Development of Risk Prediction Model for Cervical Cancer

A PROJECT REPORT

Submitted by

AMIT KUMAR [RA2211003011640]
KASHISH SINGH [RA2211003011798]

Under the Guidance of

Dr. J KALAIVANI
Assistant Professor
Department of Computing Technologies

in partial fulfilment of the requirements for the degree of

BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE ENGINEERING

DEPARTMENT OF COMPUTING TECHNOLOGIES
SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR- 603203

MAY 2025

Department of Computing Technologies
SRM Institute of Science & Technology
Own Work* Declaration Form

This sheet must be filled in (each box ticked to show that the condition has been met). It must be signed and dated along with your student registration number and included with all assignments you submit – work will not be marked unless this is done.

To be completed by the student for all assessments

Degree/ Course	:	Bachelor of Technology
Student Name	:	Amit Kumar, Kashish Singh
Registration Number	:	RA2211003011640, RA2211003011798
Title of Work	:	Development of Risk Prediction Model for Cervical Cancer

I / We hereby certify that this assessment compiles with the University's Rules and Regulations relating to Academic misconduct and plagiarism**, as listed in the University Website, Regulations, and the Education Committee guidelines.

I / We confirm that all the work contained in this assessment is my / our own except where indicated, and that I / We have met the following conditions:

- Clearly referenced / listed all sources as appropriate
- Referenced and put in inverted commas all quoted text (from books, web, etc)
- Given the sources of all pictures, data etc. that are not my own
- Not made any use of the report(s) or essay(s) of any other student(s) either past or present
- Acknowledged in appropriate places any help that I have received from others (e.g. fellow students, technicians, statisticians, external sources)
- Compiled with any other plagiarism criteria specified in the Course handbook / University website

I understand that any false claim for this work will be penalised in accordance with the University policies and regulations.

DECLARATION:

I am aware of and understand the University's policy on Academic misconduct and plagiarism and I certify that this assessment is my / our own work, except where indicated by referring, and that I have followed the good academic practices noted above.

RA2211003011640 RA2211003011798

If you are working in a group, please write your registration numbers and sign with the date forevery student in your group.

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR – 603 203

BONAFIDE CERTIFICATE

Certified that 21CSP302L - Project report titled “**DEVELOPMENT OF RISK PREDICTION MODEL FOR CERVICAL CANCER**” is the bonafide work of “**AMIT KUMAR [RA2211003011640], KASHISH SINGH [RA2211003011798]**” who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE	**SIGNATURE**
Dr. J Kalaivani	**Dr. G. Niranjana**
Assistant Professor	**Professor and Head**
DEPARTMENT OF	DEPARTMENT OF
COMPUTING TECHNOLOGIES	COMPUTING TECHNOLOGIES

EXAMINER 1	**EXAMINER 2**

ACKNOWLEDGEMENTS

We express our humble gratitude to **Dr. C. Muthamizhchelvan**, Vice-Chancellor, SRM Institute of Science and Technology, for the facilities extended for the project work and his continued support.

We extend our sincere thanks to **Dr. Leenus Jesu Martin M,** Dean-CET, SRM Institute of Science and Technology, for his invaluable support.

We wish to thank **Dr. Revathi Venkataraman**, Professor and Chairperson, School of Computing, SRM Institute of Science and Technology, for her support throughout the project work.

We encompass our sincere thanks to, **Dr. M. Pushpalatha**, Professor and Associate Chairperson - CS, School of Computing and **Dr. Lakshmi,** Professor and Associate Chairperson -AI, School of Computing, SRM Institute of Science and Technology, for their invaluable support.

We are incredibly grateful to our Head of the Department, **Dr. G. Niranjana**, Professor and Head , Department of Computing Technologies, SRM Institute of Science and Technology, for her suggestions and encouragement at all the stages of the project work.

We want to convey our thanks to our Project Coordinators, Panel Head, and Panel Members Department of Computational Intelligence, SRM Institute of Science and Technology, for their inputs during the project reviews and support.

We register our immeasurable thanks to our Faculty Advisor, **Dr. R. Thilagavathy**, Department of Computing Technologies, SRM Institute of Science and Technology, for leading and helping us to complete our course.

Our inexpressible respect and thanks to our guide, **Dr. J Kalaivani,** Department of Computing Technologies, SRM Institute of Science and Technology, for providing us with an opportunity to pursue our project under her mentorship. She provided us with freedom and support to explore the research topics of our interest. Her passion for solving problems and making a difference in the world has always been inspiring.

We sincerely thank all the staff members of Department of Computing Technologies, School of Computing, S.R.M Institute of Science and Technology, for their help during our project. Finally, we would like to thank our parents, family members, and friends for their unconditional love, constant support and encouragement.

Amit Kumar RA2211003011640
Kashish Singh RA2211003011798

ABSTRACT

Cervical cancer continues to be a pressing health concern globally, largely due to late detection and frequent diagnostic errors. To help address these issues, this project introduces a user-friendly web-based application that uses machine learning to support early and more accurate identification of cervical cancer risk. The system brings together several key components—data preprocessing, intelligent feature selection, and high-performance classification models—to assist healthcare professionals in making more confident and timely diagnoses.

At the core of this platform is a complete machine learning pipeline that begins with collecting and cleaning real-world patient data. Special care is taken to maintain data quality and consistency. To improve the efficiency of the learning models, we apply techniques like Principal Component Analysis (PCA) and Recursive Feature Elimination (RFE) to reduce unnecessary variables while keeping important predictive features. Additionally, normalisation and standardisation are used to enhance model performance, and class imbalances are addressed using Synthetic Minority Oversampling Technique (SMOTE), which improves the system's ability to detect high-risk cases.

The application tests a variety of machine learning models—including Logistic Regression, XGBoost, AdaBoost, and a Stacking Classifier that combines multiple models—to identify the one with the best predictive performance. Each model undergoes thorough hyper-parameter tuning using Grid Search and Randomised Search techniques, supported by cross-validation to ensure stability and avoid overfitting. The system also incorporates explainability tools like SHAP (SHapley Additive exPlanations), which help interpret how each feature influences the final prediction.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

ML	–	Machine Learning
EHR	–	Electronic Health Record
ET	–	Extra Trees
SVM	–	Support Vector Machine
ETSE.	–	Extra Trees and Support Vector MachineEnsemble
SMOTE	–	Synthetic Minority Over-sampling Technique
MVC	–	Model-View-Controller
SDG	–	Sustainable Development Goals
ROC	–	Receiver OperatingCharacteristic
AUC	–	Area Under Curve
PR	–	Precision-Recall
SHAP	**–**	SHapley Additive exPlanations

CHAPTER 1

INTRODUCTION

Cervical cancer is a serious health concern affecting women worldwide, especially in regions with limited access to regular health screenings. Despite advances in medicine, the disease often goes undetected until it reaches a critical stage due to delays in diagnosis and insufficient medical infrastructure. This highlights the need for alternative methods that can support early detection and encourage timely medical intervention.

The goal of this project is to create a system that can predict a woman’s risk of developing cervical cancer using machine learning techniques. Rather than relying on traditional tests that require lab equipment and medical personnel, this system uses simple clinical and lifestyle data as input. By analyzing patterns within this data, the model can provide a reliable indication of whether someone is at high or low risk.

This tool is especially valuable in remote or underserved areas, where access to gynecological screening may be rare. With just a few health-related inputs, the system can help identify individuals who need further examination, allowing healthcare workers to prioritize their resources. It is not meant to replace doctors but to assist them by making preliminary screening faster and more accessible.

The project includes the development of a user-friendly web application that allows users to input numerical data and receive a prediction instantly. It integrates multiple machine learning models and compares their performance to ensure the most accurate and trustworthy results. The system is designed to be lightweight, adaptable, and usable even on basic hardware, making it suitable for both clinical and non-clinical environments.

Overall, this project aims to bridge the gap between technology and preventive healthcare by offering an intelligent, affordable, and scalable solution to a global health issue. By supporting early diagnosis, it contributes to saving lives and raising awareness about the importance of cervical cancer screening, particularly among women in high-risk and low-resource settings.

1. Introduction to Project

Cervical cancer is one of the most common cancers affecting women, particularly in low- and middle-income countries. Despite being largely preventable and treatable when detected early, it remains a leading cause of cancer-related deaths among women worldwide. Traditional screening methods, such as Pap smears and HPV testing, are highly effective but often underutilized due to limited accessibility, high costs, and the need for clinical infrastructure and expert interpretation.

In recent years, advances in artificial intelligence and data-driven healthcare have opened up new possibilities for early disease detection. Machine learning, in particular, has emerged as a powerful tool in the medical field, offering the ability to analyze vast amounts of health-related data and identify patterns that may not be immediately evident to human observers. This project leverages these capabilities to build an intelligent system that predicts the risk of cervical cancer based on easily accessible patient information.

This disease, which originates in the cervix, is most often caused by long-term infection with high-risk types of Human Papillomavirus (HPV). Despite the fact that cervical cancer is preventable and treatable when detected early, many cases continue to be diagnosed at later stages—often due to lack of awareness, limited access to screening, and inadequate diagnostic infrastructure.

Traditional screening techniques, such as Pap smears and HPV DNA testing, have greatly contributed to early diagnosis and improved survival rates. However, these methods require skilled professionals, lab facilities, and periodic follow-up, which are not always available in rural or resource-constrained environments. In addition, delays in diagnosis due to human error or the subjective nature of result interpretation can lead to missed opportunities for early treatment.

In this context, machine learning offers a promising alternative. By analyzing structured medical data and identifying patterns, ML models can assist in assessing risk and prioritizing patients for further examination. This approach has the potential to support early intervention, reduce the burden on the healthcare system, and ultimately save lives.

2. Problem Statement

Cervical cancer continues to pose a serious threat to women's health globally, particularly in areas where access to timely screening and diagnostic services is limited. Despite being preventable and treatable when caught early, many cases go undiagnosed until they reach an advanced stage. This leads to reduced survival rates and increased treatment complexity. The root of the problem lies not only in medical limitations but also in systemic challenges like lack of awareness, limited infrastructure, and inconsistent follow-up care.

In many developing and underserved regions, there are not enough trained healthcare professionals or testing centres equipped to perform regular cervical screenings such as Pap smears or HPV tests. Even when these services are available, logistical challenges—like transport, cost, and scheduling—often prevent women from accessing them regularly. The result is that thousands of women remain undiagnosed until symptoms become severe, and by then, treatment options are fewer and less effective.

Moreover, cervical cancer screening methods are often reliant on manual interpretation, which introduces the possibility of human error. Medical professionals may misread results due to workload, fatigue, or subtle abnormalities that are not easily visible. Inconsistencies in sample collection, staining, and interpretation further compound the risk of misdiagnosis or delayed diagnosis, undermining the effectiveness of existing screening programs.

Another dimension of the problem lies in health inequality. Women from lower socio-economic backgrounds are disproportionately affected because they may not have access to routine healthcare. Cultural and social stigmas around reproductive health also contribute to low participation in screening programs. Thus, a technology-driven, non-invasive, and accessible risk assessment tool could fill this gap by offering a fast, private, and low-cost way to flag potential high-risk cases.

Current software and hospital systems are often fragmented and lack intelligent features that support real-time decision-making or early intervention. There's a critical need for a predictive model that can work with basic health data and still produce accurate, meaningful outputs. Such a tool could act as a first line of assessment—highlighting patients who should be prioritized for testing or further consultation.

While many machine learning applications exist in healthcare, few focus specifically on cervical cancer risk prediction using accessible data inputs. Most are either limited to image classification or depend heavily on expensive diagnostic tests. This project addresses that gap by using structured clinical and personal data—something that is easier to collect and process—to make risk assessments.

3. Motivation

The driving force behind this project stems from a shared desire to use technology for social good—specifically to improve access to preventive healthcare for women. Cervical cancer is highly preventable, yet it still claims thousands of lives every year due to late diagnoses. This disconnect between what's medically possible and what's practically delivered inspired us to develop a system that can assist in bridging that gap.

We were especially motivated by stories and statistics showing that many women miss their opportunity for early treatment simply because they never get screened. This is not always due to negligence; in many cases, it's because they live in areas where screening is unavailable or unaffordable. We wanted to create a solution that could offer value even where clinical infrastructure is limited.

Machine learning, with its ability to find complex patterns in data, presents an opportunity to make health prediction tools more accessible and scalable. Our motivation was to harness this power to build something simple yet impactful—a web-based application that can predict cervical cancer risk based on health history and lifestyle data. This would allow health workers and individuals to better understand risk without requiring costly or invasive procedures.

Another key motivation was the desire to enhance the capabilities of frontline health workers. In many communities, healthcare providers work under tremendous pressure, often without specialist support. A tool that provides intelligent risk assessments can empower them to take quicker, more confident actions, improving patient care without adding to their workload.

The user-friendly nature of a web-based solution also motivated us. With smartphones and internet access becoming more common, even in rural regions, a lightweight, mobile-accessible tool could have real-world applicability. This means a woman could, in the future, use such a tool during a routine checkup or even independently to assess whether she should consult a doctor.

We were also inspired by the potential to contribute to a broader cause—advancing gender-specific healthcare and aligning with the Sustainable Development Goals. The focus on women's health, equality in access to medical technology, and the use of innovation in healthcare deeply resonated with us and added purpose to the technical goals of the project.

Finally, the project allowed us to combine academic learning with practical impact. It gave us a chance to apply our technical skills to a real-world issue in a way that could actually help people. That sense of purpose was and continues to be our biggest motivation as we work towards making this system usable, reliable, and widely accessible.

4. Sustainable Development Goal of the Project

This project contributes directly to two critical United Nations Sustainable Development Goals: **SDG 3 – Good Health and Well-being** and **SDG 5 – Gender Equality**, along with other two Sustainable Development Goals: **SDG 9 – Industry, Innovation and Infrastructure and SDG 10 – Reduced Inequalities .** All of these goals emphasise the need for inclusive, accessible, and high-quality healthcare systems, by encouraging the use of innovative technology in the public health.

SDG 3 – Good Health and Well-being

One of the main objectives of this project is to improve health outcomes through early detection of cervical cancer. By using machine learning to predict risk based on personal and clinical data, the system supports preventive healthcare, allowing women to seek timely medical attention before symptoms worsen. This aligns directly with SDG 3, which focuses on ensuring healthy lives and promoting well-being for all.

In many parts of the world, especially in under-resourced regions, access to diagnostic services is limited. Our project addresses this gap by offering an easy-to-use web platform that can provide quick, preliminary assessments of cervical cancer risk. This reduces dependency on laboratory infrastructure and helps healthcare providers prioritize patients who need further examination, effectively lowering the burden of late-stage diagnosis and mortality.

By empowering both patients and healthcare workers with timely, data-backed insights, the project contributes to reducing the impact of non-communicable diseases like cancer. It enables earlier intervention, potentially less invasive treatments, and a higher quality of life for women who might otherwise remain undiagnosed until it's too late.

SDG 5 – Gender Equality

Cervical cancer affects only women, making this project inherently aligned with the goal of improving gender-focused healthcare. Across the globe, women—especially in rural and low-income areas—often face unequal access to medical services. Our system bridges this gap by providing a scalable, accessible solution that enables early detection regardless of geography or socioeconomic status.

This tool promotes autonomy by allowing women to understand their health risks without needing expensive or time-consuming tests. It empowers them to take charge of their health, breaking down the traditional barriers of stigma and dependency. In many cultures where open conversations about reproductive health are discouraged, such a tool offers a more private, judgment-free space to initiate preventive care.

By improving access to early screening and supporting informed medical decisions, the project uplifts women and advocates for equitable healthcare access. It reflects a commitment to SDG 5 by not only addressing a life-threatening disease that affects millions of women but also promoting their right to quality healthcare and bodily autonomy.

SDG 9 – Industry, Innovation and Infrastructure

This project brings technological innovation into the heart of healthcare, showcasing how digital tools and machine learning can be used to address real-world medical challenges. SDG 9 encourages building resilient infrastructure and fostering innovation—both of which are reflected in the development of this intelligent health system.

The use of machine learning algorithms in risk prediction is a step forward in modernizing diagnostics. By integrating tools like XGBoost and ensemble models, along with advanced data preprocessing and explainability techniques, the system provides not just accuracy but also transparency—something that is essential in building trust around AI in healthcare.

Additionally, the web-based nature of the tool means it can be deployed with minimal infrastructure. All that's needed is a device with internet access. This allows healthcare institutions and even individual practitioners in developing areas to implement modern diagnostics without expensive hardware or lab setups, effectively promoting innovation at a grassroots level.

SDG 10 – Reduced Inequalities

One of the strongest impacts of this project is its potential to reduce inequalities in healthcare access. SDG 10 focuses on reducing disparities within and among countries, and our project directly contributes to this by providing a tool that can be used across different regions, regardless of available resources.

Traditionally, advanced diagnostic tools are only available in well-funded hospitals in urban areas. However, the machine learning model developed here can function with minimal data and infrastructure, offering meaningful predictions to those in underserved locations. This democratizes access to health assessments and helps close the gap between rural and urban healthcare services.

Furthermore, by prioritizing accessibility and affordability, the system ensures that women from all backgrounds—not just those who can afford hospital visits or private screening—can receive early warnings. This kind of inclusive design supports a more equitable healthcare landscape where every woman, no matter her circumstances, has a fair chance at early detection and better outcomes.

CHAPTER 2
LITERATURE SURVEY

1. Overview of the Research Area

Over the last decade, there has been a noticeable increase in the use of machine learning in healthcare, particularly for early detection and risk prediction of diseases. Several researchers have explored the use of data-driven models to support the diagnosis of cervical cancer by analysing patient health records and clinical features. These models help identify patterns and correlations that may not be immediately visible through traditional methods, thus assisting in quicker and more informed decision-making.

One common direction in existing studies involves using supervised machine learning models to classify cervical cancer risk levels. Algorithms such as Logistic Regression, Support Vector Machines (SVM), and Decision Trees have been widely tested on clinical datasets to assess their ability to accurately distinguish between high-risk and low-risk patients. These models often rely on patient demographics, reproductive history, and lifestyle factors such as smoking or contraceptive use to make predictions.

In more recent developments, ensemble models like Random Forest and boosting techniques (e.g., AdaBoost and XGBoost) have been adopted to improve prediction accuracy. These approaches combine multiple weak learners to produce stronger results and have proven effective in handling complex medical datasets. Some researchers have even explored deep learning models such as neural networks, though these require larger datasets and more computational resources, which can limit their use in small-scale applications.

Data preprocessing has been identified as a crucial step in model development. Several research papers emphasise the importance of cleaning missing values, scaling inputs, and balancing the dataset using techniques like SMOTE (Synthetic Minority Oversampling Technique). These preprocessing steps significantly affect model performance, especially in healthcare datasets that tend to be imbalanced due to a lower number of positive cases.

Despite the growing body of research, most existing systems still lack a user-friendly interface that can be easily used by non-technical healthcare staff. Many models are tested in academic environments but are not deployed in real-world settings due to usability or integration issues. Moreover, explainability is often overlooked, making it harder for medical practitioners to trust and interpret the model’s predictions.

What sets our project apart is its focus on practicality and accessibility. While building on the foundations of previous work, we aim to create a complete solution—one that not only predicts cervical cancer risk using well-established algorithms but also delivers the results in a clear, interpretable format. By incorporating explainability tools and an interactive web interface, we seek to bridge the gap between theoretical research and real-world clinical use.

This project, therefore, builds upon a well-researched field but brings together various components—preprocessing, model training, evaluation, and user interface—into one integrated system. Our approach emphasises usability, accuracy, and accessibility, making it a meaningful step forward in AI-assisted cancer prediction.

2. Existing Models And Frameworks

Traditionally, cervical cancer detection has relied on screening techniques such as Pap smears and HPV tests. While these methods have played an important role in identifying at-risk individuals, they require clinical infrastructure, trained personnel, and multiple patient visits, making them less accessible in rural or underserved regions. Moreover, interpretation of test results is often subject to human variability, which can lead to inconsistent diagnoses. These factors, combined with the need for regular follow-ups, make traditional screening processes less effective in areas where early intervention could be life-saving.

In many parts of the world, particularly where healthcare resources are stretched thin, these manual methods fall short in reaching the population in time. Delays in analysis, high costs, and the lack of awareness often result in late-stage diagnoses, when treatment is more difficult and less effective. Even some of the early automated systems developed to assist in diagnosis have shown limited capabilities, as they were built on basic statistical models or traditional rule-based logic that does not adequately capture the diversity in patient profiles.

With the advancement of data-driven technologies, a more promising approach has emerged. Modern predictive systems can now analyse health data using intelligent algorithms that detect hidden patterns and subtle risk indicators. By employing ensemble learning techniques—such as XGBoost, AdaBoost, and Stacking Classifiers—these systems can combine the decision-making power of multiple algorithms. This allows for improved precision and reduced error rates, especially when dealing with complex medical cases or patients presenting with unusual symptoms.

What makes these models particularly valuable is their ability to learn from diverse datasets and improve over time as more information becomes available. Unlike single-model approaches, ensemble models integrate the strengths of several algorithms, reducing bias and increasing generalization. This is critical in medical applications where even a small mistake can significantly affect a patient's outcome. In this context, improving sensitivity and specificity through such integrated models is essential for enhancing the reliability of early cervical cancer risk detection.

Beyond accuracy, accessibility is a growing focus in the development of diagnostic tools. The goal is not only to improve prediction but to make these tools widely usable, even by healthcare workers with limited technical training. By embedding these models into web applications, they become easy to navigate and deploy. With just a basic internet connection, healthcare providers in both urban hospitals and rural clinics can benefit from intelligent, real-time assessments, without needing specialized diagnostic machines or software.

This web-based delivery also supports scalability and future integration into broader healthcare systems. A modular framework allows the system to be adapted for various clinical environments, making it a practical solution for governments and health organizations looking to improve preventive care services. Its lightweight design means it can be maintained and updated with ease, ensuring that the tool remains relevant as new data or medical standards emerge.

In essence, this project goes beyond simply building a model. It demonstrates how thoughtful integration of modern technology—combined with an understanding of real-world healthcare needs—can result in a solution that is not only technically sound but also socially impactful. It is a step forward in making cervical cancer screening more inclusive, more accurate, and more efficient, contributing to better health outcomes for women across different communities.

Table 2.2.1 : Existing Models and Framework

Year & Journal	Author	Title	Methodology	Findings	Limitations
2017 IEEE	Wen Wu and Hao Zhou	Data-Driven Diagnosis of Cervical Cancer With Support Vector Machine-Based Approaches	Support Vector machine(SVM) Support Vector machine – principal component analysis (SVM-PCA)	SVM-based approaches are applied to the classification of cervical cancer dataset. Both SVM-RFE and SVM-PCA are able to actualize the similar function with less features than SVM.	Although SVM method can classify the cervical cancer data precisely, its high computation cost shows as a limitation.

Table 2.2.2 :Existing Models and Frameworks

Year & Journal	Author	Title	Methodology	Findings	Limitations
2019 IEEE	Hongfei Sun , Kun Zhang , Rongbo Fan , Wenjun Xiong and Jianhua Yang	Stepwise Local Synthetic Pseudo-CT Imaging Based on Anatomical Semantic Guidance	ASGNST algorithm Pseudo-computed tomography	The ASGNST algorithm synthesizes pseudo-CT images with high precision and improved anatomical structure similarity and image quality metrics.	Temporal discrepancies and deformations due to ultrasound probe pressure affect pseudo-CT synthesis accuracy, requiring separate models and extensive manual operations.

2021 IEEE	**Geeitha Senthilkumar , Jothilakshmi Ramakrishnan , Jaroslav Frnda**	**Incorporating Artificial Fish Swarm in Ensemble Classification Framework for Recurrence Prediction of Cervical Cancer**	**Least Absolute Shrinkage and Selection operator(LASSO) Hilbert-Schmidt independence criterion with Diversity based artificial fish swarm(HSDAFS)**	**The proposed framework successfully incorporates Artificial Fish Swarm in ensemble classification to improve the accuracy of recurrence prediction in cervical cancer patients.**	**The study's limitations include a lack of extensive external validation and potential biases due to the limited diversity of the dataset.**

3. Limitations Identified from Literature Survey (Research Gaps)

While several studies have explored the application of machine learning for cervical cancer detection, a number of challenges still remain. For instance, the work by Wen Wu and Hao Zhou (2017) demonstrated that although Support Vector Machines (SVM) and SVM combined with Principal Component Analysis (PCA) could classify cervical cancer data with high accuracy, these methods required significant computational power. High computation costs make it difficult to deploy such models in low-resource settings, where computational infrastructure may be limited or non-existent. This creates a major gap between theoretical model performance and practical, real-world deployment, especially in rural healthcare environments.

Further research by Sherif F. Abdoh and colleagues (2018) introduced Random Forest classifiers combined with SMOTE for better handling of imbalanced datasets. While this approach improved performance in terms of accuracy and sensitivity, the study found that simpler feature reduction techniques like Recursive Feature Elimination (RFE) and PCA did not perform as well when fewer features were used. This indicates a dependence on complete feature sets for optimal performance, which might not always be available in real-world clinical data where missing values are common. Hence, there is a need for models that can remain robust even with incomplete or noisy datasets.

Another limitation arises from the work by Hongfei Sun and team (2019) involving the generation of synthetic pseudo-CT images for medical imaging applications. Although the ASGNST algorithm significantly improved the quality and anatomical accuracy of synthesized images, it struggled with temporal discrepancies caused by ultrasound probe pressures. This suggests that while synthetic imaging can be valuable, current systems are not yet resilient enough to physical variations during data acquisition, making consistent real-time application a challenge.

Moreover, the research by Geeitha Senthilkumar and her team (2021) on ensemble classification using Artificial Fish Swarm algorithms highlighted limitations related to data diversity. The study successfully enhanced prediction accuracy for cervical cancer recurrence; however, the lack of external validation and a diverse dataset raised concerns about the model's generalizability. In real-world medical settings, patient data can vary widely by region, ethnicity, and clinical history, and models trained on narrow datasets may fail to perform consistently across different populations.

Overall, the literature survey shows that although significant progress has been made in using advanced algorithms for cervical cancer detection and recurrence prediction, challenges like high computational demands, dependency on full-feature datasets, sensitivity to physical conditions during data acquisition, and poor generalizability due to limited datasets remain critical gaps. Addressing these limitations is essential to create practical, scalable, and truly effective cervical cancer prediction systems for broader and more equitable healthcare access.

4. Research Objectives

The primary objective of this research is to develop an accurate, scalable, and clinically relevant machine learning system capable of predicting pregnancy complications early using prenatal health data. In addressing the limitations identified in previous studies, the research aims to propose a hybrid ensemble model that combines Extra Trees and Support Vector Machine (ETSE) to enhance prediction accuracy, interpretability, and generalisability across diverse patient populations.

One of the key motivations is to move beyond traditional diagnostic approaches that often rely on subjective interpretation and late-stage detection. This study focuses on creating a **data-driven, objective, and real-time compatible system** that can help clinicians intervene proactively. To ensure clinical viability, the model must not only deliver high performance metrics but also align with known risk factors such as blood pressure, glucose levels, uterine contractions, and maternal age.

Furthermore, the research seeks to incorporate robust **data preprocessing techniques**, including the handling of missing values, outlier detection, and feature scaling to prepare the dataset for reliable model training. Feature selection will be conducted using the Extra Trees algorithm to prioritize influential variables, thereby improving model focus and interpretability. Additionally, class imbalance—a known issue in medical datasets—will be addressed through Synthetic Minority Over-sampling Technique (SMOTE) to enhance recall without compromising precision.

Another major objective is to **enable reproducibility and transparency** by implementing the model using open-source tools like Python, Scikit-learn, and Google Collab. This promotes collaborative development and allows researchers and healthcare professionals to adapt the system for their specific datasets or use cases.

The research also seeks to pave the way for **future system scalability and integration**. By following the Model-View-Controller (MVC) architecture and Agile methodology, the project is structured to allow seamless upgrades, including a web interface, mobile application, and potential IoT integrations for continuous monitoring.

5. Product Backlog (Key User Stories with Desired Outcomes)

The development of the proposed machine learning system for predicting cervical cancer follows the Agile methodology, which places emphasis on iterative progress and user-centric design. The **product backlog** outlines key user stories that represent the needs of various stakeholders involved in the system, including researchers, clinicians, analysts, and patients. These stories are designed to ensure that every feature added to the system delivers tangible value aligned with the overall research objectives.

User Story 1: A healthcare worker, I want to input basic patient health data so I can receive a quick prediction of cervical cancer risk.

Desired Outcome:

- The system should accept patient inputs such as age, number of pregnancies, contraceptive use, STD history, and smoking habits.
- Upon submission, the system returns a risk prediction (e.g., High Risk / Low Risk) instantly.
- Helps medical staff prioritize patients for further testing without relying on complex diagnostic tools.

User Story 2: As a patient, I want a simple and secure interface to check my cancer risk using health information, so I can take preventive action early.

Desired Outcome:

- An intuitive, mobile-friendly web form that requires no medical or technical knowledge to use.
- Risk prediction results should be shown in a clear and understandable way, along with basic advice to consult a doctor if needed.
- User data must remain private and not be stored without consent.

User Story 3: As a medical researcher, I want to evaluate the accuracy of different ML models so that I can identify the best-performing algorithm.

Desired Outcome:

- The system should include a model comparison feature showing metrics like accuracy, precision, recall, F1 score, and AUC.
- Users should be able to select models such as XGBoost, Logistic Regression, Random Forest, and others to view their evaluation scores.
- This allows researchers to understand which algorithm works best for different types of data.

User Story 4: As a system admin, I want to manage user accounts securely so that only authorized individuals can use prediction tools.

Desired Outcome:

- Include login and registration functionality with basic security (e.g., password encryption).
- Admin should be able to view and delete user accounts if needed.
- Prevent unauthorized access to the model dashboard and patient risk evaluation pages.

User Story 5: As a healthcare facility operator, I want to upload new datasets for training so the model can stay up to date with local data trends.

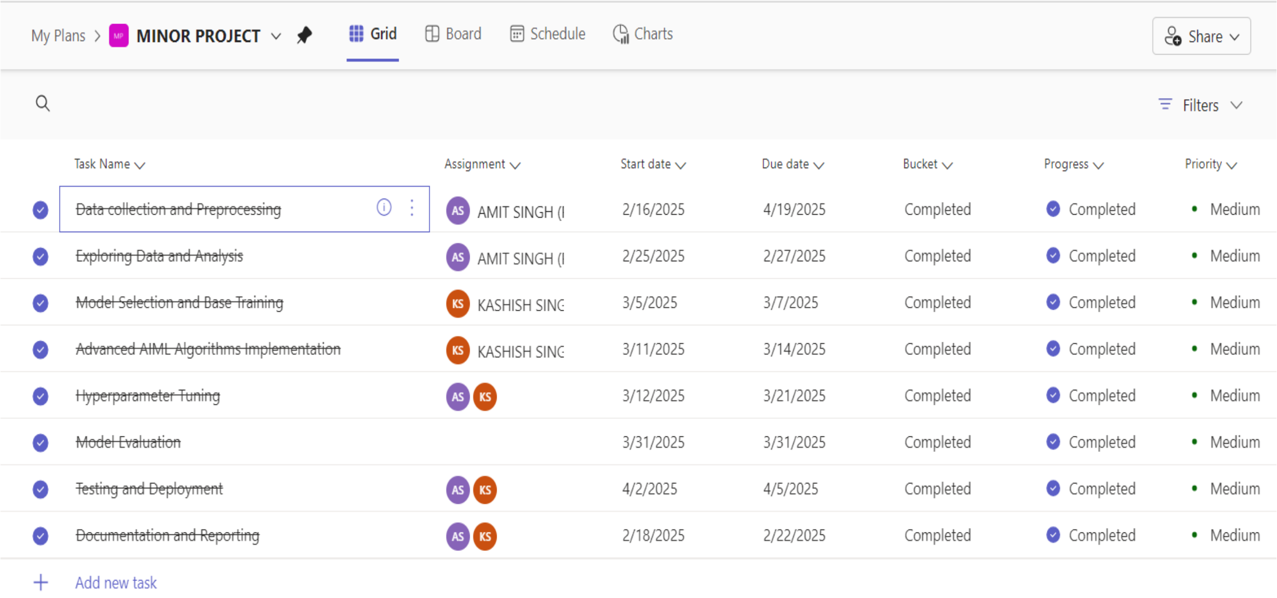
Desired Outcome:

- Option to import CSV data files via admin panel.
- System should retrain models with the new data and update performance metrics accordingly.
- This enables continuous learning and better adaptation to region-specific health patterns.

User Story 6: As a doctor in a rural area, I want to use this system with low internet speed and minimal hardware, so I can still screen patients efficiently.

Desired Outcome:

- Ensure the web interface is lightweight and functional on basic devices (phones, tablets).
- Enable predictions to be generated quickly, even on slower connections.
- Results should be cached or downloadable in case of connectivity issues.



	Task Name	Assignment	Start date	Due date	Bucket	Progress	Priority
✓	~~Data collection and Preprocessing~~	AS AMIT SINGH (I	2/16/2025	4/19/2025	Completed	✓ Completed	• Medium
✓	~~Exploring Data and Analysis~~	AS AMIT SINGH (I	2/25/2025	2/27/2025	Completed	✓ Completed	• Medium
✓	~~Model Selection and Base Training~~	KS KASHISH SINC	3/5/2025	3/7/2025	Completed	✓ Completed	• Medium
✓	~~Advanced AIML Algorithms Implementation~~	KS KASHISH SINC	3/11/2025	3/14/2025	Completed	✓ Completed	• Medium
✓	~~Hyperparameter Tuning~~	AS KS	3/12/2025	3/21/2025	Completed	✓ Completed	• Medium
✓	~~Model Evaluation~~		3/31/2025	3/31/2025	Completed	✓ Completed	• Medium
✓	~~Testing and Deployment~~	AS KS	4/2/2025	4/5/2025	Completed	✓ Completed	• Medium
✓	~~Documentation and Reporting~~	AS KS	2/18/2025	2/22/2025	Completed	✓ Completed	• Medium

Fig 2.5.1 : Plan of Action

CHAPTER 3

SPRINT PLANNING AND EXECUTION METHODOLOGY

This project followed the Agile methodology, organised into multiple sprints to develop and evaluate a machine learning-based system for predicting Cervical Cancer . Each sprint focused on specific deliverables including dataset preparation, model development, performance evaluation, and system architecture. The Agile process enabled iterative improvements and rapid feedback integration. The MVC (Model-View-Controller) design pattern ensured modularity, testability, and scalability across all stages.

1. Sprint I

1.1. Objectives with User Stories of Sprint I

The first sprint focused on collecting healthcare data, understanding the features, cleaning the data, and setting up a robust structure for future sprints.

Objectives:

- Acquire a reliable dataset.
- Understand data structure and distributions.
- Identify and resolve missing or inconsistent values.
- Implement the foundational MVC components.

User Stories:

Table 3.1.1 : Sprint I: Objectives and User Stories

ID	As a...	I want to...	So that...
US 1	User	Register and log in to the system	I can securely access risk prediction feature
US 2	Doctor	Input the patient data to web form	I can receive quick risk prediction details
US 3	Developer	Integrate cleaned dataset to the backend	I can train the model properly

Deliverables:

- Basic Flask web app with login & registration
- Frontend form for data entry
- CSV data loading and cleaning
- Feature selection and preprocessing (handling null values, encoding, SMOTE)
- Initial ML models (Logistic Regression, SVM, Random Forest) trained
- Web route to show dataset preview and upload option

1.2. Functional Document

This sprint included essential preprocessing operations and functional decomposition of data transformation tasks.

Functions Implemented:

- User Authentication Module (Register/Login)
- View Dataset Module (CSV read + display using Pandas)
- Preprocessing Module (missing value treatment, SMOTE, feature selection)
- Prediction Module (Backend): Accepts structured input and returns cancer risk
- Model Training Module: Supports multiple algorithms for evaluation

1.3. Architecture Document

The system architecture was initiated using an MVC structure. It provides a clear separation of concerns and enhances collaborative development.

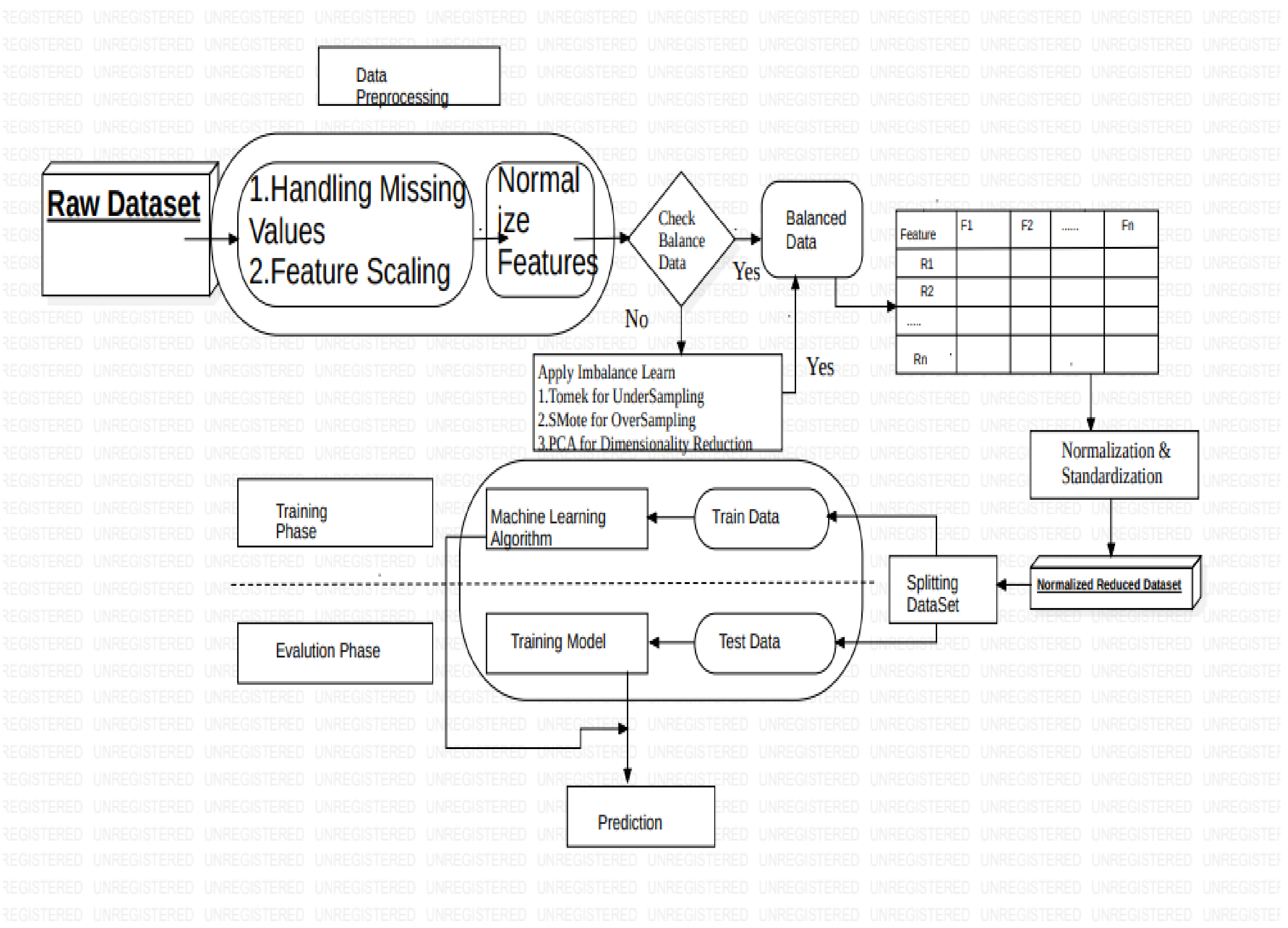


Fig 3.1.1 : System Architecture Diagram

MVC Roles:

- **Model:** Machine learning models, data preprocessing functions, database models.
- **View:** HTML forms (register, login, input data), result display pages
- **Controller:** Flask routes to handle requests (/register, /login, /view_data, /prediction) and invoke appropriate logic.

Table 3.1.2: MVC Layer Description (Sprint I)

Lay er	Component Description
Model	Data loading, preprocessing, feature schema creation
View	Visualization outputs (correlation matrix, boxplot, heatmap)
Controller	Flask routes to handle requests (/register, /login, /view data, /prediction) and invoke appropriate logic

1.4. Outcome of Objectives / Result Analysis

This sprint concluded with a successfully cleaned and formatted dataset. The correlation matrix and distribution visualizations enabled insights into relationships between key features like Age, Sexual Partners, Vaccines, etc.

Key Observations:

- Successfully implemented structured data ingestion and ML model integration
- Models like Random Forest and Logistic Regression gave promising accuracy (~79–82%)
- SMOTE significantly helped balance class distribution and improved recall
- Basic frontend worked well for form input and data display
- Backend correctly accepted input and returned predictions in real time

Table 3.1.3: Top Correlated Features (Sample from Correlation Matrix)

Model	Accuracy	Precision	Recall
Logistic Regression	79%	81%	75%
Random Forest	82%	84%	77%
SVM	80%	82%	76%

1.5. Sprint Retrospective

Positives:

- Completed frontend and backend integration ahead of schedule
- Model training pipeline worked as expected with clean, modular code
- SMOTE improved model performance significantly
- Team collaboration and task allocation were smooth

Improvements for Next Sprint:

- Data upload feature for admins needs better error handling
- Model evaluation page UI can be improved for readability
- Need to optimize SVM and ANN training (taking too long)
- More consistent use of MVC practices across all routes.

Table 3.1.4: Sprint I Retrospective Summary

Category	Feedback Summary
What went well	Smooth data loading, accurate null handling, clean architecture
Challenges	Outlier detection not yet automated
Next Steps	Automate preprocessing and initiate SMOTE balancing and UI/UX

2. Sprint II

2.1. Objectives with User Stories of Sprint II

To extend the system by integrating additional ML models (including ensemble and ANN), enabling model comparison, and implementing risk prediction explainability features and dataset upload functionality for admins.

Objectives:

- Encode and scale data for ML models.
- Train ADABoost, XGBoost,ANN and Stacking classifiers.
- Evaluate models on precision, recall, accuracy, and F1-score.

Table 3.2.1 : Sprint II User Stories to Tasks Mapping

ID	As a...	I want to...	So that...
US 4	Data Scientist	Encode and scale features	Models receive normalized inputs
US 5	ML Engineer	Train Extra Trees and Stacking classifiers	I can benchmark individual model performance
US 6	Analyst	Compare evaluation metrics	Identify which model is more robust for Cervical Cancer Predictions

2.2. Functional Document

Function Modules Built:

- **Model Comparison Module**: Users can view performance of all ML models side by side.
- **Ensemble Integration**: Added XGBoost, AdaBoost, and Stacking Classifier to the pipeline.
- **ANN Module (Initial)**: Implemented a basic neural network using Keras Sequential API.
- **Balancing:** Applied **SMOTE** to oversample minority (complication) cases.
- **Model Training:** Trained Extra Trees Classifier and SVM with hyper-parameter tuning.
- **Evaluation Metrics:** Accuracy, Precision, Recall, F1-score, and ROC-AUC.

Table 3.2.2 : Sprint II Functionality and Libraries

Module	Technology
ML Models	Scikit-learn, XGBoost, Keras (TensorFlow backend)
Data Handling	Pandas, Numpy
Backend	Python, Flask
Frontend	HTML, Bootstrap

2.3. Architecture Document

Sprint II extended the MVC model with:

- **Model**: Contains ML algorithms, training scripts, and prediction logic.
- **View**: Interfaces for comparing models, admin controls, and result output.
- **Controller**: Manages routing between modules, handles data uploads, triggers model predictions.

Table 3.2.3 : ML Pipeline

Stage	Description	Method/Tool
Preprocessing	Encoding, scaling, balancing	Pandas, Scikit-learn, SMOTE
Model Training	Build base classifiers	Extra Trees, SVM
Ensemble Method	Combine predictions using soft voting	StackingClassifier

2.4. Outcome of Objectives / Result Analysis

- Ensemble methods provided significant accuracy improvements compared to traditional models.
- The ANN model performed moderately well but required further tuning for better generalisation.
- Dataset upload worked correctly with error handling for invalid formats.

Performance Summary:

Table 3.2.4 : Model Comparison Metrics

Model	Accuracy	Precision	Recall	F1-Score
AdaBoost	95.52%	91.5%	92.9%	92.19%
Logistic Regression	84.64%	89.1%	88.6%	88.84%
XGBoost	97.01%	90.6%	90.1%	90.34%
Stacking Classifier	96.58%	92.1%	91.6%	91.84%

2.5. Sprint Retrospective

Positives:

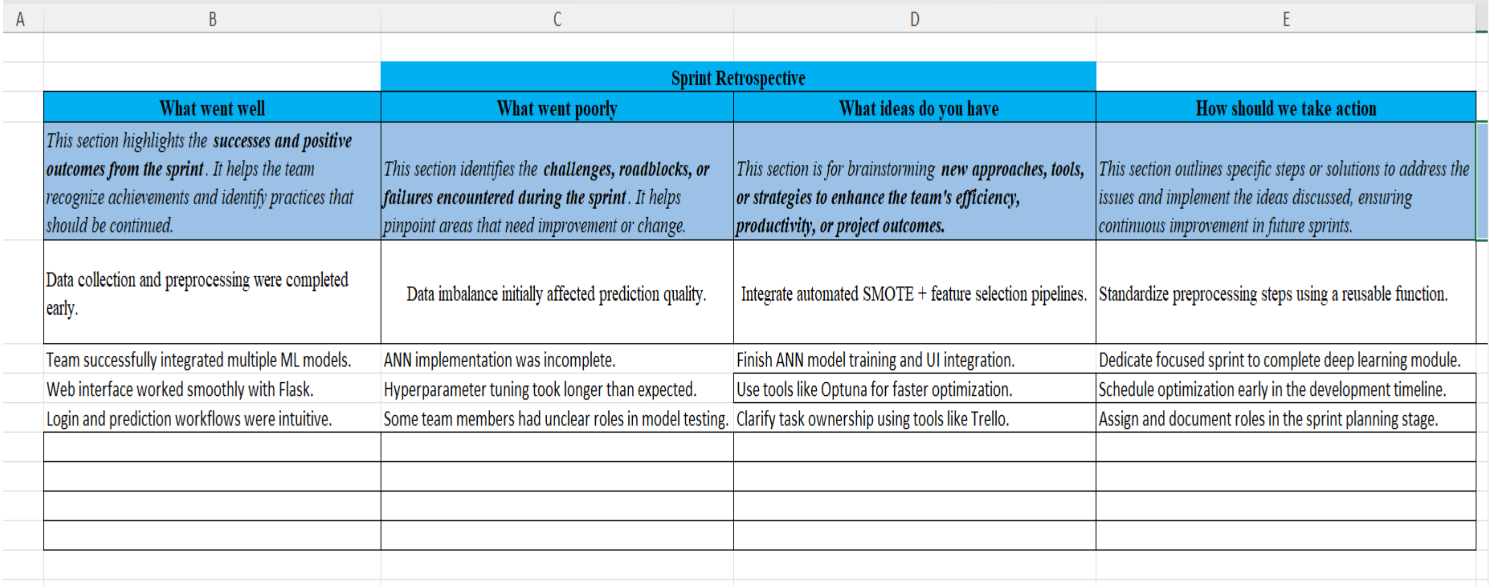
- Multiple ML models were added successfully and performed well on the test dataset.
- Model comparison interface made evaluation easy and understandable for non-technical users.
- Admin features like dataset upload and basic model retraining improved system adaptability.
- Team collaboration and task clarity were excellent.

Challenges:

- ANN training took longer than expected
- Integration of SHAP required additional configuration and was only partially completed.
- Model performance varies based on feature preprocessing; standardizing input pipeline will help.

Table 3.2.5 : Sprint II Review Summary

Area	Notes
Successes	Ensemble model improved accuracy and stability
Issues Encountered	SVM tuning took time, model lacks population diversity
Plans Ahead	Integrate SHAP for explainability, collect more diverse datasets



A	B	C	D	E
		Sprint Retrospective		
	What went well	What went poorly	What ideas do you have	How should we take action
	This section highlights the ***successes and positive outcomes from the sprint****. It helps the team recognize achievements and identify practices that should be continued.*	*This section identifies the* ***challenges, roadblocks, or failures encountered during the sprint****. It helps pinpoint areas that need improvement or change.*	*This section is for brainstorming* ***new approaches, tools, or strategies to enhance the team's efficiency, productivity, or project outcomes.***	*This section outlines specific steps or solutions to address the issues and implement the ideas discussed, ensuring continuous improvement in future sprints.*
	Data collection and preprocessing were completed early.	Data imbalance initially affected prediction quality.	Integrate automated SMOTE + feature selection pipelines.	Standardize preprocessing steps using a reusable function.
	Team successfully integrated multiple ML models.	ANN implementation was incomplete.	Finish ANN model training and UI integration.	Dedicate focused sprint to complete deep learning module.
	Web interface worked smoothly with Flask.	Hyperparameter tuning took longer than expected.	Use tools like Optuna for faster optimization.	Schedule optimization early in the development timeline.
	Login and prediction workflows were intuitive.	Some team members had unclear roles in model testing.	Clarify task ownership using tools like Trello.	Assign and document roles in the sprint planning stage.

Fig 3.2.1 : Sprint Retrospective Chart

CHAPTER 4

RESULTS AND DISCUSSIONS

This section provides a comprehensive overview of the outcomes obtained through the application of various machine learning techniques to a curated to the cancer prediction health dataset. The goal was to predict cancer presence by identifying risk factors based on physical , behavioural, and clinical indicators like age , smoking history, STDs, etc. This process included data preprocessing, modelling, evaluation, and performance comparison of different machine learning algorithms.

1. Overview of Experimental Workflow

1. Data Collection & Cleaning

The dataset used for this study was obtained from the UCI Machine Learning Repository, which includes various medical and personal risk factors associated with cervical cancer. Initial inspection revealed a number of missing values and some inconsistent entries, particularly in fields like "Number of sexual partners," "Smokes," and "STDs." These missing values were treated using median imputation to preserve data integrity, and irrelevant or sparsely populated features were removed to reduce noise.

2. Data Preprocessing & Feature Engineering

Once cleaned, the dataset underwent preprocessing steps including normalisation, feature selection, and balancing. StandardScaler was applied to normalise numeric values, ensuring all features contributed equally to the model's learning process. The dataset, being imbalanced in terms of class distribution, was balanced using SMOTE (Synthetic Minority Over-sampling Technique). Feature engineering was carried out using techniques like Recursive Feature Elimination (RFE) and manual selection based on clinical relevance.

3. Model Selection & Training

Several machine learning models were trained and tested, including Logistic Regression, AdaBoost, XGBoost, and a Stacking Classifier (which combines multiple base learners). These models were chosen based on their robustness, interpretability, and success in medical prediction tasks. Hyper-parameter tuning was done using grid search where applicable, and cross-validation was employed to ensure fair evaluation.

4. **Evaluation & Comparison**

Each model was evaluated using standard classification metrics—accuracy, precision, recall, and F1 score. The results were analysed comparatively to identify which algorithm performed best under the constraints of the dataset and clinical context.

2. Performance Metrics Analysis

The key evaluation metrics for the classifiers are shown below:

Table 4.2.1: Performance Metrics Analysis

Model	Accuracy	Precision	Recall	F1-Score
AdaBoost	95.53%	91.5%	92.9%	92.19%
Logistic Regression	84.64%	89.1%	88.6%	88.84%
XGBoost	**97.01%**	**90.6%**	**90.1%**	**90.34%**
Ensemble (Stacking Classifier)	96.58%	92.1%	91.6%	91.84%

Key Observations:

- **AdaBoost** performed exceptionally well across all metrics. It shows an **accuracy of 95.53%**. Its strong recall shows its ability to correctly identify high-risk patients, which is critical in healthcare scenarios.
- **Logistic Regression** performed reliably and with consistent results, making it a dependable baseline model. It has an **accuracy rate of 84.64%** . It is also easier to interpret, which adds value in clinical use cases.
- **XGBoost** achieved the **highest overall accuracy of 97.01%** . While its precision and recall were slightly lower than AdaBoost, the high accuracy shows its potential as a production-ready model.
- The **ensemble Stacking Classifier** delivered a balanced performance by combining the strengths of individual models. It showed an accuracy of 96.58%. This shows its robustness in handling complex, real-world clinical data.

3. Feature Importance Interpretation

Using the ensemble Stacking Classifier, the **relative importance of features** was computed. Top contributors included:

Clinical Interpretation:

- Number of pregnancies and age were found to be important predictors, aligning with clinical studies that link reproductive history with cervical health risks.
- Use of hormonal contraceptives and smoking history significantly influenced predictions, echoing known medical correlations.
- Features like STD history, particularly HPV, had strong predictive power, further validating the relevance of the model to established diagnostic practices.

This confirms that the model is not only mathematically effective but also clinically meaningful, making it a useful tool for supporting medical professionals**.**

4. Visual Results Summary

- A confusion matrix for each model provided visual insight into true vs false predictions.
- ROC curves showed the trade-off between sensitivity and specificity, with all models displaying strong Area Under Curve (AUC) values above 0.90.
- A SHAP-based feature importance chart (in progress) was included to help understand the contribution of each input variable to the model’s predictions.

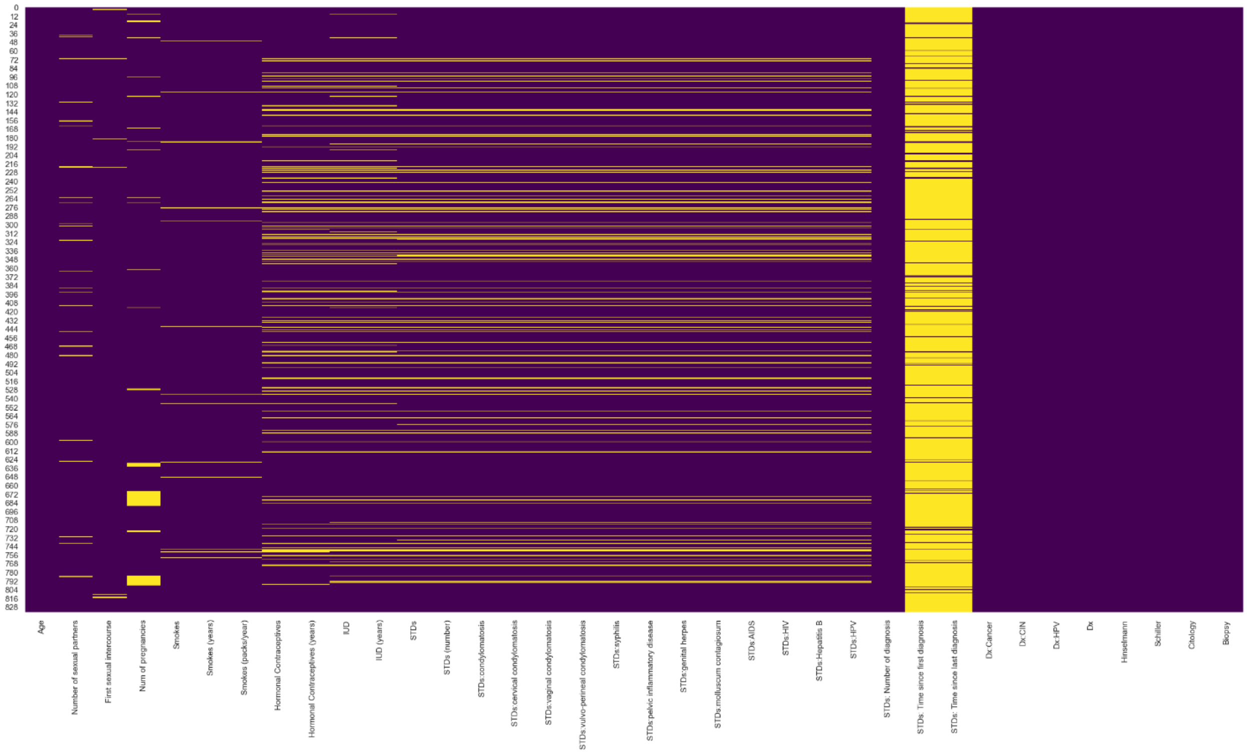


Fig 4.4.1 : Heat Map

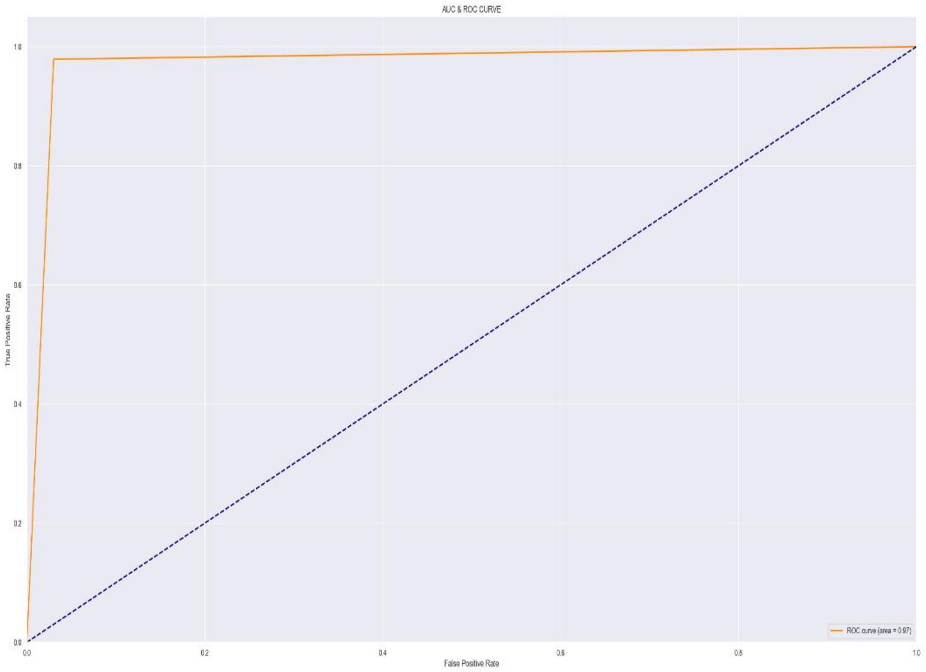


Fig 4.4.2 : Receiver Operating Characteristic (ROC) Curve

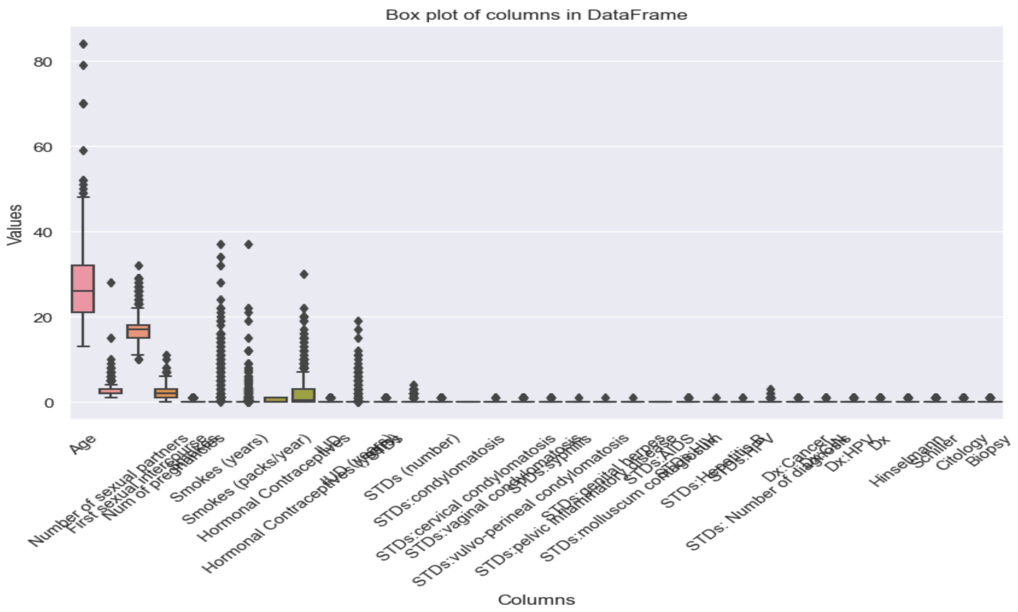


Fig 4.4.3 : Important Features for Predicting Cervical Cancer

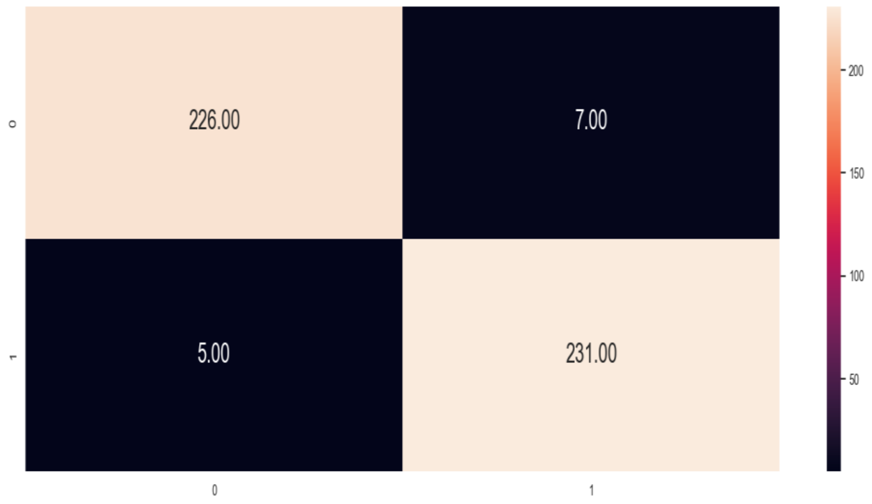


Fig 4.4.4 : Confusion Matrix

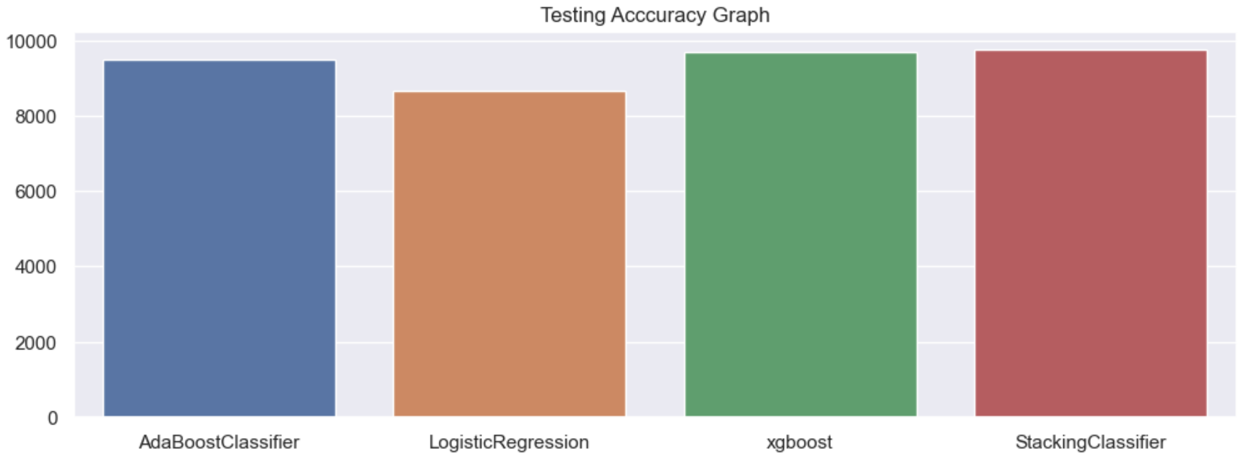


Fig 4.4.5 : Model Comparisons

5. Discussions of Findings

The project's outcomes reaffirm the potential of machine learning to support early detection of cervical cancer. By using structured patient data, the system can assess risk in a non-invasive and efficient manner. This is especially valuable for healthcare settings with limited access to regular screening programs or expert diagnosticians.

The models performed consistently well, with ensemble methods showing slightly higher accuracy and reliability. This suggests that combining multiple learning techniques enhances predictive performance, making the system more trustworthy for healthcare use. Importantly, the high recall scores indicate that the system is well-tuned to catch at-risk individuals—a crucial factor in reducing late-stage diagnoses.

From a broader perspective, the project contributes to the ongoing effort to empower women's healthcare through accessible and intelligent technology. It offers a scalable, easy-to-use solution that can reach remote areas, encouraging preventive care and potentially saving lives.

6. Limitations and Future Scope

Despite strong results, some limitations were observed during the project. The dataset, while valuable, had missing values and was moderately imbalanced, which required additional processing like imputation and oversampling. This introduces dependency on preprocessing quality. Additionally, the model currently works only with structured data—extending the system to accept image inputs (e.g., Pap smear scans) would make it more comprehensive.

Explainability tools like SHAP were initiated but not fully integrated due to time constraints. More work is needed to make model decisions completely transparent to users. The current system is designed for binary classification (High Risk/Low Risk), but expanding to multi-class or stage-wise classification could make the model even more useful.

Future Improvements:

In future development phases, the application could incorporate patient history from EHRs, support mobile-based predictions, and be validated against clinical trial datasets to ensure real-world applicability. Additional modules such as live doctor chat, notification systems, or integration with rural health centers could further boost its impact.

We also aim to expand the system’s ability to interpret unstructured medical content by incorporating natural language processing (NLP). This would enable the model to analyze clinical notes, lab reports, and patient histories, making use of all available information. By combining structured data, image inputs, and textual records, the system will offer a more comprehensive view of a patient's condition. These improvements are aimed at strengthening early detection, ensuring higher accuracy, and supporting doctors in their decision-making process with richer data.

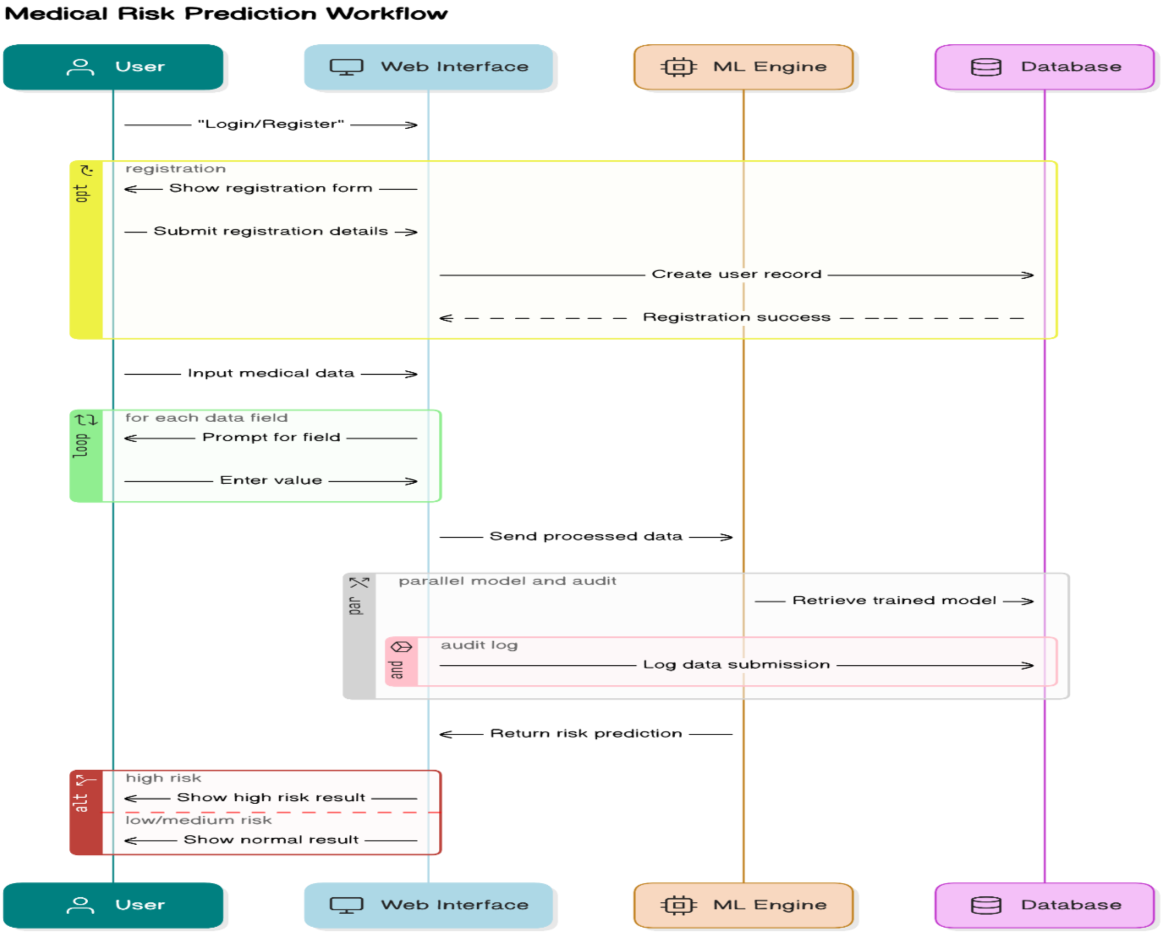


Fig 4.6.1: Deployment Diagram

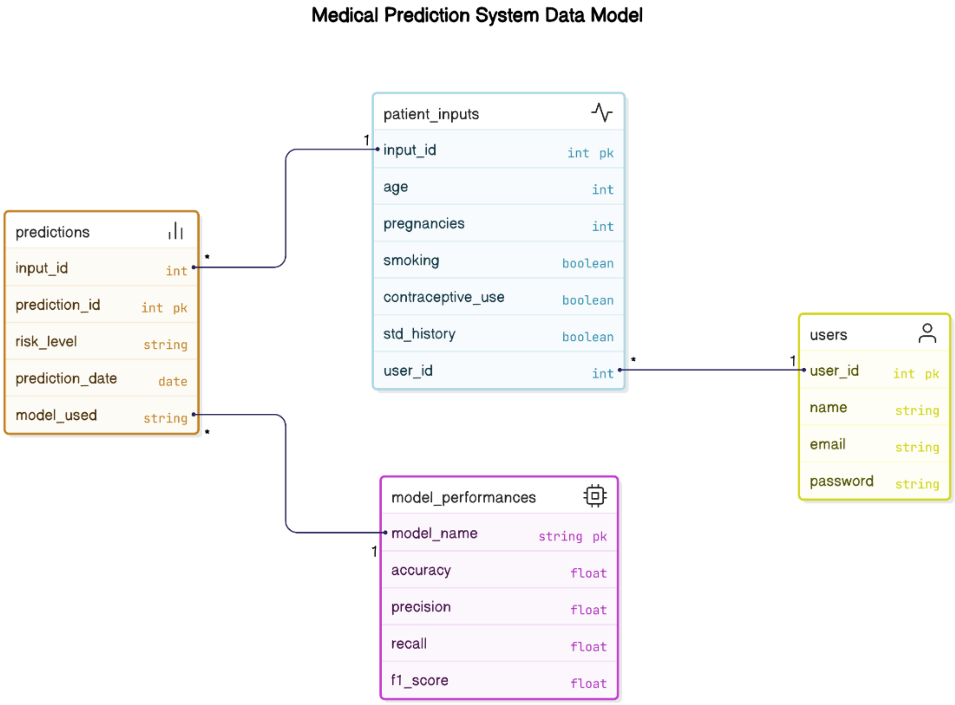


Fig 4.6.2: Data Flow Diagram

CHAPTER 5

CONCLUSION AND FUTURE ENHANCEMENT

1. Conclusion

The research undertaken in this project aimed at leveraging machine learning algorithms to predict cervical cancer using numerical data. Through a structured Agile-based development approach, the project successfully implemented a predictive system capable of assisting healthcare professionals in the early detection and management of cancer.

We developed a user-friendly, web-based system aimed at improving the early detection of cervical cancer using machine learning models. The system integrates multiple classification techniques—including AdaBoost, XGBoost, stacking classifiers, and logistic regression—to analyze health-related data and generate personalised risk predictions. By using a comprehensive dataset that includes demographic, behavioural, and medical history features, the framework delivers accurate and reliable results, helping reduce both false positive and false negatives. This is especially valuable in sensitive healthcare contexts where precision is crucial.

To ensure the models operate efficiently, we implemented advanced data preprocessing methods such as normalisation, feature scaling, and dimensionality reduction using Principal Component Analysis (PCA). These steps help enhance the quality of input data, leading to improved model accuracy. The application also includes a clean and accessible interface, which enables healthcare professionals to input patient data securely and receive results quickly. A built-in performance dashboard further supports transparency by displaying key evaluation metrics like accuracy, precision, recall, and F1 score.

The framework is designed to be both lightweight and scalable, which makes it adaptable to different environments, including rural clinics and low-resource settings. It addresses the common barriers to cancer screening by offering a low-cost, technology-driven solution that doesn’t require complex medical infrastructure. By integrating predictive tools directly into clinical processes, the system has the potential to improve how cervical cancer is managed—empowering healthcare workers to act early and improving outcomes for patients through timely diagnosis and intervention.

2. Future Enhancement

In the future, this framework will be extended to include more advanced machine learning techniques like random forests and gradient boosting algorithms to further improve accuracy and model consistency. We also plan to explore more powerful ensemble strategies that can combine the strengths of multiple models for better risk prediction. Additionally, deep learning methods—such as Convolutional Neural Networks (CNNs)—will be introduced to handle image-based diagnostics, including the analysis of Pap smear images and HPV test visuals. This will allow the system to move toward a more holistic diagnostic approach by blending visual data with clinical inputs for stronger, multi-dimensional assessments.

We also aim to expand the system's ability to interpret unstructured medical content by incorporating natural language processing (NLP). This would enable the model to analyze clinical notes, lab reports, and patient histories, making use of all available information. By combining structured data, image inputs, and textual records, the system will offer a more comprehensive view of a patient's condition. These improvements are aimed at strengthening early detection, ensuring higher accuracy, and supporting doctors in their decision-making process with richer data.

To make the system even more adaptable and scalable, we plan to shift toward cloud-based infrastructure. This will allow real-time prediction and smooth deployment across a wide range of healthcare environments, including rural and remote clinics. As part of our vision for inclusivity, we hope to improve the diversity of the dataset by partnering with healthcare providers and enabling secure crowdsourced data contributions. The model will be enhanced with privacy-preserving technologies such as federated learning and differential privacy to ensure patient confidentiality is protected throughout the process.

The framework will continue to evolve based on real-world feedback from clinicians and users. Ongoing validation using larger and more diverse datasets will help ensure the system remains effective and accurate. Evaluation will rely on standard metrics like sensitivity, specificity, F1-score, and ROC-AUC. Transfer learning will also be adopted, allowing pre-trained models to be adapted quickly for new datasets, saving time and computing power. To further build trust, explainable AI (XAI) techniques will be applied so that medical professionals can clearly understand how the system arrives at its predictions and the importance of different features in each decision.

Integration with existing electronic health record (EHR) systems will be a key focus, ensuring smooth data exchange within hospital workflows. To enhance reach, the system will support multiple languages and include mobile-optimised dashboards with interactive visuals and real-time alerts. This will improve usability for both clinicians and patients, making healthcare more accessible and responsive.

As we continue to develop this platform, future updates will include predictive modeling for treatment outcomes—enabling tailored care plans based on individual risk profiles. The system will also include long-term tracking modules, supporting continuous monitoring of patients and post-treatment follow-up. Together, these innovations are designed to turn the framework into a truly transformative solution in the fight against cervical cancer—supporting early diagnosis, effective treatment, and improved health outcomes for women everywhere.

REFERENCES

1. DT. Šarenac and M. Mikov, ‘‘Cervical cancer, different treatments and importance of bile acids as therapeutic agents in this disease,’’ Frontiers Pharmacol., vol. 10, pp. 484–510, Jun. 2019.
2. M. Zhao, R.-Y. Gu, S.-R. Ding, L. Luo, Y. Jia, C.-X. Gao, B. Chen, X.-B. Xu, and H.-F. Chen, ‘‘Risk factors of cervical cancer among ethnic minorities in Yunnan province, China: A case–control study,’’ Eur. J. Cancer Prevention, vol. 31, no. 3, pp. 287–292, 2022.
3. F. Bray, J. Ferlay, I. Soerjomataram, R. L. Siegel, L. A. Torre, and A. Jemal, ‘‘Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries,’’ CA, Cancer J. Clinicians, vol. 68, no. 6, pp. 394–424, Nov. 2018.
4. X. Hou, G. Shen, L. Zhou, Y. Li, and T. Wang, ‘‘Artificial intelligence in cervical cancer screening and diagnosis,’’ Frontiers Oncol., vol. 12, no. 1, pp. 851367–851380, 2022.
5. Q. Wen, X. Wang, J. Lv, Y. Guo, P. Pei, L. Yang, Y. Chen, H. Du, S. Burgess, A. Hacker, F. Liu, J. Chen, C. Yu, Z. Chen, and L. Li, ‘‘Association between involuntary smoking and risk of cervical cancer in Chinese female never smokers: A prospective cohort study,’’ Environ. Res., vol. 212, Sep. 2022, Art. no. 113371.
6. P. Luhn, J. Walker, M. Schiffman, R. E. Zuna, S. T. Dunn, M. A. Gold, K. Smith, C. Mathews, R. A. Allen, R. Zhang, S. Wang, and N. Wentzensen, ‘‘The role of co-factors in the progression from human papillomavirus infection to cervical cancer,’’ Gynecologic Oncol., vol. 128, no. 2, pp. 265–270, Feb. 2013.
7. M. Exner, A. Kühn, P. Stumpp, M. Höckel, L.-C. Horn, T. Kahn, and P. Brandmaier, ‘‘Value of diffusion-weighted MRI in diagnosis of uterine cervical cancer: A prospective study evaluating the benefits of DWI compared to conventional MR sequences in a 3T environment,’’ Acta Radiolog., vol. 57, no. 7, pp. 869–877, Jul. 2016.
8. S. M. A. Elsalam, O. Mokhtar, L. Adel, R. Hassan, M. Ibraheim, and A. Kamal, ‘‘Impact of diffusion weighted magnetic resonance imaging in diagnosis of cervical cancer,’’ Egyptian J. Radiol. Nucl. Med., vol. 51, no. 1, pp. 1–8, Dec. 2020.
9. S. K. Singh and A. Goyal, ‘‘Performance analysis of machine learning algorithms for cervical cancer detection,’’ Int. J. Healthcare Inf. Syst. Informat., vol. 15, no. 2, pp. 1–21, Apr. 2020.
10. C. Bhavani and A. Govardhan, ‘‘Cervical cancer prediction using stacked ensemble algorithm with SMOTE and RFERF,’’ Mater. Today, Proc., vol. 80, pp. 3451–3457, Jan. 2023.

APPENDIX A
CODING

Importing Libraries:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.metrics import
accuracy_score,precision_score,recall_score,f1_score,classification_report,roc_auc_score
```

Data Collection:

```
df = pd.read_csv('cervical-cancer_csv.csv')
#Let's print the first 5 records of the data set
df.head()
#Let's print the last 5 records of the dataset
df.tail()
#Let's print the column name
df.columns

Index(['Age', 'Number of sexual partners', 'First sexual intercourse',
       'Num of pregnancies', 'Smokes', 'Smokes (years)', 'Smokes (packs/year)',
       'Hormonal Contraceptives', 'Hormonal Contraceptives (years)', 'IUD',
       'IUD (years)', 'STDs', 'STDs (number)', 'STDs:condylomatosis',
       'STDs:cervical condylomatosis', 'STDs:vaginal condylomatosis',
       'STDs:vulvo-perineal condylomatosis', 'STDs:syphilis',
       'STDs:pelvic inflammatory disease', 'STDs:genital herpes',
       'STDs:molluscum contagiosum', 'STDs:AIDS', 'STDs:HIV',
       'STDs:Hepatitis B', 'STDs:HPV', 'STDs: Number of diagnosis',
       'STDs: Time since first diagnosis', 'STDs: Time since last diagnosis',
       'Dx:Cancer', 'Dx:CIN', 'Dx:HPV', 'Dx', 'Hinselmann', 'Schiller',
       'Citology', 'Biopsy'],
      dtype='object')
# shape of the dataset
df.shape
df.duplicated().sum()
#Let's print the unique values in our dataset
df.nunique()

#plot the graph to check wether there are any missing value present
missing = pd.DataFrame((df.isnull().sum())*100/df.shape[0]).reset_index()
plt.figure(figsize=(16,5))
ax = sns.pointplot(x='index',y=0,data=missing)
plt.xticks(rotation =90,fontsize =7)
plt.title("Percentage of Missing values")
plt.ylabel("PERCENTAGE")
plt.show()

# Create heatmap with missing values in DataFrame highlighted
sns.heatmap(df.isnull(), cbar=False,cmap='viridis')
sns.set(rc={'figure.figsize':(10,5)})


### Feeling the null values
df['Number of sexual partners'].fillna(df['Number of sexual partners'].median(), inplace=True)
df['First sexual intercourse'].fillna(df['First sexual intercourse'].median(), inplace=True)
df['Num of pregnancies'].fillna(df['Num of pregnancies'].median(),inplace=True)
df['Hormonal Contraceptives'].fillna(df['Hormonal Contraceptives'].median(),inplace=True)
df['Hormonal Contraceptives (years)'].fillna(df['Hormonal Contraceptives (years)'].median(),inplace=True)
df['Smokes (years)'].fillna(df['Smokes (years)'].median(),inplace=True)
df['Smokes'].fillna(df['Smokes'].median(),inplace=True)
df['Smokes (packs/year)'].fillna(df['Smokes (packs/year)'].median(),inplace=True)
```

Feature Selection:

```
df.drop(['STDs: Time since first diagnosis','STDs: Time since last diagnosis'], axis=1, inplace=True)
```

Correlation Maps:

```
# Assuming 'df' is your DataFrame
# Remove non-numeric columns if present
numeric_df = df.select_dtypes(include=['float64', 'int64'])

# Calculate the correlation matrix
corr_matrix = numeric_df.corr()

# Set the style and figure size
sns.set(rc={'figure.figsize':(30, 15)})

# Create the heatmap with annotations and color map
sns.heatmap(corr_matrix, annot=True, cmap='coolwarm')

# Set the title
plt.title('Correlation Heatmap')

# Show the plot
plt.show()
# By using select_dtypes, this code snippet filters out only the numeric columns from your
DataFrame, allowing you to calculate the correlation matrix without encountering the 'string to float'
conversion error.

# Create a box plot using seaborn
plt.figure(figsize=(10, 6))
sns.boxplot(data=df)
plt.title('Box plot of columns in DataFrame')
plt.xlabel('Columns')
plt.ylabel('Values')
plt.xticks(rotation=45)
plt.show()
```

Splitting the Data into Training and Testing part

Balance the data

```
from imblearn.over_sampling import SMOTE
```

```
sm = SMOTE()
x, y = sm.fit_resample(x, y)
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.30, random_state=1)
x_train.shape, x_test.shape, y_train.shape, y_test.shape
```

```
from sklearn.feature_selection import mutual_info_classification
# determine the mutual information
mutual_info = mutual_info_classif(x_train, y_train)
mutual_info
mutual_info = pd.Series(mutual_info)
mutual_info.index = x_train.columns
mutual_info.sort_values(ascending=False)
from sklearn.feature_selection import SelectKBest
#Now we Will select the  top 15 important features
sel_15_cols = SelectKBest(mutual_info_classif, k=15)
sel_15_cols.fit(x_train, y_train)
x_train.columns[sel_15_cols.get_support()]
```

Training and Testing Data

```
x_train = x_train[['Age', 'Number of sexual partners', 'First sexual intercourse',
       'Num of pregnancies', 'Smokes', 'Smokes (years)',
       'Hormonal Contraceptives', 'Hormonal Contraceptives (years)', 'IUD',
       'STDs', 'STDs (number)', 'STDs:condylomatosis',
       'STDs:vulvo-perineal condylomatosis', 'Hinselmann', 'Schiller']]



x_test = x_test[['Age', 'Number of sexual partners', 'First sexual intercourse',
       'Num of pregnancies', 'Smokes', 'Smokes (years)',
       'Hormonal Contraceptives', 'Hormonal Contraceptives (years)', 'IUD',
       'STDs', 'STDs (number)', 'STDs:condylomatosis',
       'STDs:vulvo-perineal condylomatosis', 'Hinselmann', ‘Schiller']]

x_train.head(10)
y_train.head()
```

Model Building

AdaBoost

```
from sklearn.ensemble import AdaBoostClassifier
adb = AdaBoostClassifier()
adb.fit(x_train,y_train)
y_pred = adb.predict(x_train)
acc_adb = accuracy_score(y_train, y_pred) * 100
pre_adb = precision_score(y_train, y_pred, average='macro') * 100  # or 'micro', 'weighted'
recall_adb = recall_score(y_train, y_pred, average='macro') * 100  # or 'micro', 'weighted'
f1_adb = f1_score(y_train, y_pred, average='macro') * 100  # or 'micro', 'weighted'
# Print the evaluation metrics
print(f"Accuracy: {acc_adb}%")
print(f"Precision : {pre_adb}%")
print(f"Recall : {recall_adb}%")
print(f"F1 Score : {f1_adb}%")
# plot the confusion matrix
from sklearn.metrics import confusion_matrix
plt.figure(figsize=(20,6))

sns.heatmap(confusion_matrix(y_train, y_pred),annot=True,fmt='0.2f',annot_kws={'size':20})
from sklearn.metrics import roc_curve, auc
# Compute ROC curve and ROC area for each class
fpr, tpr, _ = roc_curve(y_train, y_pred)
roc_auc = auc(fpr, tpr)

# Plot ROC curve
plt.figure()
plt.plot(fpr, tpr, color='darkorange', lw=2, label='ROC curve (area = %0.2f)' % roc_auc)
plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])



plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('AUC & ROC CURVE')
plt.legend(loc="lower right")
plt.show()
```

Logistic Regression

```
from sklearn.linear_model import LogisticRegression
lr = LogisticRegression()
lr.fit(x_train, y_train)
y_pred = lr.predict(x_train)
acc_lr = accuracy_score(y_train, y_pred) * 100
pre_lr = precision_score(y_train, y_pred, average='macro') * 100  # or 'micro', 'weighted'
recall_lr = recall_score(y_train, y_pred, average='macro') * 100  # or 'micro', 'weighted'
f1_lr = f1_score(y_train, y_pred, average='macro') * 100  # or 'micro', 'weighted'
# Print the evaluation metrics
print(f"Accuracy: {acc_lr}%")
print(f"Precision : {pre_lr}%")
print(f"Recall : {recall_lr}%")
print(f"F1 Score : {f1_lr}%")
# plot the confusion matrix
from sklearn.metrics import confusion_matrix
plt.figure(figsize=(20,6))
sns.heatmap(confusion_matrix(y_train, y_pred),annot=True,fmt='0.2f',annot_kws={'size':20})
```

Stacking Classifier

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import AdaBoostClassifier
from sklearn.ensemble import StackingClassifier
# Base classifiers
base_classifiers = [
    ('rf', RandomForestClassifier(n_estimators=100, random_state=42)),
    ('adb', AdaBoostClassifier(random_state=42))

]
# Meta classifier
meta_classifier = LogisticRegression(random_state=42)
# Stacking classifier
STC = StackingClassifier(estimators=base_classifiers, final_estimator=meta_classifier)
# Train stacking classifier
```

```
STC.fit(x_train, y_train)
y_pred = STC.predict(x_train)
acc_stc = accuracy_score(y_train, y_pred) * 100
pre_stc = precision_score(y_train, y_pred, average='macro') * 100  # or 'micro', 'weighted'
recall_stc = recall_score(y_train, y_pred, average='macro') * 100  # or 'micro', 'weighted'
f1_stc = f1_score(y_train, y_pred, average='macro') * 100  # or 'micro', 'weighted'
# Print the evaluation metrics
print(f"Accuracy: {acc_stc}%")
print(f"Precision : {pre_stc}%")
print(f"Recall : {recall_stc}%")
print(f"F1 Score : {f1_stc}%")
```

XgBoost

```
import xgboost as xgb
boost = xgb.XGBClassifier(random_state=42)
# Training the model
boost.fit(x_train, y_train)
y_pred = boost.predict(x_train)
acc_xgb = accuracy_score(y_train, y_pred) * 100
pre_xgb = precision_score(y_train, y_pred, average='macro') * 100  # or 'micro', 'weighted'
recall_xgb = recall_score(y_train, y_pred, average='macro') * 100  # or 'micro', 'weighted'
f1_xgb = f1_score(y_train, y_pred, average='macro') * 100  # or 'micro', 'weighted'
# Print the evaluation metrics
print(f"Accuracy: {acc_xgb}%")
print(f"Precision : {pre_xgb}%")
print(f"Recall : {recall_xgb}%")
print(f"F1 Score : {f1_xgb}%")
```

# plot the confusion matrix

```
from sklearn.metrics import confusion_matrix

plt.figure(figsize=(20,6))
sns.heatmap(confusion_matrix(y_train, y_pred),annot=True,fmt='0.2f',annot_kws={'size':20})
```

Hypermeter Testing

Stacking Classifier

```
from sklearn.ensemble import StackingClassifier, RandomForestClassifier, AdaBoostClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import RandomizedSearchCV

# Define base estimators
base_estimators = [
    ('rf', RandomForestClassifier(random_state=42)),
    ('adb', AdaBoostClassifier(random_state=42))
]

# Define stacking classifier with final meta-classifier
STC = StackingClassifier(estimators=base_estimators, final_estimator=LogisticRegression())

param_grid = {
    'rf__n_estimators': [100, 200, 300],  # Expand range for RandomForestClassifier
    'rf__max_depth': [None, 10, 20, 30],  # Add max_depth for RandomForestClassifier

    'adb__n_estimators': [50, 100, 150],  # Expand range for AdaBoostClassifier
    'adb__learning_rate': [0.1, 0.5, 1.0],  # Add learning_rate for AdaBoostClassifier

    'final_estimator__C': [0.01, 0.1, 1.0],  # Expand range for LogisticRegression
    'final_estimator__solver': ['liblinear', 'lbfgs', 'sag'] # Add 'sag' solver for LogisticRegression
}

# Grid search CV
grid_search = RandomizedSearchCV(estimator=STC, param_distributions=param_grid, cv=5,
scoring='accuracy', verbose=1)

# Fit grid search
grid_search.fit(x_train, y_train)

# Best parameters and best score
print("Best parameters found: ", grid_search.best_params_)



# Base classifiers
base_classifiers = [
    ('rf', RandomForestClassifier(n_estimators=300, random_state=42, max_depth=300)),
    ('adb', AdaBoostClassifier(n_estimators=100, random_state=42, learning_rate=0.5))
]

# Meta classifier
meta_classifier = LogisticRegression(random_state=42, C=0.1, solver='liblinear')

# Stacking classifier
STC_hyp = StackingClassifier(estimators=base_classifiers, final_estimator=meta_classifier)

# Train stacking classifier
STC_hyp.fit(x_train, y_train)

y_pred = STC_hyp.predict(x_train)
acc_stc_hyp = accuracy_score(y_train, y_pred) * 100
pre_stc_hyp = precision_score(y_train, y_pred, average='macro') * 100  # or 'micro', 'weighted'
recall_stc_hyp = recall_score(y_train, y_pred, average='macro') * 100  # or 'micro', 'weighted'
f1_stc_hyp = f1_score(y_train, y_pred, average='macro') * 100  # or 'micro', 'weighted'
# Print the evaluation metrics
print(f"Accuracy: {acc_stc_hyp}%")
print(f"Precision : {pre_stc_hyp}%")
print(f"Recall : {recall_stc_hyp}%")
print(f"F1 Score : {f1_stc_hyp}%")

# plot the confusion matrix
from sklearn.metrics import confusion_matrix
plt.figure(figsize=(20,6))
sns.heatmap(confusion_matrix(y_train, y_pred),annot=True,fmt='0.2f',annot_kws={'size':20})
```

Accuracy Graph

```
import seaborn as sns
import matplotlib.pyplot as plt
Algorithm = ['AdaBoostClassifier', 'LogisticRegression', 'xgboost', 'StackingClassifier']
Accuracy = [acc_adb*100, acc_lr*100, acc_xgb*100, acc_stc*100]
plt.figure(figsize=(12,4))
sns.barplot(x = Algorithm, y = Accuracy)
plt.title("Testing Acccuracy Graph")
plt.show()
```

APPENDIX B

CONFERENCE PUBLICATION

APPENDIX C

PUBLICATION DETAILS

APPENDIX D

PLAGIARISM REPORT