MODELS

Model	MPE (BIAS)	MAD	MRE	Pred (0.25)	Pred (0.30)
BN	0.09	0.15	0.24	0.67	0.73
RA	0.11	0.19	0.32	0.26	0.47

B. *Reliability Measures of the Model*

The reliability of the two prediction models is evaluated solely based on failure observations from testing or operation of the simulation program. The simulation was performed for 100 seconds of the time unit. A counting function was assigned to keep track of the cumulative number of failures a given system has had from time 0 seconds to time 100 seconds. During the simulation of this prediction model, the conditions and the assumption that holds for the evaluation of the reliability of the models are treated offline in this paper.

During the 100 seconds of the test, the failures observed were used as the simulation data. The data were reported as times between failures i.e. observed failure occurrences in terms of execution time. The simulation result for the two models under consideration is given in Table III. The numbers of failures against seconds of the execution time of the CPU time of the two models were compared. A study of the data in Table III and of the plot in Fig. 5 indicates that the failure rate (the number of failures per seconds) decreases with test time and thus, the cumulative failure rate increases with test time. The decrease of failure rates in the BN model is significantly lower than the RA model.

It was discovered that the increase in time-between-failure of the BN model is significantly higher than the RA model. The reliability, R(t) of the two models are then compared in Fig. 6. It shows that the reliability of the BN model is significantly higher than the RA model. The higher the reliability of a model, the better is the performance of the model.

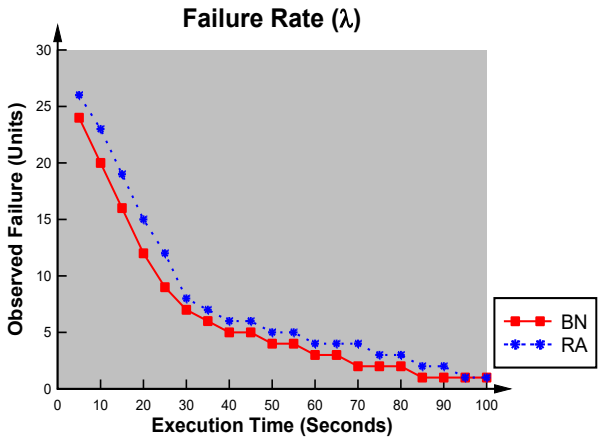

Fig. 5. Number of failures per seconds of the two models.

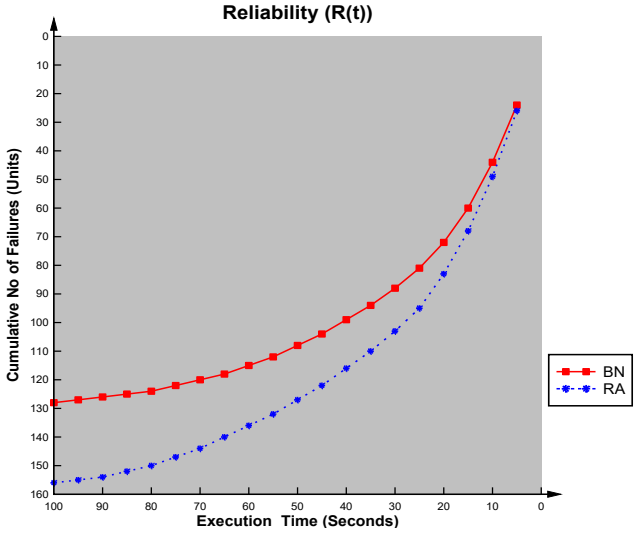

Fig. 6. Plot of reliability of the two models.

The simulation data in Table III shows that the prediction model that utilizes BN prediction model has a low failure rate. Similarly, Equation 5 was simulated to evaluate the failure rate of both models. During the 100 seconds of the total execution time of the simulation, the BN model has *1.28 failure rates,* while RA model has *1.56 failure rates*. This means that the performance comparison of the failure rate of the BN model gives a decrease of 17.95% over the RA model.

Also, Equation 6 was simulated to evaluate the MTBF and reliability of both models. The MTBF is the inverse of the failure rate, which results for BN model to *1/1.28=0.78*

seconds, which implies 78% reliability, while the MTBF for RA model results to *1/1.56=0.64 seconds,* which implies 64% reliability. This means that the performance comparison of the MTBF of the BN model gives a 14% increase over the RA model. Thus, higher reliability was achieved in the proposed model.

TABLE III: OBSERVED AND ESTIMATED VALUES FOR TIME-BETWEEN FAILURE MODEL

Execution time (Seconds)	BN		RA	
	Observed No of Failure (Units)	Cumulative Failures (Units)	Observed No of Failure (Units)	Cumulative Failures (Units)
5	24	24	26	26
10	20	44	23	49
15	16	60	19	68
20	12	72	15	83
25	9	81	12	95
30	7	88	8	103
35	6	94	7	110
40	5	99	6	116
45	5	104	6	122
50	4	108	5	127
55	4	112	5	132
60	3	115	4	136
65	3	118	4	140
70	2	120	4	144
75	2	122	3	147
80	2	124	3	150
85	1	125	2	152
90	1	126	2	154
95	1	127	1	155
100	1	128	1	156

C. Availability Measures of the Model

This simulation also measured how likely the prediction model is available for use for the two models under consideration, taking into account the time between failures and repairs. The time between failures and repairs follows the assumption made during the reliability measures of the model. The repairs are made by modifying the design to make it robust against conditions that can trigger a new failure. The theory is that each failure is fixed as it is discovered. However, the availability of the two prediction models under consideration is evaluated based on the uptime i.e. length of time for a failure discovery or detection and downtime i.e. length of time for a repair of each model.

The simulation was performed for 100 seconds of the time unit. A counting function was assigned to keep track of the cumulative number of failures a given system has had from time 0 seconds to time 100 seconds. The lifetime of each model was determined using the failure detection rate, d(t), which is a step function that jumps up one every time a failure occurs and stays at the new level until the next failure. The downtime of each model was determined using the failure repair rate, r(t), which is a step function that jumps up one every time a failure occurs and record the length of time is being repaired until the next failure. Each model has its own observed d(t) and r(t) function over time.

Table IV shows the functions observed to perform a comparison of availability factors of the two models under consideration i.e. Observed failure discovery and repair rate. The length of time between defect discoveries against the number of observed failures depicted in Fig. 7 shows that the failure discovery rate increases as the observed failure increases, while the time of execution decreases. The decrease in the number of failures in the code results in the decrease in the failure discovery rate.

TABLE IV: OBSERVED FAILURE DISCOVERY AND REPAIR RATE OF THE TWO MODELS

Execution time (Seconds)	BN			RA		
	Failure detected and repaired (unit)	Failure discovery rate (sec)	Failure Repair rate (sec)	Failure detected and repaired (unit)	Failure discovery rate (sec)	Failure Repair rate (sec)
5	24	0.21	4.80	26	0.19	5.00
10	20	0.42	2.00	23	0.38	2.30
15	16	0.63	1.07	19	0.58	1.27
20	12	0.83	0.60	15	0.77	0.75
25	9	1.04	0.36	12	0.96	0.48
30	7	1.25	0.23	8	1.15	0.27
35	6	1.46	0.17	7	1.35	0.20
40	5	1.67	0.13	6	1.54	0.15
45	5	1.88	0.11	6	1.73	0.13
50	4	2.08	0.08	5	1.92	0.10
55	4	2.29	0.07	5	2.12	0.09
60	3	2.50	0.05	4	2.31	0.07
65	3	2.71	0.05	4	2.50	0.06
70	2	2.92	0.03	4	2.69	0.06
75	2	3.13	0.03	3	2.88	0.04
80	2	3.33	0.03	3	3.08	0.04
85	1	3.54	0.01	2	3.27	0.02
90	1	3.75	0.01	2	3.46	0.02
95	1	3.96	0.01	1	3.65	0.01
100	1	4.17	0.01	1	3.85	0.01

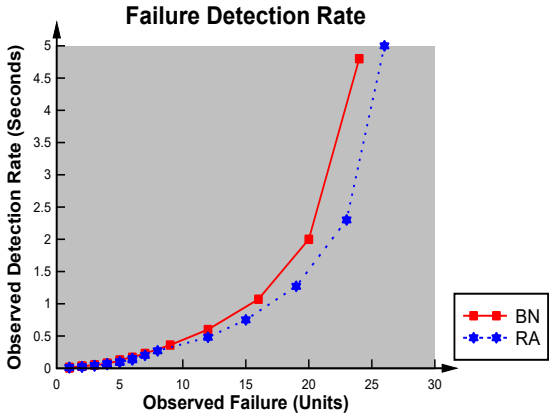

Fig. 7. Failure discovery rate of the two models.

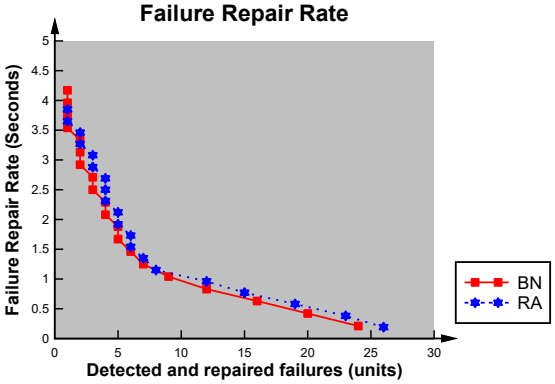

Fig. 8. Failure repair rate of the two models.

Fig. 8 shows that the failure repair rate decreases as the number of failures detected (and repaired) increase starting from the start of the simulation. It reveals that the frequency of repairs is decreasing at a roughly constant rate. It was observed that the failure repair rate is proportional to the observed failure in the code. Each time a failure is repaired; there is a less total failure in the code, so the failure repair rate decreases as the number of defects detected (and repaired) increase.

Fig. 9 shows that the availability, A(t) of the two models under consideration. The rates of failure detection and repair of the two models were compared. It reveals that the failure discovery rate decreases as the number of repairs increases in the code. Similarly, Equation 7 was simulated to evaluate the availability rate of both models. During the 100 seconds of the total execution time of the simulation, under the BN model, the sum of the observed length of time for a failure discovery or detection is *43.77 seconds* while the sum of the observed length of time for repair is *9.85 seconds,* the resulting Availability rate, (A(t)) for the BN model is given as *100 - (43.77 / 9.85)= 95.56 seconds.* For RA model, the sum of the observed length of time for a failure discovery or detection is *40.38 seconds* while the sum of the observed length of time for repair is *11.07 seconds,* the resulting Availability rate, (A(t)) for RA model is given as *100 - (40.37 / 11.07) = 96.35 seconds.* This implies that out of the 100 seconds execution time of the simulation, BN model was available for *95.56 seconds* which implies *95.56%* availability rate and RA model was available for *96.35 seconds* which implies *96.35%* availability rate. This means that the performance comparison of the availability rate of the two models results in a difference of *0.79 seconds* which means that the BN model gives a decrease of 0.8% over the RA model.

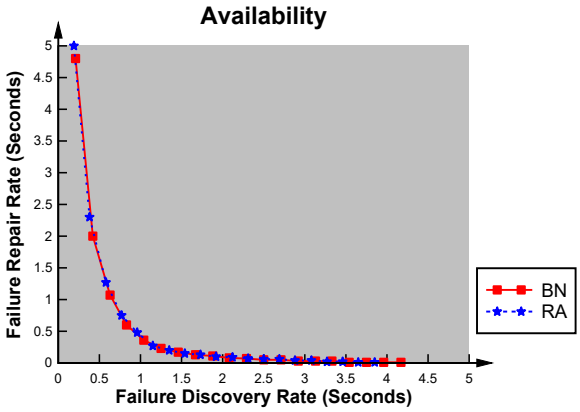

Fig. 9. Availability measures of the two models.

Model assume they are negligible or handled by the statistical fit of the software reliability growth model to the data. This connotes that the BN model still delivers relative availability despite the mathematical complexity rate of the model.

D. Evaluation Results

According to the Occam's razor principle, if two models of different complexity both fit the data approximately equally well, then the simpler one usually is a better predictive model in the future. It is obvious in this simulation that:

1) The Bayesian network prediction model has been able to achieve significantly better prediction accuracy than the regression-based models for the dataset used. This implies that the BN model has a better performance to trust the data that is collected, develop consensus about the results and consistently predicts values with acceptable accuracy.
2) There is a reduction in the failure rate in the BN model than the RA model, thus the BN model has a better performance in increasing the effectiveness of the prediction model in terms of fulfilling its reliability requirements during or after a given time span under given application conditions. This implies that the BN model is a robust computational model that provides for higher MTBF thereby improving the reliability properties of the prediction model.
3) There is a reduction in the rate of availability in the BN model. The RA model does better but with a little significance, thus the RA model has a better performance in increasing the effectiveness of the prediction model in terms of higher availability rate. This is because, RA model have relatively low memory and computation overhead compared to BN model, a finite amount of code of BN model have a finite number of defects. Repair and new functionality may introduce new defects, which increases the original finite number of defects. BN model explicitly accounts for new defect introduction during the test while the RA model assumes they are negligible or handled by the statistical fit of the software reliability growth model to the data. This connotes that the BN model still delivers relative availability despite the mathematical complexity rate of the model.

V. Summary and Conclusion

In this paper, techniques for assessing the suitability of the proposed model presented by [2] have been demonstrated. The performances of the Bayesian Network –based model were assessed by benchmarking it with an existing regression analysis model using prediction accuracy, reliability, and availability as performance measures.

In comparison with the Regression Analysis model, the prediction accuracy of the Bayesian Network model is higher in all the measures, the reliability is high, but the availability rate is lower. The result from the performance evaluation shows that there is an improvement in the prediction of security situations in a network using BN prediction model. It is obvious in the simulation that the BN model has a better performance in optimizing and in increasing the effectiveness of the network in terms of its Confidentiality, Integrity, and Availability. It has been shown that the proposed model optimizes the network performances by gathering information about the inherent network risks to deliver the higher prediction accuracy, higher reliability, and relative availability despite the computational complexity of the scheme.

In conclusion, the proposed prediction model for risk management in a network can be adapted by the network administrators for more effective network management in a minimum time and at a minimum expense (resources) and provision of network services delivered at the right time, available at the right place, present in the right shape, satisfying the quality requirements and obtained at the lowest possible costs.

VI. Recommendation for Future Work

A number of open problems needed to be solved to allow the development of a truly general prediction system. These problems suggest a variety of research directions that need to be pursued to make such a system feasible. One such direction would be to investigate into allowing automatic learning of the structure of the probabilistic model. The current framework requires that the model is specified explicitly. It could also be adapted to a client-server with a distributed administration and peer-to-peer networks.

Also, a simulator should be designed to assist in simulating and evaluating the performances of the model. Either an excessively optimistic or pessimistic expectation of the quality of these prior beliefs will distort the entire network and invalidate the results. This suggests that the performance of the Bayesian network models may vary depending on the characteristics of the dataset and/or depending on what simulator tools are used. Also, the availability rate of the model should also be taken into considerations.

Acknowledgement

This Research was funded by the TETFund Research Fund and Africa Centre of Excellence OAK-Park.

References

[1] B. O. Akinyemi, A. O. Amoo, and E. A. Olajubu, "An adaptive decision-support model for data communication network security risk management," *International Journal of Computer Applications*, vol. 106, no. 8, pp. 1-7, 2014.

[2] B. O. Akinyemi, A. O. Amoo, and G. A. Aderounmu,. "Performance prediction model for network security risk management," *Communications on Applied Electronics*, vol. 2, no. 8, pp. 1-7, 2015.

[3] J. D. Musa, A. Iannino, and K. Okumoto, *Software Reliability: Measurement, Prediction, Application*, *Professional Edition: Software Engineering Series*, New York: McGraw–Hill, 1990.

[4] B. O. Akinyemi, A. I. Oyebade, A. O. Amoo, T. O. Oyegoke, T. A. Aladesanmi, and G. A. Aderounmu, "System simulation of a Bayesian network-based performance prediction model for data communication networks," *International Journal of Computer*, vol. 31, no. 1, pp. 119-136, 2018.

[5] M. Hamdi and N. Boudriga, "Algebraic specification of network risk management," in *Proc. the ACM Workshop on Formal Methods in Security Engineering*, Washington, D.C., 2003, pp. 52-60.

[6] M. Hamdi, N. Boudriga, J. Krichene, and M. Tounsi, "NetRAM: A novel method for network security risk management," in *Proc. the Seventh Nordic Workshop on Secure IT System (NordSec)*, Gjovik, Norway, 2003, pp. 25-35.

[7] M. Hamdi and N. Boudriga, "An abstract reduction model for computer security risk," in *Proc. IFIP World Computer Congress (WCC-SEC)*, Toulouse, France, 2004.

[8] W. Xia and H. Wang, "Prediction model of network security situation based on regression analysis," in *Proc. IEEE International Conference on Wireless Communications, Networking and Information Security*, Beijing, China, 2010, pp. 616-619.

[9] N. Poolsappasit, R. Dewri, and I. Ray, "Dynamic security risk management using Bayesian attack graphs," *IEEE Transactions on Dependable and Secure Computing*, vol. 9, no. 1, pp. 61-74, 2012.

Akinyemi Bodunde.O. is a Senior Lecturer at Obafemi Awolowo University, Ile –Ife. Her current research interest include CyberSecurity, Data communication and networking. She is a Member of the Nigeria Computer Society (MNCS) and a chartered IT practitioner (MCPN).

Aladesanmi Temitope.A. is an IT technical staff at Obafemi Awolowo University), Ile-Ife, Nigeria. His current research interest include cyber security and risk mitigation. He is a chartered IT practitioner (MCPN); a life member of the Nigeria Computer Society (MNCS).

Oyebade Adedoyin.I. is a Post Graduate research student of Obafemi Awolowo University, Ile –Ife. His current research interest include Data Mining, Data communication and networking.

Aderounmu Ganiyu.A. is a professor at Obafemi Awolowo University, Ile-Ife, Nigeria. He is a Full member of the Nigeria Society of Engineers (FNSE), Nigeria Computer Society (FNCS), a chartered IT practitioner (CPN) and certified Engineer (COREN).

Kamagate Beman.H. is a Lecturer and researcher at ESATIC (Ecole Supérieure Africaine des Technologies de l'Information et de la Communication), and LARIT (Laboratoire de Recherche en Informatique et Télécommunication), Abidjan, Côte d'Ivoire.