ISSN 2319-5991 www.ijerst.com

Peer Reviewed, Referred & Indexed Journal

Research Paper

AI-Based Decision Support System for Startup Investment Analysis

[1]Anupoju Bhuvan,[2]Kurimilli Harsha Sri Sai, [3]Thota Srinivas Swamy,[4]Polisetti Subhakar,
[5]Mr. K.S.R. Prasad
[1,2,3,4]U. G Student, Department of CSE (Artificial Intelligence & Machine Learning),
D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY (AUTONOMOUS)
Balusumudi, Bhimavaram, West Godavari District, Andhra Pradesh -534202
[5]Assistant Professor, Department of CSE (Artificial Intelligence & Machine Learning),
D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY (AUTONOMOUS)
Balusumudi, Bhimavaram, West Godavari District, Andhra Pradesh -534202

ABSTRACT

Software startups businesses create inventive, programming escalated items or administrations. Such imaginativeness converts into vulnerability in regards to a matching requirement for an item from expected clients, addressing a potential determinant justification for startup disappointment. Research has shown that trial and error, a methodology in light of the utilization of tests to direct a few parts of programming advancement, could further develop these organizations' prosperity rate by encouraging the assessment of presumptions about clients' necessities previously fostering an undeniable item. By and by, programming new companies are not involving trial and error true to form. In this review, we researched the explanations for such a jumble among hypothesis and practice. To accomplish it, we played out a subjective review study of 106 failed software startups. We built the eXperimentation Progression model (XPro), demonstrating that the effective adoption and implementation of experimentation is a staged process: first, groups ought to know about trial and error, then they need to create an aim to explore, play out the examinations, break down the outcomes, lastly act in view of the acquired learning. In light of the XPro model, we further distinguished 25 inhibitors that keep a group from advancing along the stages appropriately.

Our discoveries illuminate analysts of how to foster practices and strategies to further develop trial and error reception in programming new businesses. Professionals could learn different variables that could prompt their startup disappointment so they could make a move to stay away from them.

KEYWORDS: - XGBOOST, Support Vector Machines (SVM), Gradient Boost, Ada-boost, Random Forest, Machine Learning

INTRODUCTION

The most recent 20 years, creative software-intensive prod- ucts changed many aspects of our lives. For example, these days, individuals move around in numerous urban areas around the world utilizing ride-sharing applications like Uber or Lyft instead of taxi administrations. We are becoming acclimated to booking facilities for our get-aways or excursions for work through Airbnb rather than on lodging sites. Our everyday specialized devices are Whatsapp, Twitter, and Zoom rather than calls or on the other hand even messages. These product items, disturbing long-standing customary businesses, are for the most part made by new what's more, arising organizations, the purported programming new companies. These associations foster inventive programming concentrated items or administrations and quest for repeatable and versatile plans of action. This journey for a practical plan of action alongside the risk of originality prompts higher uncer- tainties than in laid out organizations . It is fundamental to underline that the startup is a transitory stage prompting a merged organization or the exercises' end. Regardless of numerous effective cases as those referenced above, over 90% of programming new companies neglect to become solid and productive organizations . There are different expected explanations behind fall flat- ure, including requesting economic situations, absence of group responsibility, and monetary issues however off-base business advancement is one key determinant.

RELATED WORK

Recent research on software startup success has emphasized the role of data-driven decision-making and experimentation in reducing failure rates. Studies highlight that many startups fail due to a lack of validated learning and improper understanding of customer needs. Machine learning techniques such as XGBoost, Support Vector Machines, Random Forest, and Gradient Boosting have been applied to predict startup success and failure patterns. Researchers have analyzed factors like funding, team composition, market conditions, and product innovation using predictive models. Experimental approaches such as A/B testing and lean startup methodologies have been widely studied to validate assumptions before product development. Some works propose frameworks that integrate continuous experimentation into the software development lifecycle. Data analytics and feature engineering have been used to identify key indicators influencing startup growth. Comparative studies show ensemble learning methods provide higher accuracy in predicting outcomes. Additionally, case studies of failed startups reveal common inhibitors such as poor strategy and lack of market fit. Overall, existing research demonstrates that combining experimentation with machine learning can significantly improve decision-making and increase the chances of startup success.

LITERATURE SURVEY

Startup success prediction has gained significant attention with the advancement of machine learning and data-driven analytics. Early research applied traditional algorithms such as decision trees and logistic regression to analyze business features influencing startup performance. Later studies introduced ensemble techniques like Random Forest to improve prediction accuracy by handling complex and nonlinear relationships within startup datasets. Researchers also integrated sentiment analysis methods to evaluate public opinion and market perception using textual data, enhancing predictive capability. Support Vector Machines and gradient boosting models demonstrated strong classification performance in identifying successful startups. Data-driven approaches emphasized the importance of high-quality datasets, feature engineering, and preprocessing techniques for reliable outcomes. Neural networks were further explored to capture hidden patterns and interactions among financial, operational, and market variables. Comparative studies evaluated multiple machine learning models to understand their strengths, limitations, and suitability for different datasets. Despite promising results, challenges such as data imbalance, limited historical records, and model generalization remain significant concerns. Overall, existing literature confirms that machine learning techniques provide effective tools for predicting startup success and supporting investment decision-making processes.

EXISTING METHOD

The existing startup investment analysis systems mainly rely on traditional machine learning algorithms such as Linear Regression, basic Decision Tree models, and simple statistical classifiers for predicting startup success or failure based on limited historical data. These systems focus primarily on financial indicators and past trends, providing only basic prediction results without considering multiple performance factors. Most existing approaches lack an integrated decision support platform and do not offer real-time prediction through web-based applications. Visualization and reporting are mostly static, making it difficult for investors to interpret complex outcomes. Additionally, the prediction accuracy of these systems is limited due to the use of single or less robust algorithms and the absence of ensemble learning techniques. There is no unified framework to combine multiple machine learning models for comparative analysis. As a result, existing systems are not efficient in providing reliable, scalable, and user-friendly startup success predictions for investment decision-making.

PROPOSED METHOD

The proposed system implements a Flask-based web application to predict startup success rates using multiple machine learning algorithms. It integrates models such as Logistic Regression, Decision Tree Classifier, Random Forest Classifier, Gaussian Naive Bayes, Gradient Boosting, and AdaBoost to ensure accurate and robust predictions. The application collects startup-related data including funding, market size, team strength, and business model characteristics. Data preprocessing techniques such as normalization, encoding, and feature selection are applied to improve model performance. Each algorithm is trained and evaluated to compare prediction accuracy and reliability. Ensemble methods like Random Forest and Gradient Boosting enhance overall prediction stability. The system provides real-time predictions through a user-friendly web interface built using Flask. Visualization tools are included to display insights and success probability scores. The model helps entrepreneurs make informed decisions by identifying key success factors. Overall, the proposed approach combines machine learning and web technology to deliver an intelligent and practical startup success prediction system.

SYSTEM ARCHITECTURE

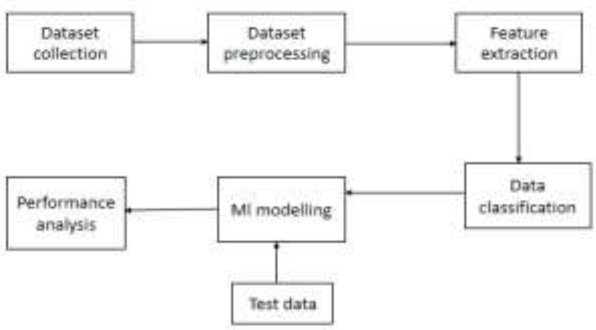

Figure 1: Architecture of the Project

METHODOLOGY DESCRIPTION

Dataset Collection: This module allows users to upload datasets through an HTML form interface. Flask handles file uploads (CSV, images, etc.) and stores them on the server. It validates file formats and ensures proper data intake. The uploaded data becomes the input for further processing. This module acts as the entry point of the system.

Data Preprocessing Module: In this module, Flask processes the uploaded dataset by cleaning and transforming it. It removes missing values, normalizes data, and encodes categorical variables. The processed data is stored temporarily or in a database. HTML pages display preprocessing status and summaries. This step ensures the dataset is ready for model training.

Feature Extraction Module: This module extracts important features from the processed dataset using Python libraries. Flask executes feature engineering logic such as selecting key attributes or applying transformations. The results are displayed using HTML tables or charts. It reduces unnecessary data and improves model efficiency. This module prepares optimized input for classification.

ML Model Module (Training & Classification): Flask integrates machine learning models like Random Forest, SVM, or Logistic Regression. Users can trigger model training through the HTML interface. The module splits data into training and testing sets automatically. Predictions are generated and displayed to users. This is the core module where learning and classification happen.

Performance Analysis Module: This module evaluates model performance using metrics like accuracy, precision, recall, and F1-score. Flask computes results and sends them to HTML templates for visualization. Graphs like confusion matrix or charts can be displayed. It helps users understand model effectiveness. Feedback from this module can be used to improve earlier stages.

RESULTS AND DISCUSSION

This project shows the details of profile how we can detect easily.

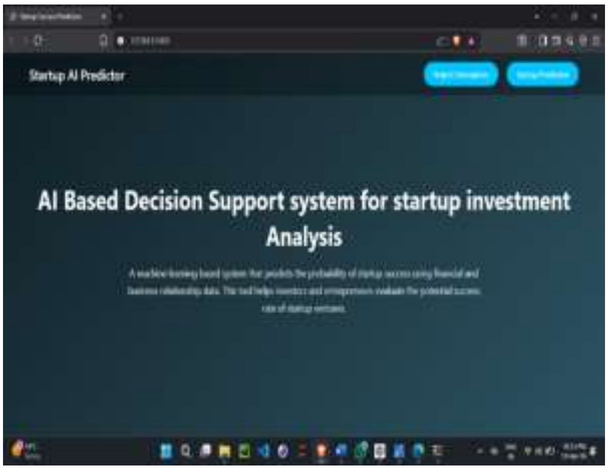

Figure 2.1: Home Page

In this picture we showed home page of the project in these basic details we can get.

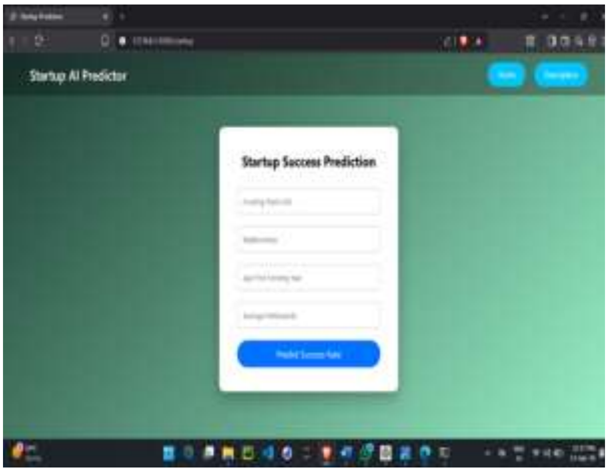

Figure 2.2: User Input Page

In the above shown Figure, the user needs to fill input details.

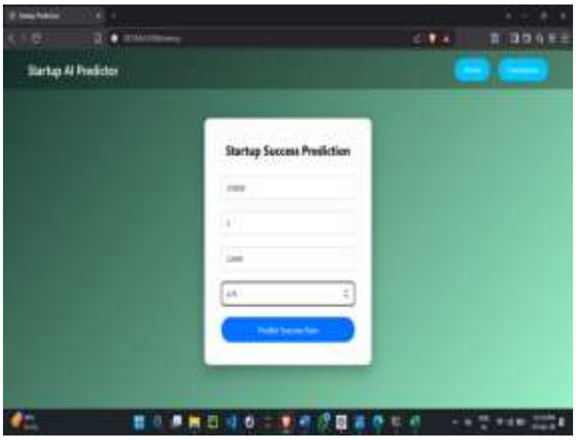

Figure 2.3: Uploading input data page

In the above shown figure, the user filled inputs details for startup success Prediction.

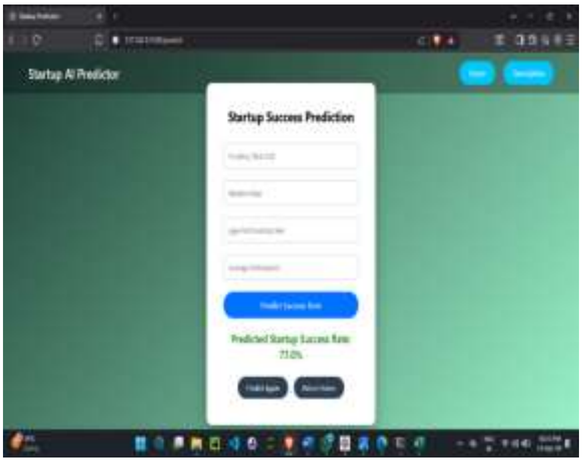

Figure 2.4: Prediction Page

The model predicts a startup success rate of 77.0% based on the provided inputs.

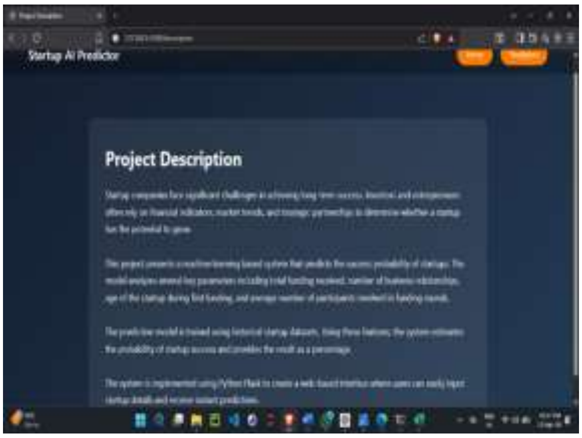

Figure 2.5: Project Description Page

This project uses a machine learning model to predict the success probability of startups based on key financial and relational factors.

CONCLSUION

The proposed XPro-based system effectively demonstrates the importance of structured experimentation in improving software startup success rates. By guiding teams through stages such as awareness, experimentation, analysis, and action, it reduces uncertainty in product development. The integration of machine learning and systematic workflows enables data-driven decision-making. The system also identifies key inhibitors that hinder experimentation adoption, helping startups avoid common pitfalls. Overall, the approach enhances innovation, improves product-market fit, and increases the likelihood of long-term startup success.

FUTURE SCOPE

Future enhancements can include integrating AI-driven analytics to automate hypothesis generation and experiment recommendations. The system can be extended with cloud-based and mobile platforms for better accessibility and scalability. Incorporating advanced visualization dashboards and real-time monitoring will improve decision-making efficiency. Integration with tools like Jira and Google Analytics can streamline workflows and collaboration. Expanding the model with adaptive learning and predictive capabilities will make it a comprehensive intelligent startup support system.

REFERENCES

[1] S. Blank, "Why the Lean Start-Up Changes Everything," *Harvard Business Review*, vol. 91, no. 5, pp. 63–72, 2013.

[2] E. Ries, *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*. New York, NY, USA: Crown Business, 2011.

[3] B. Cooper and P. Vlaskovits, *The Entrepreneur's Guide to Customer Development*. BookBaby, 2010.

[4] T. R. Eisenmann, E. Ries, and S. Dillard, "Hypothesis-Driven Entrepreneurship: The Lean Startup," *Harvard Business School*, 2012.

[5] A. Maurya, *Running Lean: Iterate from Plan A to a Plan That Works*. O'Reilly Media, 2012.

[6] N. Furr and J. Dyer, *The Innovator's Method*. Harvard Business Review Press, 2014.

[7] Y. Wang and P. K. Ahmed, "Organizational innovativeness," *European Journal of Innovation Management*, vol. 7, no. 4, pp. 303–313, 2004.

[8] R. G. McGrath, "Business models: A discovery-driven approach," *Long Range Planning*, vol. 43, pp. 247–261, 2010.

[9] A. Ghezzi and A. Cavallo, "Agile business model innovation," *Journal of Business Research*, vol. 94, pp. 193–206, 2018.

[10] S. Blank, *The Four Steps to the Epiphany*. K&S Ranch Press, 2007.

[11] E. Ries, *The Startup Way*. Currency, 2017.

[12] N. Furr and D. Snow, *Nail It Then Scale It*. NISI Institute Press, 2015.

[13] S. Thomke, *Experimentation Matters*. Harvard Business School Press, 2003.

[14] A. C. Klotz and D. O. Neubaum, "The dark side of entrepreneurship," *Academy of Management Annals*, 2016.

[15] K. M. Eisenhardt, "Building theories from case study research," *Academy of Management Review*, 1989.

[16] S. D. Sarasvathy, "Causation and effectuation," *Academy of Management Review*, 2001.

[17] J. Gans, E. Scott, and S. Stern, "Strategy for startups," *Harvard Business Review*, 2018.

[18] S. Blank and B. Dorf, *The Startup Owner's Manual*. K&S Ranch Press, 2012.

[19] R. G. Cooper, "Stage-Gate process," *Journal of Product Innovation Management*, 2008.

[20] E. Ries, "Continuous innovation for entrepreneurs," *Harvard Business School Press*, 2009.

[21] A. Bhide, *The Origin and Evolution of New Businesses*. Oxford University Press, 2000.

[22] C. M. Christensen and M. Raynor, *The Innovator's Solution*. Harvard Business School Press, 2003.

[23] McKinsey & Company, "The State of AI," McKinsey Insights, 2016.

[24] S. Blank, "Why startups fail," *Harvard Business Review Blog*, 2010.

[25] A. Ghezzi, "Digital entrepreneurship," *Technological Forecasting and Social Change*, 2019.

[26] A. Maurya, *Scaling Lean*. Portfolio, 2016.

[27] KPMG, "Startup ecosystem report," KPMG Insights, 2017.

[28] S. Thomke and E. Von Hippel, "Customers as innovators," *Harvard Business Review*, 2002.

[29] J. Gans, E. Scott, and S. Stern, "Entrepreneurship in technology startups," *Strategic Entrepreneurship Journal*, 2018.

[30] A. Croll and B. Yoskovitz, *Lean Analytics: Use Data to Build a Better Startup Faster*. O'Reilly Media, 2015.