SIMATS ENGINEERING
Engineer to Excel

SAVEETHA TRANSDISCIPLINARY ANNUAL RESEARCH SUMMIT STAR SUMMIT 2024 SIMATS ENGINEERING

TECH STAR SUMMIT 2024

Name: Mr. Yalaka Nikhil Reddy
Register Number: 192211188
Guided by E.K.Subramanian

Machine Learning based Software Fault Prediction in Software Efficiency using Artificial Neural Network Comparing with Naive Bayes for Improved Accuracy

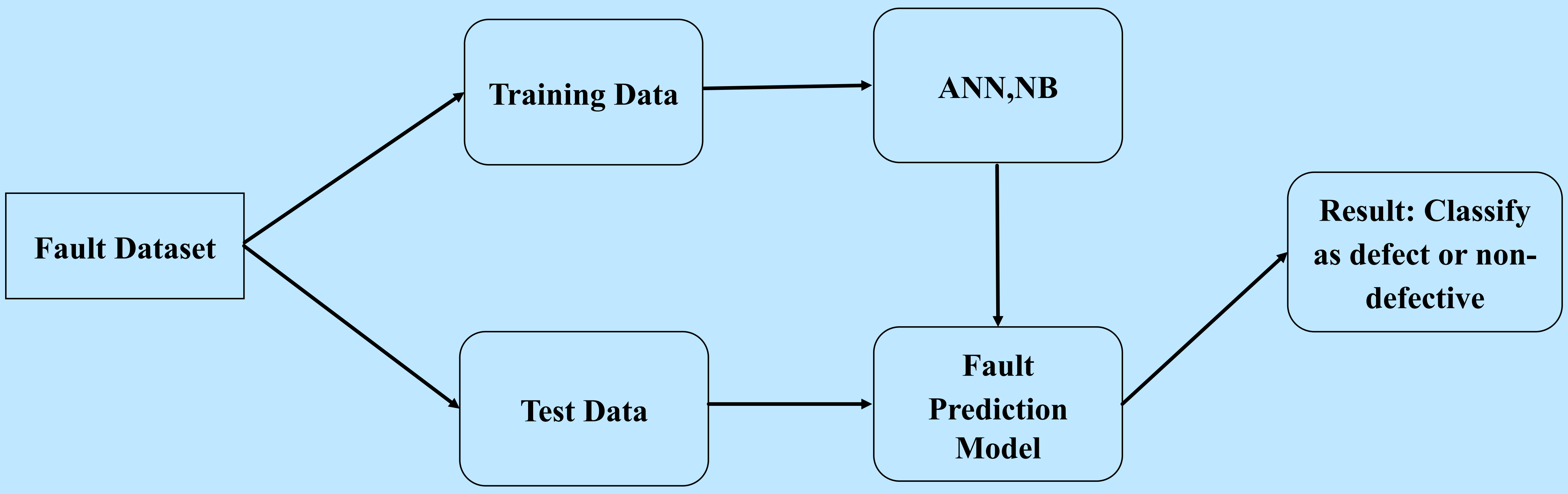
INTRODUCTION

- The aim of this study is to develop a machine learning-based software fault prediction system using Artificial Neural Network (ANN) and compare its performance with Naive Bayes to enhance accuracy in assessing software efficiency.
- It is crucial for ensuring the reliability and efficiency of software systems. By accurately predicting faults, developers can preemptively address potential issues, leading to improved software performance, reduced maintenance costs.
- This research has broad applications across various industries reliant on software systems, including but not limited to finance, healthcare, telecommunications, and e-commerce. It can benefit software development teams, quality assurance professionals, and project managers striving to deliver robust and dependable software solutions.
- Mimics the human brain's structure and functioning. Suitable for handling complex, nonlinear relationships within data . Allows for learning from patterns and making predictions based on
- A probabilistic classifier based on Bayes' theorem. Assumes independence among features, making it computationally efficient . Often used as a baseline model in classification tasks.
- Software fault prediction dataset in kaggle includes metrics like code lines, complexity, defect density, enabling robust model training.

Software fault prediction

MATERIALS AND METHODS

- The programming lab of the Saveetha Institute of Medical and Technical Sciences was the site of this investigation. There are two groups. There are two versions of software fault prediction; one is ANN and the other is NB. The sample size was determined using Clinal. Test statistic with a 0.05 threshold, 80% power, and 95% confidence interval the optimal sample size.
- Intel Core i7 processor, 512 GB of RAM, Google Collab, Jupiter Notebooks, and Windows 11th generation. In order to determine accuracy, we conducted statistical analyses using IBM SPSS.



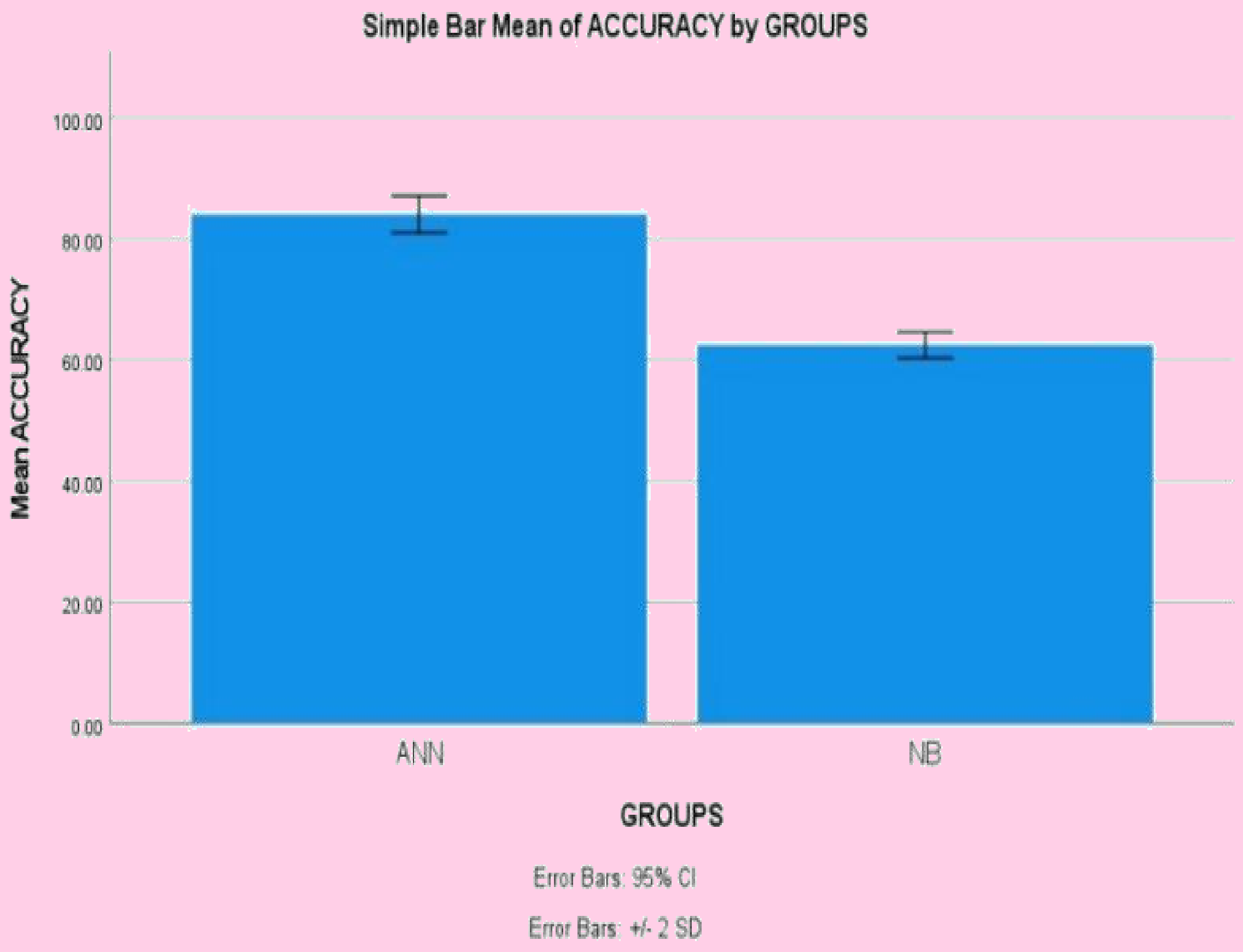
Software Fault Prediction

RESULTS

- Algorithm for software fault prediction using Artificial Neural Networks The T-tests for independent samples The following table shows the difference in significance, mean, and standard error between the artificial neural networks and Naive Bayes approaches, as determined by SPSS analysis (p = 0.001) (p<0.05).
- The results of the ANN and NB algorithms were input into SPSS in order to find the average, dispersion, and error for the two networks. ANN attained 84.10% accuracy, with Naive Bayes and 62.50% also reaching their goals.
- Using a 95% confidence interval and a standard deviation of +/- 2 as parameters, SPSS calculated an accuracy of 84.10% for the suggested method ANN and 62.50% for the NB.

Statistical Analysis of the difference in significance, mean, and standard error between the Artificial Neural Network and Naive Bayes approaches

	Algorithm	N	Mean	Std. Deviation	Std. Error Mean
Accuracy	ANN	10	84.10	1.52388	0.48189
	NB	10	62.50	1.08012	0.34157



Shows the comparison with Artificial Neural Network and Naïve Bayes with accuracy 84.10% and 62.50%

DISCUSSION AND CONCLUSION

- Through T-test Statistical analysis, a significance value of p=0.001 was obtained, indicating a significant difference between the performance of Artificial Neural Network (ANN) and Naive Bayes in software fault prediction.
- The ANN model demonstrates superior accuracy over Naive Bayes, with accuracy details revealing a clear advantage ANN: 84.10%, Naive Bayes: 62.50%.
- This suggests that ANN is more effective in predicting software faults compared to Naive Bayes.
- The study concludes that ANN outperforms Naive Bayes in software fault prediction, contributing to improved accuracy and reliability in assessing software efficiency.
- These findings are crucial for software development teams and quality assurance professionals, enabling them to preemptively address potential issues and enhance software performance.
- Future research could explore the integration of ensemble learning techniques or advanced deep learning architectures to further enhance the accuracy of software fault prediction.

BIBLIOGRAPHY

- Rathore, S.S., Kumar, S. An empirical study of ensemble techniques for software fault prediction(2021).https://link.springer.com/article/10.1007/s10489-020-01935-6
- Mehta, Sweta, and K. Sridhar Patnaik. 2021. “Improved Prediction of Software Defects Using Ensemble Machine Learning Techniques.” Neural Computing & Applications 33 (16): 10551–62.https://doi.org/10.1007/s00521-021-05811-3
- Assim, Marwa, Qasem Obeidat, and Mustafa Hammad. n.d. “Software Defects Prediction Using Machine Learning Algorithms.” Accessed March 18, 2024.http://dx.doi.org/10.1109/ICDABI51230.2020.9325677
- Bahaweres, Rizal Broer, Fajar Agustian, Irman Hermadi, Arif Imam Suroso, and Yandra Arkeman. n.d. “Software Defect Prediction Using Neural Network Based SMOTE.” Accessed March 18, 2024.http://dx.doi.org/10.23919/EECSI50503.2020.9251874
- Cetiner, Murat, and Ozgur Koray Sahingoz. n.d. “A Comparative Analysis for Machine Learning Based Software Defect Prediction Systems.” Accessed March 18, 2024. http://dx.doi.org/10.1109/ICCCNT49239.2020.9225352