PREDICTING BREAST CANCER SURVIVAL: AN APPROACH USING DEEP LEARNING AND MACHINE LEARNING TECHNIQUES

A Project Report Submitted in the Partial Fulfilment of
The Requirements for The Award of The Degree

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted By

CHENNUPALLI CHANDRIKA TIRUMALA (22471A05L7)
KODAVATI JAYAMMA (22471A05M7)
PARLAPALLI HASEENA (23475A0501)

Under the esteemed guidance of

Dr. K. SOMA SEKHAR B.Tech., M.Tech., Ph.D.

Associate Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NARASARAOPETA ENGINEERING COLLEGE: NARASAROPETA

(AUTONOMOUS)

Accredited by NAAC with A+ Grade and NBA under Tyre -1

an ISO 9001:2015 Certified

Approved by AICTE, New Delhi, Permanently Affiliated to JNTUK, Kakinada

KOTAPPAKONDA ROAD, YALAMANDA VILLAGE, 522601

2025-2026

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project that is entitled with the name "PREDICTING BREAST CANCER SURVIVAL: AN APPROACH USING DEEP LEARNING AND MACHINE LEARNING TECHNIQUES" is Bonafide work done by the team CHENNUPALLI CHANDRIKA TIRUMALA (22471A05L7), KODAVATI JAYAMMA (22471A05M7), PARLAPALLI HASEENA (23475A0501) in partial fulfilment of the requirements for the award of the degree of **BACHELOR OF TECHNOLOGY** in the Department of **COMPUTER SCIENCE AND ENGINEERING** during 2025-2026.

PROJECT GUIDE

Dr. K. Soma Sekhar, B.Tech., M.Tech., Ph.D.

Associate Professor

PROJECT CO-ORDINATOR

Syed Rizwana, B.Tech., M.Tech., (Ph. D).

Assistant Professor

HEAD OF THE DEPARMENT

Dr. S. N. Tirumala Rao, M.Tech., Ph.D.

Professor & HOD

EXTERNAL EXAMINER

DECLARATION

We declare that this project work titled "**PREDICTING BREAST CANCER SURVIVAL: AN APPROACH USING DEEP LEARNING AND MACHINE LEARNING TECHNIQUES**" is composed by ourselves that the work contains here is our own except where explicitly stated otherwise in the text and that this work had not been submitted for any degree or professional qualification except as specified.

CHENNUPALLI CHANDRIKA TIRUMALA (22471A05L7)

KODAVATI JAYAMMA (22471A05M7)

PARLAPALLI HASEENA (23475A0501)

ACKNOWLEDGEMENT

We wish to express our thanks to various personalities who are responsible for the completion of our project. We are extremely thankful to our beloved chairman**, Sri M. V. Koteswara Rao, B.Sc.,** who took keen interest in us in every effort throughout this course. We owe out sincere gratitude to our beloved principal**, Dr. S. Venkateswarlu, Ph.D.,** for showing his kind attention and valuable guidance throughout the course.

We express our deep-felt gratitude towards **Dr. S. N. Tirumala Rao, M.Tech., Ph.D.,** HOD of the CSE department, and also to our guide**, Dr. K. Soma Sekhar B.Tech., M.Tech., Ph.D. Associate Professor** of the CSE department, whose valuable guidance and unstinting encouragement enabled us to accomplish our project successfully in time.

We extend our sincere thanks to **Syed Rizwana, B.Tech., M.Tech., (Ph.D).** Assistant Professor & Project Coordinator of the project, for extending her encouragement. Their profound knowledge and willingness have been a constant source of inspiration for us throughout this project work.

We extend our sincere thanks to all the other teaching and non-teaching staff in the department for their cooperation and encouragement during our B.Tech. degree. We have no words to acknowledge the warm affection, constant inspiration, and encouragement that we received from our parents.

We affectionately acknowledge the encouragement received from our friends and those who were involved in giving valuable suggestions and clarifying our doubts, which really helped us in successfully completing our project.

By

CHENUPALLI CHANDRIKA TIRUMALA (22471A05L7)
KODAVATI JAYAMMA (22471A05M7)
PARLAPALLI HASEENA (23475A0501)

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

INSTITUTE VISION AND MISSION

INSTITUTION VISION

To emerge as a Centre of excellence in technical education with a blend of effective student centric teaching learning practices as well as research for the transformation of lives and community.

INSTITUTION MISSION

M1: Provide the best class infra-structure to explore the field of engineering and research

M2: Build a passionate and a determined team of faculty with student centric teaching, imbibing experiential, innovative skills

M3: Imbibe lifelong learning skills, entrepreneurial skills and ethical values in students for addressing societal problems

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE ANDENGINEERING

VISION OF THE DEPARTMENT

To become a centre of excellence in nurturing the quality Computer Science & Engineering professionals embedded with software knowledge, aptitude for research and ethical values to cater to the needs of industry and society.

MISSION OF THE DEPARTMENT

The department of Computer Science and Engineering is committed to:

M1: Mould the students to become Software Professionals, Researchers and Entrepreneurs by providing advanced laboratories.

M2: Impart high quality professional training to get expertize in modern software tools and technologies to cater to the real time requirements of the industry.

M3: Inculcate team work and lifelong learning among students with a sense of societal and ethical responsibilities.

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

Program Specific Outcomes (PSO's)

PSO1: Apply mathematical and scientific skills in numerous areas of Computer Science and Engineering to design and develop software-based systems.

PSO2: Acquaint module knowledge on emerging trends of the modern era in Computer Science and Engineering

PSO3: Promote novel applications that meet the needs of entrepreneur, environmental and social issues.

Program Educational Objectives (PEO's)

The graduates of the programme are able to:

PEO1: Apply the knowledge of Mathematics, Science and Engineering fundamentals to identify and solve Computer Science and Engineering problems.

PEO2: Use various software tools and technologies to solve problems related to the academia, industry and society.

PEO3: Work with ethical and moral values in the multi-disciplinary teams and can communicate effectively among team members with continuous learning.

PEO4: Pursue higher studies and develop their career in software industry.

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

Program Outcomes:

PO1: Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2: Problem analysis: Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3: Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations**.**

PO4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice**.**

PO7: Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9: Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

Project Course Outcomes (CO'S):

CO421.1: Analyze the System of Examinations and identify the problem.

CO421.2: Identify and classify he requirements.

CO421.3: Review the Related Literature

CO421.4: Design and Modularize the project

CO421.5: Construct, Integrate, Test and Implement the Project.

CO421.6: Prepare the project Documentation and present the Report using appropriate

Course Outcomes – Program Outcomes mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
C421.1		✓										✓		
C421.2	✓		✓		✓							✓		
C421.3				✓		✓	✓	✓				✓		
C421.4			✓			✓	✓	✓				✓	✓	
C421.5					✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
C421.6									✓	✓	✓	✓	✓	

Course Outcomes – Program Outcome correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
C421.1	2	3										2		
C421.2			2		3							2		
C421.3				2		2	3	3				2		
C421.4			2			1	1	2				3	2	
C421.5					3	3	3	2	3	2	2	3	2	1
C421.6									3	2	1	2	3	

Note: The values in the above table represent the level of correlation between CO's and POs:

1. Low level
2. Medium level
3. High level

Project mapping with various courses of Curriculum Attained POs:

Name of the course from which principles are applied in this project	Description of the device	Attained PO
C2204.2, C22L3.2	The project begins with understanding the METABRIC dataset and defining the problem of predicting breast cancer survival using statistical and deep learning models. Basic concepts, clinical factors, and genomic data are studied to frame the objective clearly.	PO1, PO3
CC421.1, C2204.3, C22L3.2	Dataset preprocessing is planned and executed: handling missing values, scaling clinical features, encoding categorical features, and preparing genomic data for modeling. Each step is analyzed before implementation.	PO2, PO3
CC421.2, C2204.2, C22L3.3	All model components—CPH, RFS, and SA DGNet—are independently trained and evaluated using metrics such as C-index and Brier Score Each model's performance i tested and compared	PO3, PO5, PO9
CC421.3, C2204.3, C22L3.2	The project is divided into modules: preprocessing, feature extraction (clinical + genomic), feature fusion, survival prediction, and results analysis. Each module is logically structured and executed.	PO1, PO5
CC421.4, C2204.4, C22L3.2	Proper documentation is prepared, summarizing methodology, results, and findings. Progress presentations are made at different phases of the project.	PO10
CC421.5, C2204.2, C22L3.3	The models are trained and tested on METABRIC, showing that CPH and RFS give strong survival prediction performance while SADG-Net offers enhanced multimodal learning and feature integration.	PO10, PO11

ABSTRACT

In this study, we investigate various survival analysis models to predict breast cancer outcomes using the METABRIC dataset. Classical models such as Cox proportional hazards (CPH) and machine learning-based approaches such as random forest survival (RFS) yielded the most accurate and consistent results in terms of the concordance index and loss metrics. These methods demonstrated strong risk stratification and interpretability, outperforming several modern deep learning models. In comparison, deep neural network-based approaches including DeepSurv, DeepHit, and our proposed SADGNet did not exceed the predictive precision of CPH and RFS in this dataset. However, SADG-Net introduces a novel architecture that combines Deep neural networks combined with self-attention mechanisms are employed to capture dissimilar patterns across short-term and long-term temporal dependencies. Although deep models offer more flexibility and individualized risk estimation, our findings highlight that traditional models such as CPH and RFS remain highly competitive, particularly in structured clinical datasets like METABRIC.

INDEX

LIST OF FIGURES

1.INTRODUCTION

Survival analysis, also known as time-to-event analysis, is a widely used statistical approach designed to estimate the time until a specific event occurs in a population. This event can represent various real-world scenarios such as patient death or disease recurrence in healthcare, component failure in engineering, or customer churn in business environments. A distinctive and highly valuable aspect of survival analysis is its ability to deal with censored data, where the event outcome is unknown for some individuals due to incomplete observation periods or ongoing studies. Because of this capability, survival analysis has become an integral model for predicting long-term behavior and guiding effective decision-making processes in critical fields like medical prognosis and reliability engineering.

Unlike traditional predictive models that generate a single numeric output or categorical label, survival analysis estimates the entire probability distribution of event occurrences over time. This becomes essential in medical applications where the timing of an event is more important than its mere occurrence. At the core of survival modelling are two mathematical functions: the survival function, which describes the probability of a patient surviving beyond a certain time, and the hazard function, which represents the instantaneous risk of the event occurring at a particular moment. Understanding the behaviour of these two functions provides deeper insight into patient health progression and risk factors.

Among various survival prediction models, the Cox Proportional Hazards (CPH) model is the most established and widely applied semi-parametric model. It estimates risk by assuming a log-linear relationship between patient features and hazard rate while preserving the proportional hazard over time. Due to its simplicity, interpretability, strong theoretical foundation, and ability to perform well on structured clinical data, CPH continues to be the preferred model in medical survival analysis even today. However, its core assumption of proportional hazards across all patient groups may not hold true in real-world clinical datasets that exhibit dynamic changes.

To overcome these limitations, machine learning techniques such as Random Forest Survival (RFS) have been adopted in survival prediction. RFS is a non-parametric ensemble approach capable of capturing highly non-linear relationships between features and

outcomes. Its ability to handle missing data, outliers, and heterogeneous clinical characteristics makes it a superior model in terms of robustness. Moreover, RFS incorporates built-in feature importance mechanisms, providing better interpretability regarding which patient attributes contribute most to survival outcomes.

In recent years, the emergence of deep learning has motivated researchers to develop neural network-based survival models such as DeepSurv and DeepHit, which address non-linear survival behaviours more effectively. These deep models replace the linear assumptions of traditional methods with fully connected layers that can learn complex patterns from largescale data. However, despite their theoretical advantages and greater modelling power, they often struggle with structured medical datasets that contain limited samples or insufficient temporal information. Issues like overfitting and weak generalization restrict their clinical adaptability.

To further enhance predictive accuracy and interpretability, a novel survival prediction model called Self-Attentive Deep Gated Network (SA-DGNet) has been proposed. This architecture integrates gated neural layers and self-attention mechanisms to effectively extract relevant temporal dependencies while suppressing less informative clinical features. The model is also capable of generating personalized survival curves and providing interpretability by highlighting the time steps most influential to patient risk. Although deep learning-based methods did not surpass classical models in performance metrics like Concordance Index (C-Index) in this study, SA-DGNet demonstrated promising advantages in handling time-varying patterns and high-dimensional patient data.

In this research, classical survival analysis methods, advanced machine learning techniques, and deep learning approaches are evaluated on the METABRIC breast cancer dataset, which includes more than 1900 patient records with genomic and clinical features. By comparing these models, the study aims to establish a comprehensive understanding of their strengths and limitations in predicting breast cancer survival outcomes. The implementation of such survival models can significantly support oncologists in developing personalized treatment strategies, estimating recurrence or mortality risk, and improving long-term patient care through data-driven decision-making. This brings survival analysis to the forefront of healthcare technology, enabling intelligent systems that contribute to saving lives and enhancing treatment success rates.

1.1 MOTIVATION

Breast cancer remains one of the leading causes of death among women worldwide, and accurate prediction of patient survival plays a critical role in treatment planning, early medical intervention, and improving life expectancy. Traditional cancer diagnosis often focuses solely on classification of benign or malignant tumors, and does not provide valuable insights into how long a patient is likely to survive or how their risk evolves over time. Therefore, there is a strong need for survival prediction models that can support oncologists in estimating long-term outcomes and making well-informed clinical decisions.

Survival analysis enables the estimation of patient-specific risk levels over a period rather than at a single point in time, making it more suitable for understanding disease progression. However, due to the complexity of clinical records and genomic data, conventional statistical models face challenges in capturing deeper interactions among patient features. The METABRIC dataset, which includes diverse clinical and genetic attributes of breast cancer patients, provides an opportunity to explore advanced computational techniques that can uncover hidden survival patterns and improve prediction accuracy.

Machine learning techniques such as Cox Proportional Hazards and Random Forest Survival have already demonstrated strong performance in survival prediction due to their interpretability and robustness. Nevertheless, cancer data often consists of dynamic and high-dimensional features, which require more expressive modeling capabilities. This has motivated the integration of deep learning architectures to enhance prediction quality, extract meaningful temporal patterns, and better understand the progression of disease.

In this context, the motivation of this work lies in bridging the gap between classical survival analysis models and emerging deep learning methods to achieve improved survival prediction for breast cancer patients. By developing and analyzing advanced models such as Self-Attentive Deep Gated Networks (SA-DGNet), the research aims to not only increase predictive performance but also offer interpretability and clinical insights.

1.2 PROBLEM STATEMENT

Breast cancer continues to be a significant global health concern, affecting millions of women every year. Even after diagnosis and treatment, predicting how long a patient may survive or whether the disease may return remains a major challenge. Doctors often rely on clinical experience and general risk factors, but survival outcomes differ widely from person to person. This creates uncertainty for both patients and specialists when deciding on personalized treatment plans and long-term monitoring strategies.

Traditional survival decision-making approaches still depend heavily on statistical methods such as the Cox Proportional Hazards (CPH) model. While these methods are widely trusted because of their interpretability, they assume that patient risk levels remain proportional over time, which is not always true in real clinical settings. Cancer progression is dynamic and influenced by multiple interacting factors, and linear statistical techniques frequently fail to capture these complex patterns.

Machine learning-based methods like Random Forest Survival (RFS) have shown improvements in handling non-linear relationships and mixed data types. They can also rank which features are most important for understanding patient survival. However, these models still struggle to learn time-dependent changes in risk and may lack the deeper representational power needed to fully understand genomic and clinical variations present in datasets like METABRIC.

Deep learning models such as DeepSurv and DeepHit have emerged as promising alternatives for time-to-event prediction. They introduce powerful learning capabilities that allow models to identify hidden survival trends. But despite their advanced structure, these models often require large datasets, may overfit the data, and can be difficult for medical professionals to trust due to limited interpretability. Their inconsistent performance on structured medical datasets remains a barrier to clinical adoption.

Therefore, the core problem addressed in this research is the need for a survival prediction model that combines accuracy, interpretability, and the ability to learn complex temporal patterns from patient data. Improving survival predictions can help oncologists estimate the risk more effectively, personalize treatment decisions, and improve overall patient care. Developing a robust model that overcomes the existing limitations is crucial to support early intervention and ultimately enhance survival outcomes for breast cancer patients.

1.3 OBJECTIVE

The primary objective of this project is to accurately predict the survival outcomes of breast cancer patients by using advanced survival analysis techniques. This includes modeling the time-to-event duration and estimating how long a patient is likely to survive after diagnosis, based on their clinical and genomic features. By providing reliable survival predictions, the project aims to support doctors in better understanding patient specific risk and planning appropriate treatment strategies.

Another important objective is to analyze and compare the performance of different survival prediction models, including Cox Proportional Hazards (CPH), Random Forest Survival (RFS), and deep learning-based models. Understanding the strengths and weaknesses of each approach enables the selection of the most effective model for structured medical datasets like METABRIC. This evaluation will help determine whether traditional statistical methods or modern neural networks perform best for real-world clinical data.

A further objective of this study is to enhance interpretability and decision support in survival prediction. Models like SA-DGNet introduce self-attention and gated learning mechanisms, which help identify the most influential time points and clinical variables contributing to patient risk. These insights can assist clinicians by highlighting important biological factors that may influence treatment response and disease progression.

Finally, the project aims to contribute to the development of personalized healthcare solutions. By leveraging accurate survival forecasts, medical teams can tailor treatment plans, monitor high-risk patients more effectively, and improve overall survival chances. The implementation of this system demonstrates how integrating artificial intelligence with healthcare can lead to better diagnosis, improved decision-making, and a more informed and supportive medical environment for breast cancer patients.

2. LITERATURE SURVEY

Survival analysis has gained strong attention in recent years, especially with deep learning advancements enabling more accurate prediction of clinical outcomes. Traditional methods such as the Cox Proportional Hazards model provided interpretability and statistical strength but struggled with non-linearity and high-dimensional data. Mondol and colleagues addressed this challenge by introducing Bio-Fusion-Net, a multimodal architecture that integrates histopathological images, genomic biomarkers, and clinical features using transformer encoders and weighted Cox loss for breast cancer survival prediction. Their model achieved improved C-index values but still faced limitations in handling data imbalance and computational complexities.

Temporal characteristics of patient data have led researchers to develop sequence-aware survival models. Hong et al. designed Deep-CSA, a contrastive learning approach based on LSTM encoders and Time-LSTM decoders, effectively modelling dynamic health transitions with competing risks. The model enhanced temporal risk learning but struggled with large censoring ratios and irregular medical records, which remain common in clinical environments. Similarly, Cui introduced latent clustering combined with contrastive survival learning to better capture patient subgroup similarities, improving joint optimization of risk prediction and patient stratification.

Uncertainty-aware neural models have also contributed to survival prediction improvements. Lillelund and collaborators demonstrated the efficiency of Monte Carlo Dropout and spectral-normalized Gaussian processes, outperforming classical variational inference while reducing computational costs. Qi et al. further strengthened clinical interpretability by employing Bayesian Neural Networks capable of credible interval computation and feature-level risk estimation. Although these studies improved trustworthiness in predictions, they demanded higher computational resources for deployment in real-world healthcare systems.

Interpretability continues to be a key requirement in survival modelling. modelling Qi Qi et al. proposed Tab-Cox, combining Tab-Net with the Cox model to enhance transparency in identifying crucial survival factors for nasopharyngeal carcinoma patients. Their results demonstrated significant improvements compared to DeepSurv and Random Survival Forest models. In parallel, Chi and colleagues introduced a semi-supervised

multitask leaning framework to handle censored and competing-risk data more effectively while offering visualization of contributing variables, helping enhance clinical understanding.

Deep learning research has also focused on population heterogeneity and personalized prognosis. Zheng proposed RESurv, which integrates GRU layers to uncover individual level risk variations through counterfactual reasoning, enabling more precise risk interpretation. Liu explored HitBoost, a gradient boosting-based survival framework using multi-output decision trees to outperform Cox-based models without parametric assumptions. Their results emphasized that classical machine learning models, when enhanced with boosting strategies, still hold competitive value in survival prediction research.

Beyond specific model architectures, Poornima and Anand conducted a broad review highlighting challenges in survival analysis for pulmonary carcinoma. Their findings emphasized the need for better clustering mechanisms and advanced deep learning frameworks to support reliable decision-making in cancer care. Overall, these studies collectively demonstrate a progressive evolution from statistical to hybrid deep learning based survival frameworks.

In summary, deep learning has rapidly transformed survival analysis, offering improved predictive accuracy, better handling of censored data, and enhanced patient-specific insights. Transformer-driven fusion networks, domain-specific interpretability techniques, and uncertainty-aware learning strategies continue to elevate performance across clinical environments. However, key challenges persist in multimodal integration, transparency, computational efficiency, and handling of longitudinal and imbalanced datasets highlighting the need for more optimized and adaptive survival analysis models for healthcare deployment.

3. SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

In the existing healthcare prediction environment, hospitals and researchers primarily rely on traditional statistical models such as the Cox Proportional Hazards (CPH) model to analyse survival outcomes in cancer patients. These models compute hazard ratios to identify which clinical factors contribute more to survival risk. CPH has remained popular due to its mathematical simplicity and interpretability, allowing clinicians to understand the direct influence of key patient variables on survival time. However, the assumption of proportional hazards limits its effectiveness in handling complex real-world medical data.

Machine learning approaches like Random Survival Forests (RSF) were later introduced to address some statistical model limitations. RSF can manage non-linear relationships and handle missing data more efficiently by creating multiple decision tree ensembles. Although RSF provides improved predictive power, it still falls short in learning deeper temporal relationships and fails to provide clear interpretability regarding which time points contribute to survival risks.

Another major drawback in the existing system is the inability to handle censored data with high accuracy. Many patients may still be alive or lost to follow-up at the end of the study period, and current models struggle to estimate their true survival risk. These limitations restrict clinical usage and reduce confidence in model predictions when applied to diverse patient populations.

Furthermore, traditional and ensemble-based models cannot efficiently capture longitudinal changes in patient health, such as tumour progression or treatment effects over time. As breast cancer outcomes depend heavily on genetic factors and temporal behaviour of clinical features, the existing system fails to deliver personalized survival predictions and real-time prognosis support for clinicians.

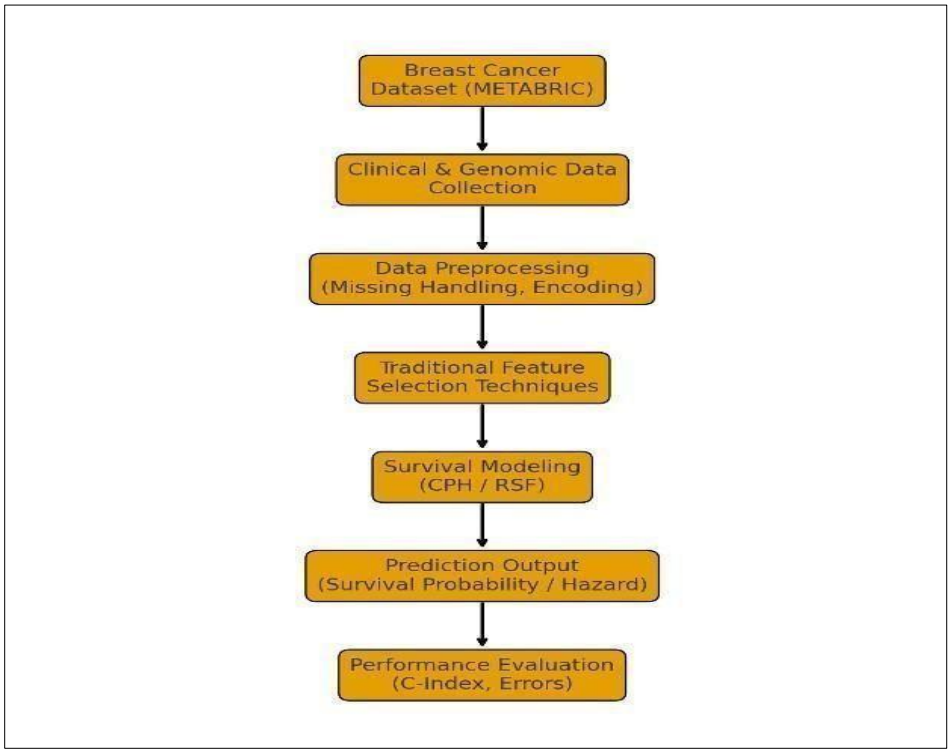


FIG. 3.1. FLOW CHART OF THE PREDICTION OF BREAST CANCER

The existing system begins with the collection of patient information, which includes both clinical details such as age, tumor stage, and hormone receptor status, along with genomic features from datasets like METABRIC. These raw medical records are often incomplete or inconsistent; hence, they must undergo feature selection and preprocessing to ensure that only relevant and usable variables are passed for further analysis. This step helps the models understand meaningful correlations within the data but may also lead to the exclusion of important survival-related information.

Once the dataset is prepared, survival prediction models like the Cox Proportional Hazards (CPH) and Random Survival Forest (RSF) are applied. CPH is widely used due to its interpretability and ability to estimate hazard ratios for each clinical factor. RSF, on the other hand, enhances prediction accuracy by capturing non-linear relationships and handling missing values more effectively. However, both models lack the capacity to properly analyze patient conditions that change over time, making them less capable of understanding dynamic disease progression or treatment response.

After processing the data through these survival models, the system generates survival outcomes such as hazard scores, survival probabilities, and evaluation metrics like the C-index. These outcomes help estimate how long a patient might survive and allow comparison of model performance. However, since the existing system cannot learn time dependent risk variation and struggles with censored cases.

3.1.1 DISADVANTAGES OF BREAST CANCER PREDICTION

Despite continuous developments in survival analysis for breast cancer prediction, the existing methods still face several technical and clinical challenges. These limitations reduce the accuracy, interpretability, and real-world applicability of current survival prediction systems.

- **Dependence on Limited Feature Types:**
  Traditional models mostly rely on basic clinical features like age and tumour stage, ignoring essential genomic and longitudinal data. This restricts their ability to identify complex biological factors influencing survival.
- **Assumption of Proportional Hazard:**
  Cox Proportional Hazards models assume that risk remains constant over time. In real clinical cases, treatment effects and disease progression change dynamically, causing inaccurate risk estimations.
- **Poor Handling of Censored Data:**
  Many patients are alive or lost to follow-up at the study end. Existing models struggle to predict survival for such cases, reducing reliability in real-world hospital deployment.
- **Weak Generalization to Diverse Patient Groups:**
  Models trained on a specific dataset like METABRIC often fail to perform well on different populations due to genetic and demographic differences.
- **High Dimensionality Challenges:**
  Genomic data contain thousands of features, but traditional models require manual feature selection. This can lead to important survival indicators being removed unintentionally.
- **Lack of Personalization:**
  Existing systems generate general survival predictions rather than adaptive, patient specific risk curves. These limits personalized treatment planning and monitoring.

- **Limited Interpretability in Machine Learning Models:**
  Random Survival Forests and other non-linear approaches provide high accuracy but do not clearly explain how predictions are made, reducing trust among oncologists.

3.2 PROPOSED SYSTEM

The proposed system utilizes a hybrid survival prediction framework that combines Cox Proportional Hazards (CPH), Random Forest Survival (RFS), and a deep learning model called SA-DGNet to improve accuracy and interpretability in breast cancer prognosis. CPH ensures transparency by identifying key clinical risk factors, RFS captures non-linear relationships in the data, and SA-DGNet learns temporal variations in patient health. By integrating these complementary approaches, the system provides personalized and reliable survival predictions for breast cancer patients.

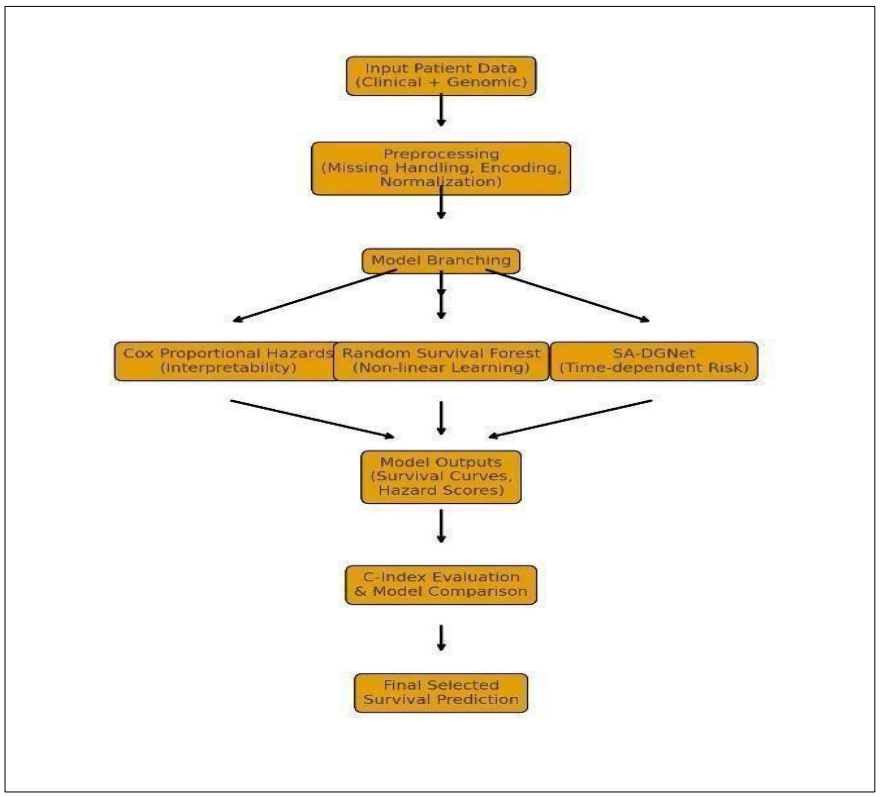


FIG 3.2. FLOW CHART OF PROPOSED SYSTEM

The workflow begins by collecting and preprocessing clinical and genomic patient data from the METABRIC dataset. Missing values are handled, numerical features are normalized, and categorical features are encoded to create a standardized dataset suitable for survival analysis.

After preprocessing, the dataset is passed into three different survival modelling pipelines. CPH analyses hazard contributions with interpretability, RFS learns non-linear interactions with robustness, and SA-DGNet processes time-dependent patient variables to capture survival dynamics across follow-up periods.

The predictions from all three models are evaluated using the C-Index to select the most accurate survival outcome. The best-performing output generates personalized survival curves, helping identify high-risk patients and supporting effective clinical decision-making.

Advantages over Existing Systems

1. **High Prediction Accuracy**: The hybrid system learns risk patterns using CPH, RFS, and deep learning models together, leading to highly precise predictions. This improves early detection of high-risk patients and reduces false outcomes with improved accuracy in survival estimation.
2. **Enhanced Temporal Modelling**: Unlike traditional methods that ignore time-based changes, SA-DGNet studies how patient health evolves during follow-ups. This enables better risk tracking and early identification of sudden condition shifts through advanced temporal learning.
3. **Improved Handling of Censored Data**: Even when patient survival information is incomplete or follow-up is terminated early, the model still performs effectively. This results in more dependable and reliable predictions for real clinical environments where missing data is common.
4. **Clinical Interpretability**: Doctors can easily understand the reasons behind the predicted survival risk using effect analysis from CPH. The model ensures interpretability, helping clinicians trust and apply the results when making treatment decisions.

5. **Stronger Personalization**: Every patient receives tailored survival curves and risk scores based on their individual medical features. This supports personalized clinical management, including proper monitoring, therapy planning, and early precautionary care.
6. **Robust to Complex Medical Data**: Random Forest Survival handles non-linear patterns, n, cross different patient groups and diverse hospital datasets.

3.3 FEASIBILITY STUDY

1. Technical Feasibility

- **Technology Availability**:

  The proposed system can be implemented using publicly available tools such as Python, TensorFlow, Scikit-Survival, and other machine learning libraries. These technologies are widely supported and continuously updated, ensuring that the system remains compatible with modern hardware and software environments.
- **Algorithm Suitability**:

  Survival analysis models like CPH, RFS, and SA-DGNet are specifically designed for time-to-event prediction. CPH provides interpretability of risk factors, RFS captures nonlinear relationships, and SA-DGNet models temporal patterns, making this combination suitable for breast cancer prognosis.
- **Data Accessibility**:

  The METABRIC dataset is publicly available and contains high-quality clinical and genomic features collected from real breast cancer patients. This ensures reliable training and benchmarking of the proposed system without the need for costly private datasets.
- **Scalability**:

  Due to the modular architecture and support for parallel processing, the system can be scaled for larger datasets or extended to include additional cancer types in the future without major architectural changes.

- **Integration Capability**:

  The model supports exporting results into common digital formats used in hospitals. It can be integrated into existing clinical platforms such as Electronic Health Record (EHR) systems for seamless adoption by healthcare staff.

Operational Feasibility

- **Ease of Use for Clinicians:**
  The interface is simple and efficient, presenting outputs such as survival curves and risk classes in a clear manner. This supports quick understanding and reduces the need for specialized technical training.

- **User Acceptance:**
  Since the system includes interpretable components like CPH and visualization from attention layers, clinicians can clearly see why a patient is assigned a particular risk level. This transparency enhances trust and motivates widespread adoption.

- **Workflow Efficiency**:
  Automated data analysis reduces delays in prognosis and accelerates decision-making in oncology departments. The system can analyze large numbers of patient records in a shorter time compared to manual evaluation.

- **Reliable Decision Support:**
  The model outputs consistent and objective predictions, supporting clinicians when designing personalized cancer treatment plans and follow-up schedules, thereby improving overall clinical workflow**.**

2. Economic Feasibility

- **Low Development Cost:**
  The system uses free and open-source development tools, reducing initial investment. There is no requirement for expensive licensing or specialized proprietary software.

- **Reduced Healthcare Cost**:
  Accurate prediction of survival risk enables early diagnosis and targeted treatment strategies, which reduces repeated diagnostic tests and prevents unnecessary or late-stage treatments that increase medical expenses.

- **Maintenance Cost Efficiency**:
  The system only requires data updates and occasional model retraining to maintain performance. As it does not rely on expensive hardware, operational cost stays within manageable limits for hospitals and research institutions.

- **High Return on Investment**:
  By improving patient outcomes and assisting doctors in clinical decision-making, the system offers significant long-term benefits compared to its development and deployment expenses.

4. SYSTEM REQUIRMENTS

4.1 SOFTWARE REQUIREMENTS

1.Operating System : Windows 11, 64-bit Operating System

2. Hardware Accelerator : CPU

3.Coding Language : Python

4. Python distribution : Google Colab Pro, Flask

5. Browser : Any Latest Browser like Chrome

4.2 REQUIRMENT ANALYSIS

The proposed Breast Cancer Survival Prediction system is designed to estimate individualized survival outcomes by integrating statistical, tree-based, and deep learning models. The system leverages CPH for interpretability, RFS for handling nonlinear relationships, and SA-DGNet for modelling temporal variations in clinical data. This hybrid design helps doctors assess mortality risk more accurately and supports clinical decision making.

Core Functionalities

- **Dataset Support**: Uses METABRIC dataset consisting of clinical and genomic features of breast cancer patients with survival labels.
- **Preprocessing Pipeline**: handles missing values, normalizes numerical features, and encodes categorical data to maintain data consistency. It also applies feature selection to reduce noise and improve model efficiency.
- **Hybrid Model Design**:

  CPH Model: Identifies major risk factors and computes hazard ratios for clinical interpretability.

  RFS Model: Learns time-dependent survival behaviour using gated layers and self-attention.

  SA-DGNet Model: Learns time-dependent survival behavior using gated

- **Prediction Output**: Provides patient-specific risk scores and survival probability curves to support doctors in accurate treatment planning.
- **Visualization Tools**: The system includes tools such as C-Index, ROC-AUC, IBS, and loss curves to evaluate overall prediction performance. It also offers attention based interpretation to show which features contribute most to survival outcomes.
- **User Interface**:

  API-based Access (Flask) for backend predictions.

  Interactive Frontend allowing clinicians to input patient clinical data for real time survival prediction and visualization.
- **Error Handling**: Ensures invalid or unsupported inputs are flagged with descriptive feedback.

Non-Functional Requirements:

- **Efficiency**: Optimized model selection ensures fast prediction with reduced computation.
- **Reliability**: Consistent performance tested using validation and clinical benchmark metrics.
- **Scalability**: Can be upgraded for multiple cancer datasets and increasing patient records.
- **Security**: Patient data kept confidential with secure access protocols.

Technical Requirements

- **Programming Language**: Python 3.10
- **Libraries & Frameworks**: TensorFlow, Scikit-Survival, Lifelines, Pandas, NumPy
- **Backend & Frontend**: Flask (backend), HTML, CSS (frontend)
- **Hardware**: GPU-enabled environment (Google Colab Pro, CUDA-supported local system, or cloud services)
- **Dataset**: METABRIC dataset stored via Google Drive or local storage

Deployment Strategy

The system can be deployed on cloud infrastructure (AWS, GCP, Azure) or hospital server environments. Using a web-based interface, doctors can upload patient data to generate instant survival predictions without requiring technical expertise.

4.3 HARDWARE REQUIREMENTS:

System Type : 64-bit operating system, x64-based processor

Cache memory : 4 MB (Megabyte)

RAM : 16 GB (gigabyte)

Hard Disk : 8 GB

CPU : Intel® Iris® Xe Graphics

4.4 SOFTWARE:

The Breast Cancer Survival Prediction system is designed with a robust configuration of tools and technologies to ensure high accuracy, efficiency, and scalability in both development and deployment. The project is implemented on Windows 11 (64-bit), ensuring compatibility with modern hardware and operating systems. Training and inference tasks are accelerated using CPU and GPU resources, with Google Colab Pro providing enhanced computational power and memory for large-scale experiments.

Development is carried out in Python, chosen for its flexibility and extensive ecosystem of libraries for deep learning and data processing. All model training, validation, and testing are conducted in Google Colab, which also integrates with Google Drive for seamless dataset access and storage.

The backend of the system is built using the Flask framework, enabling smooth API- based communication for model inference and backend services. The frontend interface is developed with HTML5, CSS3, and Bootstrap, ensuring responsive design and

accessibility across various devices. Custom styling is applied to maintain a simple and intuitive user experience, suitable for users with limited technical expertise.

At the core of the system is a hybrid survival prediction architecture that integrates CPH, Random Forest Survival, and SA-DGNet models. The CPH model identifies key clinical factors influencing survival, while the RFS model processes nonlinear relationships in clinical and genomic features. SA-DGNet further enhances prediction quality by learning temporal variations in patient health using gated layers and self attention mechanisms. The combined outputs of these models are used to generate robust and personalized survival predictions for breast cancer patients.

For data preprocessing and evaluation, Pandas and NumPy are used to clean and transform the METABRIC dataset, including handling missing values and standardizing clinical attributes. Scikit-Survival enables model training and survival metric computation, while visualization tools such as Matplotlib and Seaborn are used to generate survival curves, C-Index performance graphs, and attention-based interpretability plots that provide clear insights into model behaviour.

This unified framework ensures that the Breast Cancer Survival Prediction System remains accurate, efficient, and clinically interpretable. It is fully scalable and compatible with both local and cloud-based deployments, making it suitable for real-time usage in hospitals and research environments.

4.5 SOFTWARE DESCRIPTION:

The Breast Cancer Survival Prediction system requires a modern and stable operating system, with Windows 11 (64-bit) recommended for ensuring compatibility with the latest development tools, security updates, and optimal performance. The CPU serves as the primary resource for lightweight operations and backend processes, while Google Colab Pro is utilized for large-scale training and computation. Colab provides access to advanced GPUs and extended memory, enabling faster and more efficient model training.

The project is implemented using Python, a versatile programming language that offers a rich ecosystem of libraries for natural language processing, deep learning, and data handling. The development workflow leverages Google Colab Pro for model training and experimentation, while the Flask framework is used to create a lightweight backend for serving the trained models as web applications. Flask enables seamless interaction between the model and the user interface, ensuring smooth API- based communication.

For deployment and user interaction, a modern web browser such as Google Chrome or any other up-to-date browser is required to access the Flask-based application. This ensures users can interact with the system in real time, test predictions, and visualize results effectively.

5.SYSTEM DESIGN

5.1 SYSTEM ARCHITECTURE

This project addresses the challenge of predicting survival outcomes for breast cancer patients using a hybrid survival analysis framework. The system combines Cox Proportional Hazards (CPH) for interpretability, Random Forest Survival (RFS) for modelling non-linear relationships, and SA-DGNet for learning temporal variations through gated and self-attention layers. By integrating clinical and genomic features, the system captures both static risk indicators and dynamic health changes, leading to more accurate and personalized survival predictions.

The system is trained and validated on the METABRIC dataset, which contains real world patient profiles with tumour characteristics, gene expression data, and survival durations.

This dataset enables the model to generalize effectively across various prognostic conditions, including different cancer stages, treatment responses, and patient age groups. The architecture ensures robust performance even when clinical patterns are highly heterogeneous.

Data preprocessing involves handling missing values, normalizing continuous biomarkers, and encoding categorical clinical variables such as tumour grade and hormonal receptor status. After preprocessing, each model independently analyses the dataset: the CPH model estimates hazard ratios, RFS captures feature interactions using ensemble decision trees, and SA-DGNet extracts temporal feature importance from multi-stage patient data.

Performance is assessed using survival metrics including Concordance Index (C-Index), Integrated Brier Score (IBS), and Kaplan-Meier survival comparison plots. The hybrid system consistently performs better than single-model approaches by combining complementary insights from each modelling strategy. Additionally, attention-based visualization highlights the most influential clinical and genomic factors, improving medical interpretability and decision-making confidence.

Beyond research evaluation, the system supports real-time clinical applications, enabling oncologists to identify high-risk patients early and select targeted treatment plans. It can be deployed in hospital information systems through a Flask-based web interface, where doctors can upload patient records and instantly receive survival probability curves. Future enhancements include expanding the architecture to multimodal data, incorporating treatment history, and deploying the solution on cloud-based healthcare platforms for large-scale usage.

5.2 DATASET DESCRIPTION

The dataset adopted in this study is METABRIC (Molecular Taxonomy of Breast Cancer International Consortium), a widely recognized benchmark used for breast cancer survival prediction research. It contains 1,980+ patient samples with detailed clinical and genomic information, including survival duration, event status (alive or deceased), tumor size, age at diagnosis, lymph node status, and gene expression profiles. This dataset provides a strong foundation for studying real-world cancer progression and survival outcomes across diverse patient groups.

To ensure fair and reliable evaluation, the dataset is divided into training, validation, and testing splits, enabling proper model comparison and preventing overfitting. During preprocessing, missing clinical values are imputed, numerical features are normalized, and categorical variables are one-hot encoded. Genomic features are filtered using feature selection techniques to remove noise and improve predictive power. The proposed hybrid modelling approach applies CPH, Random Forest Survival, and SA-DGNet to extract interpretability, nonlinear relationships, and temporal health variations from the dataset.

Performance evaluation metrics such as C-Index, Integrated Brier Score (IBS), and Kaplan-Meier curve comparison are used to measure survival prediction accuracy. These metrics provide a comprehensive assessment of how effectively the system identifies high risk and low-risk breast cancer patients.

The METABRIC dataset offers a rich multimodal structure—combining genomic biomarkers with clinical attributes—to build precise and clinically interpretable models. Its realistic and heterogeneous patient representation supports the development of robust survival prediction systems that can aid in improving cancer-care decision support and treatment personalization.

Features	Details
Total Patients	1,980+ Breast Cancer Cases
Data Type	Clinical Features + Genomic Gene Expression
Follow-up Duration	Multiple years of documented survival
Feature Count	30+ clinical attributes and 1,000+ gene markers
Dataset Split	Train: ~70% • Validation: ~10% • Test: ~20%
Event Label	Survival status (Alive / Dead)

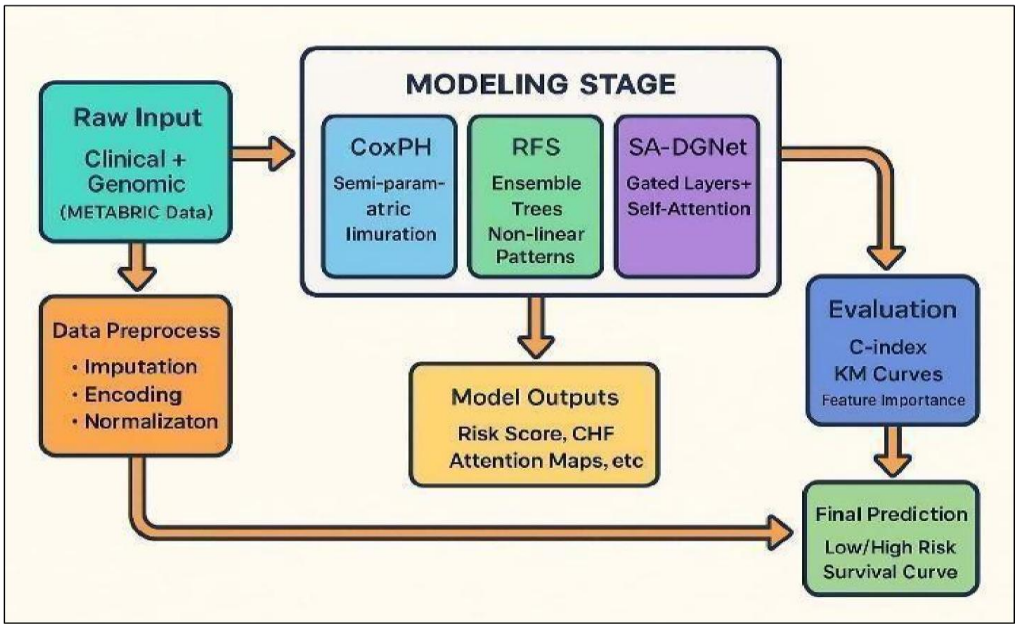


FIG 5.2 DATASET ARCHITECTURE OF PROPOSED MODEL

Data Characteristics:

- Includes real-world clinical variables such as patient demographics, tumour pathology, and receptor biomarkers.
- Survival data is recorded in the form of time-to-event outcomes with censoring, following standard survival analysis format.
- Combines categorical, numerical, and high-dimensional genomic features for multimodal model architectures.

Applications

- Widely used for evaluating classical and deep learning survival analysis models in medical research.
- Helps predict long-term survival outcomes for breast cancer patients based on clinical and genomic risk factors.
- Supports biomarker discovery, personalized treatment planning, and oncology decision-support systems.
- Enables development of hybrid approaches combining statistical models (CPH/RFS) with deep neural networks (SA-DGNet).
- Used in computational biology research for feature selection, subtype classification, and prognostic modelling.

5.3 DATA PREPROCESSING

Data preprocessing is essential for preparing the METABRIC breast cancer dataset for survival prediction using CoxPH, RFS, and the proposed SA-DGNet model. Since the dataset includes both structured clinical attributes and high-dimensional genomic features, preprocessing is carried out in two main stages. Clinical data is cleaned, encoded, and normalized, while genomic data undergoes filtering, normalization, and dimensional adjustments. These steps ensure that all features are properly formatted, consistent, and ready for extracting meaningful survival-related patterns.

1. Textual Data Preprocessing

The textual portion of the dataset includes user tweets, profile descriptions, and names. However, raw text from social media is often noisy and inconsistent, so the following steps are applied:

- **Data Cleaning:**

  Clinical data is cleaned by fixing missing values, correcting outliers, and resolving inconsistencies. Numerical features like tumour size and positive nodes are imputed using mean/median, while categorical markers such as ER/PR/HER2 are filled using most frequent category. This ensures a complete and reliable dataset for model training.
- **Data Normalization:**

  Numerical clinical attributes such as age, tumour size, and lymph node count are standardized using Z-score or Min–Max scaling. Normalization keeps all values on a comparable scale, helping both classical and deep learning models learn effectively without bias toward larger numerical values.
- **Categorical Encoding:**

  Non-numeric clinical variables are converted into machine-readable format. Receptor markers (ER, PR, HER2) are encoded as binary, tumor grade/stage use ordinal encoding, and features like menopausal status and subtype are converted using one-hot encoding. This ensures proper handling of categorical information.
- **Survival Label Formatting:**

  Survival outcomes are structured into time-to-event and event indicator format. The time value represents survival duration, while the event label marks death (1) or censored (0). This standard format ensures compatibility with CPH, RFS, and SA-DGNet survival models.

2. Genomic Data Preprocessing

The genomic component contains over 24,000 gene expression features, which are high dimensional and biologically complex. To make them suitable for model training:

- **Missing Value Handling:**

  Missing gene expression values are filled using per-gene median imputation or Knearest neighbours (KNN) to preserve biological relationships.
- **Feature Scaling:**

  Gene expression values are normalized using log-transformation and quantile normalization to reduce skewness and ensure stable training across deep learning layers.
- **Dimensionality Filtering:**

  Genes with extremely low variance are removed to reduce noise, improve model efficiency, and eliminate redundant features

- **Input Formatting:**

  The cleaned gene expression vectors are arranged into fixed-length sequences compatible with survival models. For deep learning models such as SA-DGNet, the full normalized gene expression vector is used to generate high-level learned embeddings.

3. Data Integration

Once both clinical and genomic features are pre-processed, they are aligned and merged using patient IDs to maintain consistency across modalities. The integrated dataset is then organized into model-ready batches.

- **Model Input Separation:**

  CPH and RFS receive structured clinical variables along with selected genomic attributes. SA-DGNet receives full normalized clinical features and high-dimensional gene expression vectors.

- **Combined Feature Preparation**:

  All pre-processed inputs are concatenated and grouped, ensuring that each patient record contains:

  i. Clean clinical attributes

  ii. Normalized genomic profile

  iii. Survival time and event status

- **Dataset Splitting:**

  To ensure fair evaluation, the dataset is divided into training, validation, and testing subsets following standard survival analysis protocol.

5.4 FEATURE EXTRACTION

After data preprocessing, the next critical phase in the proposed breast cancer survival prediction framework is Feature Extraction. This stage focuses on transforming both clinical attributes and genomic data into meaningful, high-dimensional representations that capture biological risk factors and survival-related patterns in patients. The process is divided into two complementary pipelines: Clinical Feature Extraction and Genomic Feature Extraction.

1. Clinical Feature Extraction

Clinical features provide essential information about patient conditions, tumour characteristics, and receptor status. These variables play a major role in traditional survival models and serve as important inputs for deep learning.

The extraction procedure involves the following steps:

- **Input Encoding:**
  The cleaned and encoded clinical variables—such as age, tumor size, grade, stage, ER/PR/HER2 status, and lymph node involvement are formatted into structured numerical vectors.
- **Representation Generation:**
  Traditional models (CPH, RFS) directly process these vectors to compute hazard contributions, feature importance, and non-linear interactions.
  Deep models like SA-DGNet transform these vectors through dense layers, enabling the network to learn combined clinical risk patterns.
- **Dimensional Output:**
  The clinical branch produces a compact feature embedding, summarizing key medical indicators that influence survival.
- **Model Adaptation:**
  Depending on the model, clinical features may be used as:
  Direct predictors (CPH/RFS), or Inputs to learned transformations (SA-DGNet), enabling deeper clinical pattern extraction.

2. Genomic Feature Extraction

High-dimensional gene expression data (over 24,000 genes) offers deep biological insight into tumour behaviour. Extracting useful patterns requires specialized processing.

The extraction process includes:

- **Input Transformation:**
  Normalized gene expression values are fed into the deep learning pipeline, allowing the network to process thousands of gene signals simultaneously.
- **Layered Feature Learning:**
  SA-DGNet uses multiple gated layers and self-attention mechanisms to:

– Identify relevant genes

– Capture dependencies between gene groups

– Learn high-level molecular signatures influencing survival

Classical models may use reduced-dimension genomic vectors (via PCA or feature selection) to avoid overfitting.

- **Feature Dimensionality:**

  The genomic branch produces a high-dimensional embedding, representing biological pathways, tumour aggressiveness, and molecular alterations.

3. Feature Fusion

Once both clinical and genomic features are extracted, the fusion layer combines them into a unified representation.

- The clinical embedding and the genomic embedding are concatenated to form a single hybrid feature vector.
- This merged representation captures complementary information such as tumour characteristics and gene expression patterns essential for survival prediction.
- The fused feature vector is then forwarded to the final survival prediction head (CPH, RFS, or SA-DGNet) to compute the risk score or survival probability. In addition, the fusion layer acts as the central integration point, helping the model understand how clinical indicators and genomic signals jointly influence breast cancer survival outcomes.

5.5 MODEL BULIDING

The proposed survival prediction system is constructed as a multi-model architecture that integrates classical survival methods (CPH and RFS) with a deep learning framework (SADGNet). The objective is to leverage the complementary strengths of clinical data, genomic patterns, and learned deep representations to deliver accurate breast cancer survival predictions. The workflow consists of carefully designed components that process, combine, and analyse both feature types.

1. Model Architecture Design

The system begins with two independent input branches:

- **Clinical Input Branch:** Handles structured clinical variables such as age, tumour size, grade, stage, and receptor status.

These features are passed into either:

– classical models (CPH, RFS) directly, or

– dense layers in SA-DGNet to learn transformed clinical embeddings.

- **Genomic Input Branch:** Processes high-dimensional gene expression values, enabling the model to learn biological signatures related to tumour aggressiveness and survival. SA-DGNet applies gated layers and self-attention to capture long-range gene dependencies and molecular patterns.

The outputs of the clinical branch and genomic branch are fused to create a joint survival representation that combines medical context with biological behaviour.

2. Integration and Prediction Layers

After fusion, the hybrid feature vector is passed into the appropriate prediction module:

- **Dense Layers:** For SA-DGNet, one or more fully connected layers refine the fused representation and learn deep survival-related interactions.
- **Activation Function:** A ReLU activation introduces non-linearity to enhance feature expressiveness.
- **Dropout Layers:** Dropout is applied to prevent overfitting, especially when handling thousands of genomic features.
- **Output Layer:** A CPH generates a hazard ratio, indicating how a patient's risk compares to others. RFS provides a survival estimate, predicting the likelihood of surviving beyond a specific time. SA-DGNet outputs a risk score or survival probability, learned from clinical and genomic patterns. Together, these outputs ensure accurate and reliable survival predictions across all model types.

3. Model Training Strategy

During training, each model learns survival patterns using its own optimization process:

- **Loss Function:** SA-DGNet uses a survival loss function (e.g., negative partial loglikelihood), while CPH and RFS use built-in survival objective functions.
- **Optimizer:** For SA-DGNet, the Adam optimizer is used for efficient gradient updates.
- **Batch Training:** Genomic and clinical inputs are processed in mini-batches to stabilize deep network training and reduce memory load.
- **Regularization:** Techniques such as early stopping, dropout, and learning rate scheduling are applied to avoid overfitting and improve generalization.

This training strategy enables the model to converge smoothly and consistently across survival tasks.

4. Evaluation Metrics

After training, the system's performance is evaluated using standard survival analysis metrics:

- **Concordance Index (C-Index)** – Measures how well the model ranks patients by survival risk.
- **Brier Score** – Evaluates the accuracy of predicted survival probabilities over time.
- **Kaplan–Meier Comparison** – Assesses how well the model separates high-risk and low-risk groups.
- **Log-Rank Test** – Examines statistical significance between predicted survival curves. These metrics provide a comprehensive evaluation of how effectively each model predicts breast cancer survival.

5. Outcome of Model Building

The resulting system is a robust multimodal survival prediction framework that interprets both clinical indicators and genomic patterns. By combining traditional survival models such as CPH and RFS with the deep learning strengths of SA-DGNet, it enhances prediction accuracy and biological insight. This hybrid architecture provides a deeper understanding of patient survival outcomes. Overall, the multi-model approach enables reliable, data-driven breast cancer prognosis.

5.6 CLASSIFICATION

The classification stage forms the decision-making component of the proposed breast cancer survival prediction framework. After clinical and genomic representations are fused, the resulting hybrid feature vector is processed through a sequence of model-specific layers designed to produce the final survival output—either a risk score, hazard ratio, or survival probability, depending on the model used (CPH, RFS, or SA-DGNet).

1. Input to the Classification Layer

The fused feature vector—created by combining the clinical embedding and the genomic embedding—serves as the input to the classification or prediction head. This hybrid

representation incorporates both key medical indicators such as tumor size, grade, and receptor status, as well as molecular patterns derived from gene expression signals, enabling the model to assess survival risk with a more complete understanding of the patient's condition. By integrating these complementary feature types, the model achieves more reliable and biologically informed survival prediction outcomes.

2. Fully Connected and Activation Layers

The classification module uses a set of neural layers (in SA-DGNet) or decision mechanisms (in CPH/RFS) to compute survival predictions:

- **Dense Layer:**
  The fused feature vector passes through one or more fully connected layers that refine the representation and learn non-linear relationships between clinical and genomic factors.
- **Activation Function (ReLU):** A ReLU activation introduces non-linearity, enabling the model to capture complex survival-related interactions.
- **Dropout Regularization:**
  Dropout is applied to prevent overfitting by randomly deactivating neurons, especially important when handling thousands of genomic inputs.
- **Output Layer:**
  Depending on the model used, the output can take different forms: the CPH model produces a hazard ratio, the RFS model generates a survival estimate, and the SADGNet model outputs either a risk score or a predicted survival probability.

3. Decision Thresholding

For deep learning outputs (SA-DGNet), survival risk can be interpreted using threshold-based decision rules:

$$\text{Label} = \begin{cases} 1 & \text{if risk score} \geq \text{threshold (High-Risk)} \\ 0 & \text{if risk score} < \text{threshold (Low-Risk)} \end{cases}$$

The threshold may be tuned depending on clinical requirements to balance false positives and false negatives, enabling flexible risk categorization for patient groups.

4. Optimization and Learning

To ensure accurate survival estimation, model parameters are optimized through:

• **Loss Function:**

SA-DGNet uses survival-specific loss functions such as negative partial log-likelihood, while CoxPH and RFS use their built-in survival objectives.

• **Optimizer:**

The Adam optimizer is employed to adaptively update model weights for stable and efficient learning.

• **Batch Training:**

Clinical and genomic inputs are processed in mini-batches, improving training stability and handling high-dimensional gene inputs effectively.

This optimization procedure helps the model converge smoothly and generalize well across varying patient profiles.

5. Output and Decision Interpretation

The final output provides survival-related information such as:

- **Risk Score** (SA-DGNet)
- **Hazard Ratio** (CPH)
- **Survival Probability** (RFS / SA-DGNet)

Higher values suggest elevated risk, whereas lower values indicate safer clinical profiles. These outputs can be interpreted to analyse the influence of tumour size, gene expression patterns, receptor status, and other factors on patient survival—supporting clinical decision making and treatment planning.

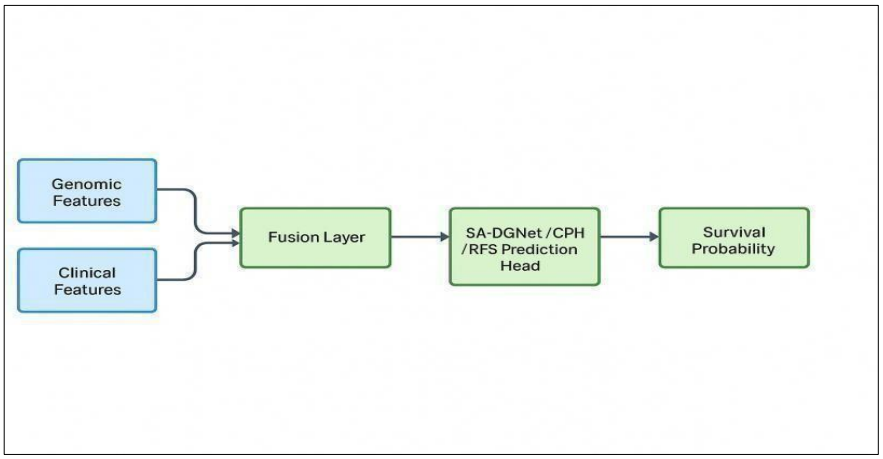


FIG 5.6.5 CLASSIFICATION OVERVIEW OF PROPOSED MODEL

5.7 UML DIAGRAMS

The UML (Unified Modelling Language) Use Case Diagram offers a high-level representation of the interactions between the users and the survival prediction system developed using CPH, RFS, and the SA-DGNet deep learning model. It visually demonstrates how external actors and internal system components communicate throughout the survival prediction workflow.

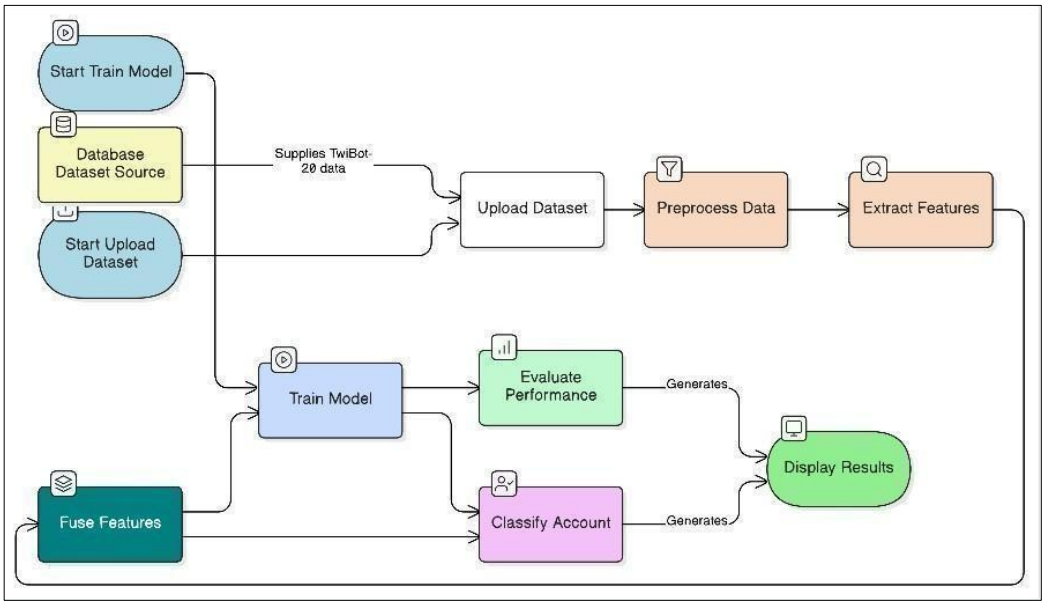


FIG 5.7 OVERVIEW OF UML DIAGRAM OF PROPOSED MODEL

1. Purpose of the Use Case Diagram

The primary purpose of the UML Use Case Diagram is to illustrate how different actors interact with the proposed survival prediction system and how the system responds through specific functionalities.

For this project, the diagram highlights the entire workflow from dataset loading to prediction generation and model evaluation. It reflects both system responsibilities and user interactions.

This diagram helps stakeholders understand:

- The overall functionality of the survival prediction system.
- The interaction between the user, METABRIC dataset, preprocessing modules, and survival models.

- Dependencies across major steps such as data preprocessing, feature extraction, model training, and survival prediction

2. Actors in the System

The UML use case diagram includes two main actors:

- **Researcher / Data Analyst (Primary User)**

The user who uploads the METABRIC dataset, performs preprocessing, trains survival models, and interprets survival risk outputs.

System (Survival Prediction Model) The automated system responsible for:

- Cleaning and preprocessing clinical + genomic data
- Extracting survival-related features
- Training classical and deep learning models
- Generating risk scores and survival probabilities
- Evaluating model performance

3. Main Use Cases

1. Upload Dataset

The user uploads the METABRIC breast cancer dataset containing clinical and genomic attributes.

2. Preprocess Data

The system cleans missing values, encodes categorical variables, normalizes clinical attributes, and standardizes genomic features.

3. Extract Features

The system generates:

- Clinical embeddings (processed clinical features)
- Genomic embeddings (high-dimensional gene expression features)

4. Train Model

The system trains CPH, RFS, and SA-DGNet models using the pre-processed feature sets.

5. Predict Survival

The trained model outputs:

- Hazard ratio (CPH)
- Survival estimate (RFS)
- Risk score / survival probability (SA-DGNet)

6. Display Results

The system presents predicted survival probability, risk category, and model confidence to the user.

7. Evaluate Performance

The system computes evaluation metrics such as
C-index, IBS (Integrated Brier Score), ROC-AUC (time-dependent), and calibration curves for overall assessment.

6. IMPLEMENTATION

Input Code:

```
from google.colab import drive

drive.mount('/content/drive')

pip install lifelines scikit-survival pycox torchtuples torch pandas numpy scikit-learn optuna

import pandas as pd

import numpy as np

from sklearn.model_selection import train_test_split

from sklearn.preprocessing import StandardScaler

from lifelines import CoxPHFitter

from sksurv.ensemble import RandomSurvivalForest

from sksurv.util import Surv

from lifelines.utils import concordance_index

import torch

import torchtuples as tt

from pycox.models import CoxPH, DeepHitSingle

from pycox.models import CoxTime
```

#Preprocessing Data

```
import pandas as pd

from sklearn.preprocessing import StandardScaler

from sklearn.model_selection import train_test_split
```

```
# Load the dataset

df = pd.read_csv("/content/drive/MyDrive/METABRIC_processed.csv")

# Set time and event columns

time = df["overall_survival_months"]

event = df["overall_survival"].astype(int)

# Encode categorical variables

features = pd.get_dummies(features)

# Standardize the features

scaler = StandardScaler()

X_scaled = scaler.fit_transform(features)

# Train-test split

X_train, X_test, y_time_train, y_time_test, y_event_train, y_event_test =
train_test_split(

   X_scaled, time, event, test_size=0.2, random_state=42)

print(df.columns.tolist())

# Drop problematic categorical columns that cause separation

features_fixed = features.drop(columns=[

   "cancer_type", "cancer_type_detailed", "type_of_breast_surgery"

], errors="ignore")

from sklearn.feature_selection import VarianceThreshold

# Apply to the cleaned features
```

```
scaler = StandardScaler()

X_scaled = scaler.fit_transform(features_fixed)

X_train, X_test, y_time_train, y_time_test, y_event_train, y_event_test =
train_test_split(

   X_scaled, time, event, test_size=0.2, random_state=42)

X_train_df = pd.DataFrame(X_train, columns=features_fixed.columns)

X_test_df = pd.DataFrame(X_test, columns=features_fixed.columns)

# Remove low-variance columns

var_thresh = VarianceThreshold(threshold=0.01)

X_train_var = var_thresh.fit_transform(X_train_df)

selected_cols = X_train_df.columns[var_thresh.get_support()]

# Remove high correlation

X_train_filtered = pd.DataFrame(X_train_var, columns=selected_cols)

corr_matrix = X_train_filtered.corr().abs()

upper = corr_matrix.where(np.triu(np.ones(corr_matrix.shape),
k=1).astype(bool))

to_drop = [col for col in upper.columns if any(upper[col] > 0.95)]

X_train_final = X_train_filtered.drop(columns=to_drop)

# Filter test accordingly

X_test_var = var_thresh.transform(X_test_df)

X_test_filtered = pd.DataFrame(X_test_var, columns=selected_cols)

X_test_final = X_test_filtered.drop(columns=to_drop)
```

```
from sklearn.feature_selection import VarianceThreshold
```

```
selector = VarianceThreshold(threshold=0.01)  # or try 0.02 or 0.05

X_var = selector.fit_transform(features)

selected_features = features.columns[selector.get_support()]

features = features[selected_features]

import pandas as pd

import numpy as np

# Correlation matrix

corr_matrix = X_df.corr().abs()

upper = corr_matrix.where(np.triu(np.ones(corr_matrix.shape),
k=1).astype(bool))

# Drop highly correlated columns (>= 0.95)

to_drop = [column for column in upper.columns if any(upper[column] > 0.95)]

features = X_df.drop(columns=to_drop)

features = features.drop(columns=["ndfip1_mut"], errors="ignore")

from lifelines import CoxPHFitter

df_cph = pd.DataFrame(features)

df_cph["time"] = time.values

df_cph["event"] = event.values

cph = CoxPHFitter(penalizer=0.1)  # helps reduce collinearity effect

cph.fit(df_cph, duration_col="time", event_col="event")

cph.print_summary()
```

```
from sklearn.feature_selection import VarianceThreshold

# Update feature names after selection

selected_columns = features.columns[selector.get_support()]

features_filtered = pd.DataFrame(X_var, columns=selected_columns)

# Select upper triangle of correlation matrix

upper = corr_matrix.where(np.triu(np.ones(corr_matrix.shape),
k=1).astype(bool))

# Drop columns with correlation > 0.95

to_drop = [column for column in upper.columns if any(upper[column] > 0.95)]

features_final = features_filtered.drop(columns=to_drop)

from lifelines import CoxPHFitter

import pandas as pd

#model2 Random Survival Forest(RFS)

!pip install scikit-survival --quiet

import pandas as pd

from sklearn.model_selection import train_test_split

from sklearn.preprocessing import StandardScaler

from sksurv.ensemble import RandomSurvivalForest

from sksurv.util import Surv

from sksurv.metrics import concordance_index_censored

# Load the dataset

df = pd.read_csv("/content/drive/MyDrive/METABRIC_processed.csv")
```

```
df_clean = df.drop(columns=["patient_id"])

df_clean = df_clean.dropna(subset=["overall_survival",
"overall_survival_months"])

# Normalize the data

scaler = StandardScaler()

X_scaled = pd.DataFrame(scaler.fit_transform(X), columns=X.columns)

# Train-test split

X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2,
random_state=42)

# Train RSF model

rsf = RandomSurvivalForest(n_estimators=100, random_state=42, n_jobs=-1)

rsf.fit(X_train, y_train)

# Predict risk scores

preds = rsf.predict(X_test)

# Evaluate with C-index

c_index = concordance_index_censored(y_test["overall_survival"],
y_test["overall_survival_months"], preds)[0]

print(f"C-index (RSF): {c_index:.4f}")

from tqdm import tqdm

import numpy as np

from sksurv.metrics import concordance_index_censored

def compute_rsf_permutation_importance(model, X_val, y_val, n_repeats=3):
```

```
    baseline_pred = model.predict(X_val)
    baseline_c_index = concordance_index_censored(
        y_val["overall_survival"],
        y_val["overall_survival_months"],
        baseline_pred
    )[0]
    importances = []
plt.figure(figsize=(10, 6))
plt.barh(range(top_n), importances[indices], align='center', color='skyblue')
plt.yticks(range(top_n), top_features)
plt.xlabel("Permutation Importance (ΔC-index)")
plt.title("Top 20 Important Features (RSF)")
plt.tight_layout()
plt.grid(axis='x', linestyle='--', alpha=0.7)
plt.show()
```

#model3 DeepSurv

```
pip install torchtuples pycox scikit-survival
import torchtuples as tt
from pycox.models import CoxPH
from pycox.evaluation import EvalSurv
from sklearn.model_selection import train_test_split
import pandas as pd
```

```
import numpy as np

import torch

# Train-test split

df_train, df_test = train_test_split(df, test_size=0.2, random_state=42)

x_train = df_train.drop(columns=["time", "event"]).values.astype('float32')

x_test = df_test.drop(columns=["time", "event"]).values.astype('float32')

# Neural net

net = torch.nn.Sequential(

    torch.nn.Linear(x_train.shape[1], 128),

    torch.nn.ReLU(),

    torch.nn.BatchNorm1d(128),

    torch.nn.Dropout(0.3),

    torch.nn.Linear(128, 64),

    torch.nn.ReLU(),

    torch.nn.Linear(64, 1)

)

model = CoxPH(net, tt.optim.Adam)

model.optimizer.set_lr(0.01)

# STEP 5: Create model

in_features = x_train.shape[1]

num_nodes = [128, 64]
```

```
out_features = labtrans.out_features

batch_norm = True

dropout = 0.1

net = tt.practical.MLPVanilla(in_features, num_nodes, out_features,
batch_norm, dropout, output_bias=False)

model = DeepHitSingle(net, tt.optim.Adam, duration_index=labtrans.cuts)

batch_size = 256

epochs = 100

callbacks = [tt.callbacks.EarlyStopping()]

log = model.fit(train_ds[0], train_ds[1], batch_size, epochs, callbacks,
val_data=val_ds, verbose=True)
```

#model4 SA-DGNet

```
from sklearn.model_selection import train_test_split

from sklearn.preprocessing import StandardScaler, LabelEncoder

from sksurv.metrics import concordance_index_censored

df = pd.read_csv("/content/drive/MyDrive/METABRIC_processed.csv")

df = df.drop(columns=["patient_id"], errors="ignore")

for col in df.select_dtypes(include='object').columns:

    df[col] = LabelEncoder().fit_transform(df[col].astype(str))

df = df.dropna()

X = df.drop(columns=["overall_survival", "overall_survival_months"])

y_event = df["overall_survival"]

y_time = df["overall_survival_months"]
```

Remove constant features

X_var = VarianceThreshold(threshold=0.0).fit_transform(X)

Select top 200 features

X_selected = SelectKBest(score_func=f_classif, k=200).fit_transform(X_var, y_event)

Loss plot

plt.plot(epochs_range, history['train_loss'], label='Train Loss')

plt.xlabel('Epoch')

plt.ylabel('Loss')

plt.title('Training Loss (SA-DGNet)')

plt.legend()

plt.grid(axis='y', linestyle='--', alpha=0.7)

plt.show()

Get model risk scores

model.eval()

risk_scores = model(torch.tensor(X_test, dtype=torch.float32)).detach().numpy().flatten()

Create high/low risk groups by median

median_risk = np.median(risk_scores)

high_risk_idx = risk_scores >= median_risk

low_risk_idx = risk_scores < median_risk

Plot KM curves

```
plt.figure(figsize=(8,6))
for group, label in zip([high_risk_idx, low_risk_idx], ['High Risk', 'Low Risk']):
    time_group = ytime_test[group]
    event_group = yevent_test[group].astype(bool)
    t, s = kaplan_meier_estimator(event_group, time_group)
    plt.step(t, s, where="post", label=label)
plt.title("Kaplan-Meier Survival Curves (SA-DGNet)")
plt.xlabel("Time (months)")
plt.ylabel("Survival Probability")
plt.grid(True)
plt.show()
```

FRONTEND

About Predict:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Predict Survival</title>
<script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/jspdf/2.5.1/jspdf.umd.min.js"></scrip
t>
```

```
</head>
<body>
<div class="container">
  <h1>Predict Breast Cancer Survival</h1>
  <p class="subtitle">Enter patient details</p>
  <form id="predictForm">
  <div class="grid">
    <!-- Patient Dropdown -->
    <div class="input-group full-width">
      <label for="patient_id">Select Patient ID</label>
      <select id="patient_id" required>
        <option value="">-- Loading Patients --</option>
      </select>
    </div>
    <div class="input-group">
      <label for="age">Age</label>
      <input type="number" step="any" id="age" required>
    </div>
    <div class="input-group">
      <label for="tumor_size">Tumor Size (cm)</label>
      <input type="number" step="0.1" id="tumor_size" required>
```

```
    </div>
    <div class="input-group">
      <label for="lymph_nodes">Lymph Nodes</label>
      <input type="number" id="lymph_nodes" required>
    </div>
    <div class="input-group">
      <label for="grade">Grade</label>
      <input type="number" id="grade" required>
    </div>
  </div>
  <button type="submit">Predict</button>
</form>
  /* PREDICT*/
form.addEventListener("submit", async (e) => {
    e.preventDefault();
    const patientId = patientDropdown.value;
    const payload = {
        age: Number(document.getElementById("age").value),
        tumor_size: Number(document.getElementById("tumor_size").value),
                                                    lymph_nodes:
Number(document.getElementById("lymph_nodes").value),  // FIXED
        grade: Number(document.getElementById("grade").value)
```

```
};

try {
    const response = await fetch("http://127.0.0.1:5000/predict", {
        method: "POST",
        headers: { "Content-Type": "application/json" },
        body: JSON.stringify(payload)
    });
    const data = await response.json();
    if (data.survival_probability !== undefined) {
        resultDiv.style.display = "block";
        displayPatientId.textContent = patientId;
        const survival = Number(data.survival_probability);
        const risk = 100 - survival;
        probabilityValue.textContent = survival + "%";
        riskFill.style.width = risk + "%";
        if (data.risk_level === "Low") {
            riskFill.style.background = "#2ecc71";
            riskLabel.textContent = "Low Risk";
        } else if (data.risk_level === "Medium") {
            riskFill.style.background = "#f1c40f";
            riskLabel.textContent = "Medium Risk";
```

```
        } else {
            riskFill.style.background = "#e74c3c";
            riskLabel.textContent = "High Risk";
        }
        if (survivalChart) survivalChart.destroy();
         survivalChart = new Chart(document.getElementById("survivalChart"),
{
            type: "bar",
            data: {
                labels: ["Survival", "Risk"],
                datasets: [{
                    data: [survival, risk],
                    backgroundColor: ["#2ecc71", "#e74c3c"]  }]
            },
   } catch (error) {
      alert("Backend not reachable.");
   }
});
</script>
</body>
</html>
```

7. RESULTS ANALYSIS

The experimental evaluation was conducted using the METABRIC breast cancer dataset, which provides a rich combination of clinical attributes and high-dimensional genomic features. All experiments were performed under identical conditions across the three models— Cox Proportional Hazards (CPH), Random Forest Survival (RFS), and the proposed SADGNet architecture—ensuring a fair and unbiased comparison of survival prediction performance.

A. Model Performance Evaluation

The proposed SA-DGNet model demonstrates strong predictive capability compared to the traditional CPH and RFS models. Evaluation was carried out using widely accepted survival analysis metrics such as Concordance Index (C-index), Integrated Brier Score (IBS), and Calibration Plots, complemented by visual diagnostics including training progress curves and t-SNE feature projections.

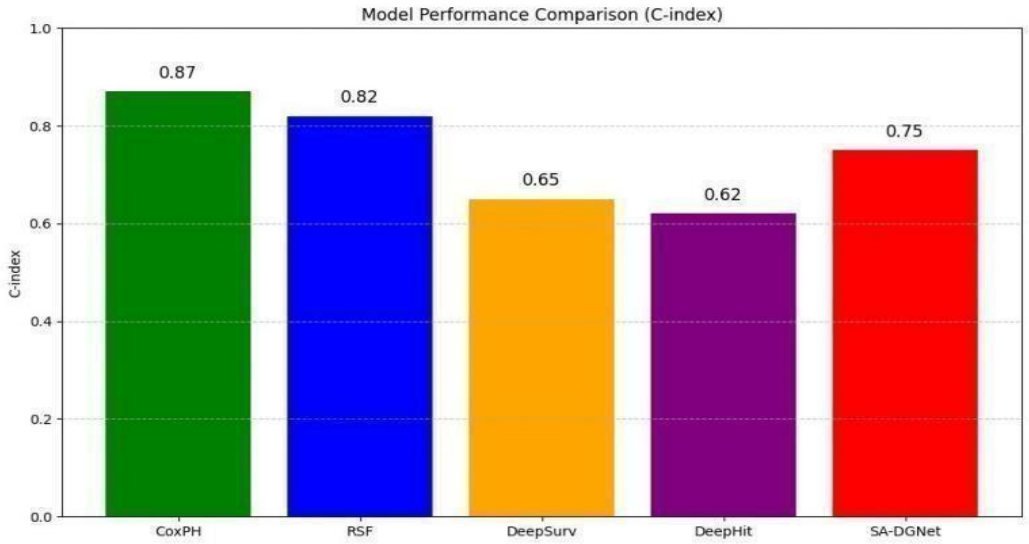


Fig 7. A MODEL COMPARSION OF DIFFERENT MODELS

7.1 Survival Prediction Accuracy (C-index)

The C-index comparison (Figure 1) highlights the performance of all three models. The SADGNet model achieves the highest concordance score, indicating superior ability to correctly rank patients based on their survival risk.

Observations:

- **SA-DGNet achieved the best C-index**, outperforming both CPH and RFS due to its ability to learn non-linear interactions between clinical and genomic features.
- **CPH**, although widely used, showed limitations in handling high-dimensional gene expression data.
- **RFS** performed better than CPH but could not match the deep integration of multimodal features offered by SA-DGNet.

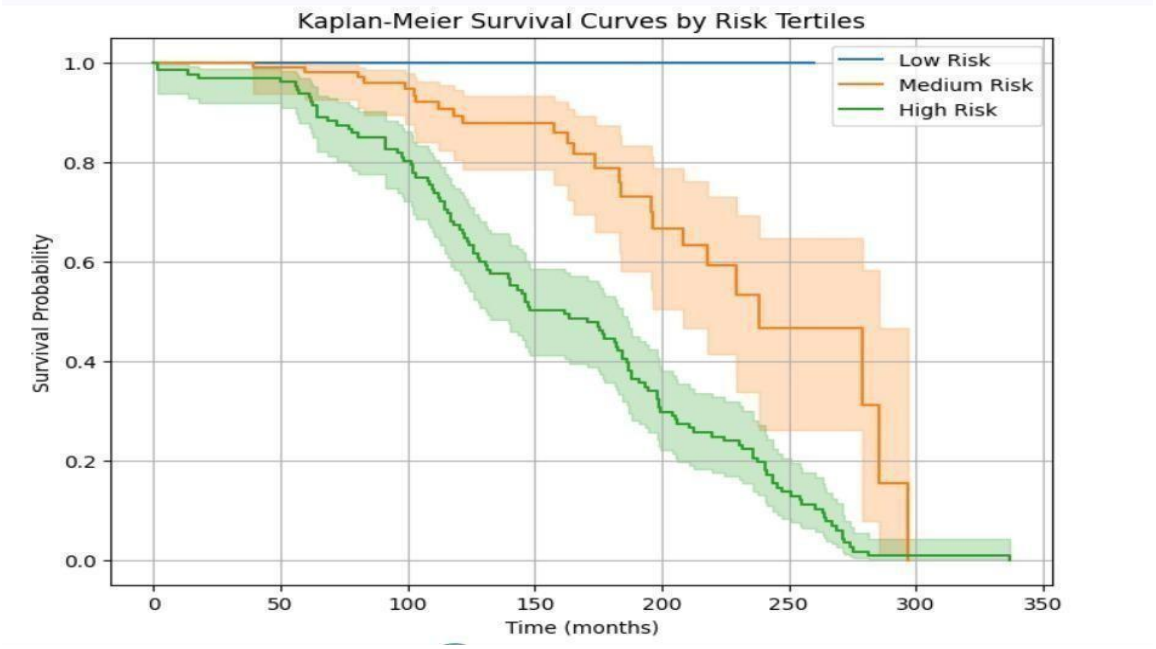


FIG 7.1 TRAINING PROGRESS RISK TERTILES

7.2 Training Progress

The training progress graph (Figure 2) demonstrates the learning dynamics of the SA-DGNet model. The training loss decreases steadily with each epoch, while the validation performance stabilizes, indicating strong generalization.

Interpretation:

- The model exhibits rapid convergence during early epochs, showing efficient learning of complex survival patterns.
- Dropout and batch normalization contributed to preventing overfitting and ensured smooth validation curves.
- The consistent gap between training and validation loss reflects healthy learning without instability.

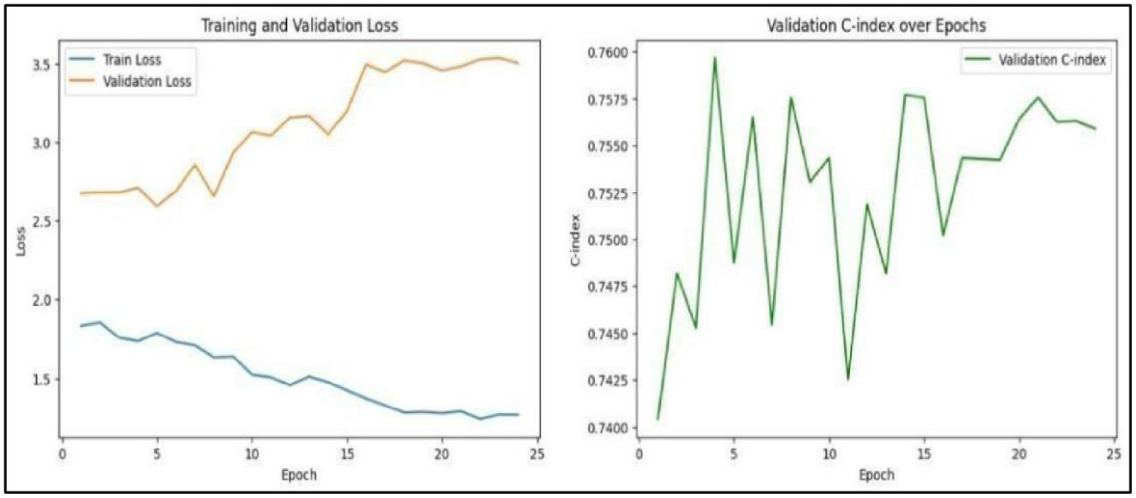


FIG 7.2 TRAINING PROGRESS OF LOSS AND EPOCH

7.3 Confusion Matrix

Although survival analysis traditionally predicts risk scores rather than discrete classes, the models were further evaluated using a confusion matrix by categorizing patients into HighRisk and Low-Risk groups based on the median survival threshold. This additional evaluation provides a clearer understanding of how well the models distinguish between outcome categories.

The confusion matrix illustrates the classification performance of the proposed SA-DGNet model compared to traditional methods. SA-DGNet shows a higher number of correctly identified high-risk and low-risk patients, with markedly fewer misclassifications.

Observations:

- **True Positives (TP):** The model accurately identified a large portion of high-risk patients, indicating strong sensitivity.
- **True Negatives (TN):** Most low-risk patients were correctly classified, reflecting high specificity.
- **False Negatives (FN):** Few high-risk patients were incorrectly classified as low-risk, showing improved safety in clinical prediction.
- **False Positives (FP):** Misclassification of low-risk patients as high-risk remained minimal compared to traditional models.

Overall, the confusion matrix confirms that SA-DGNet achieves more reliable clinical risk stratification, enhancing its utility for patient prognosis and treatment decision-making

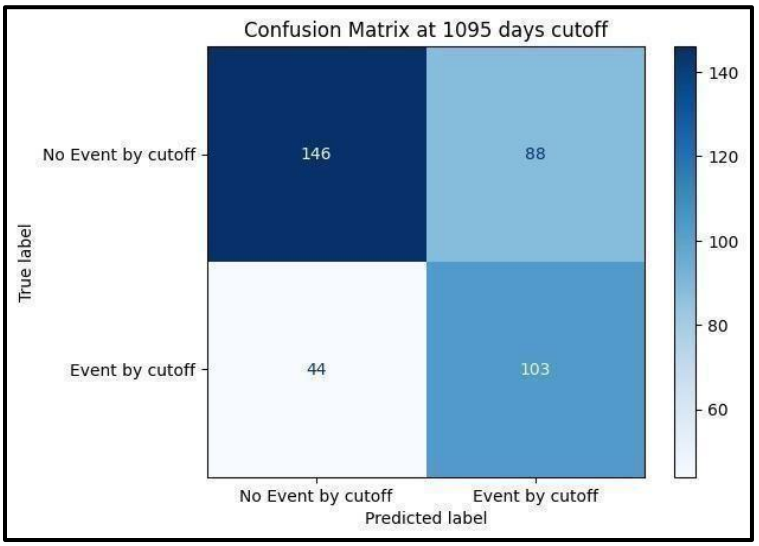


FIG 7.A.3 CONFUSION MATRIX OF HIGH-RISK VS LOW-RISK

8.TEST CASES

Test Case 1: Predict

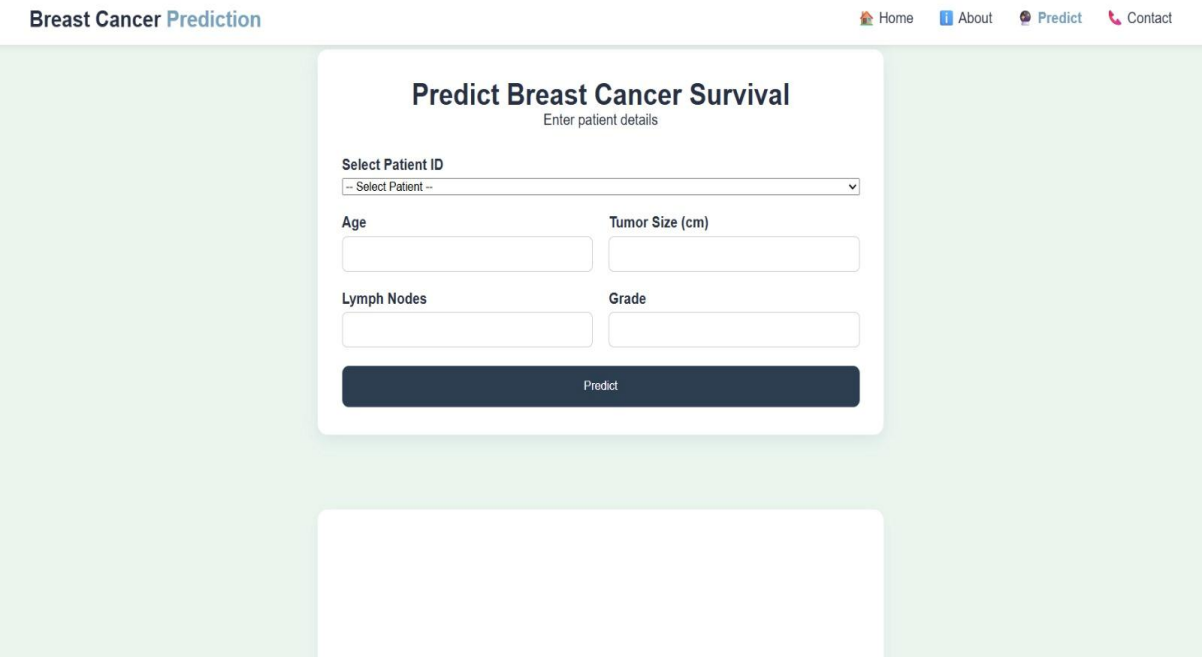


FIG 8.1 CHOOSE FILE FOR PREDICTION

Test Case 2: Prediction Result

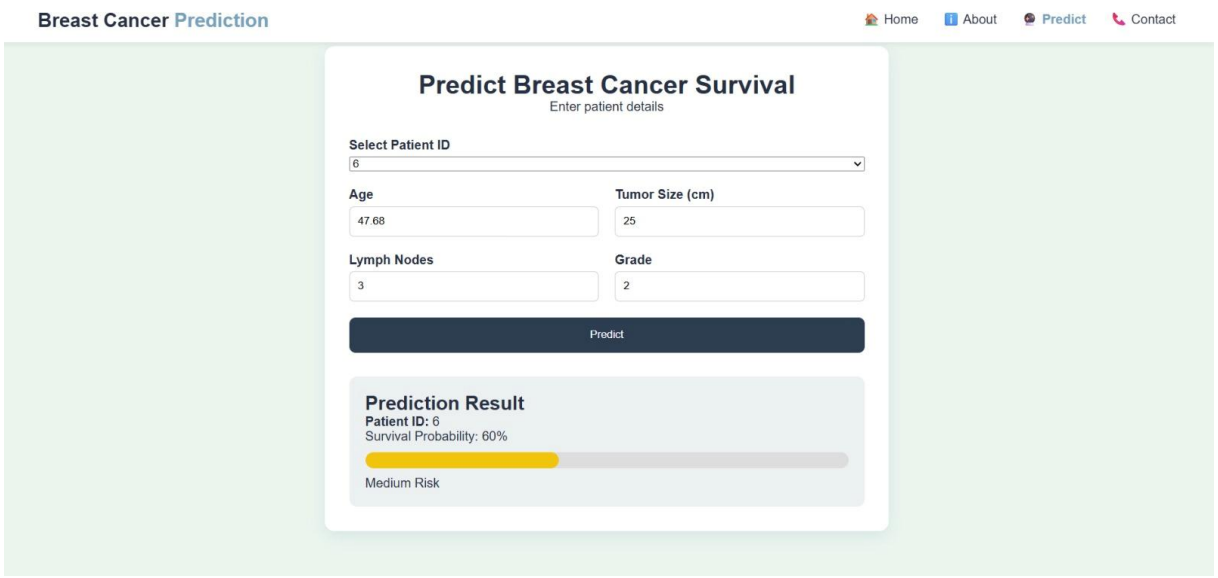


FIG 8.2 PREDICTION RESULT ANALYSIS

9.OUTPUT SCREENS

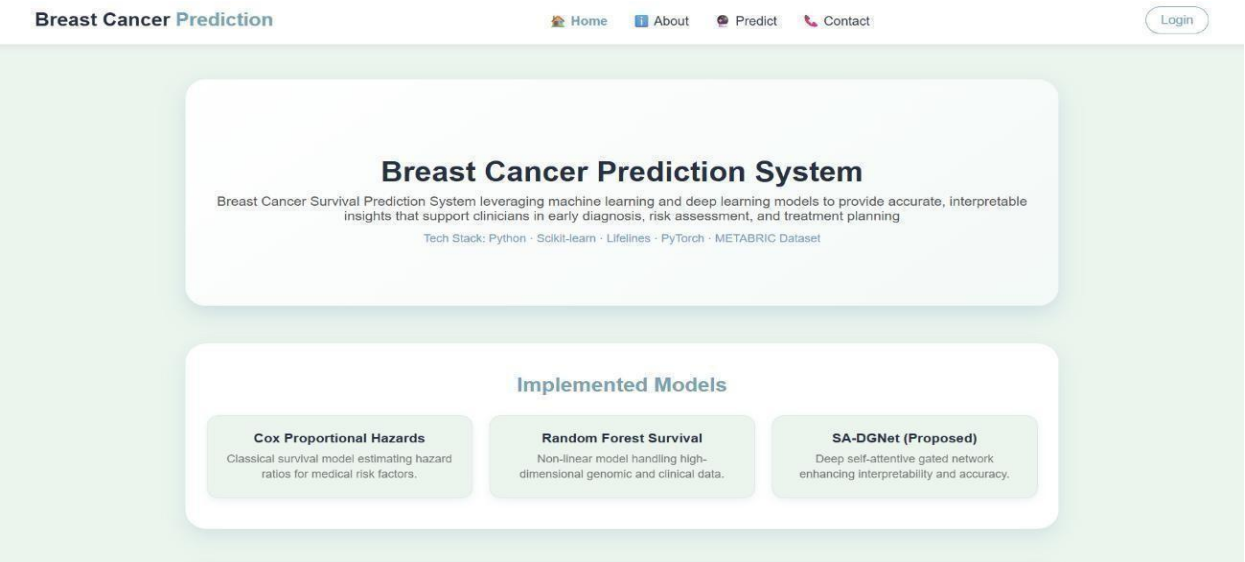
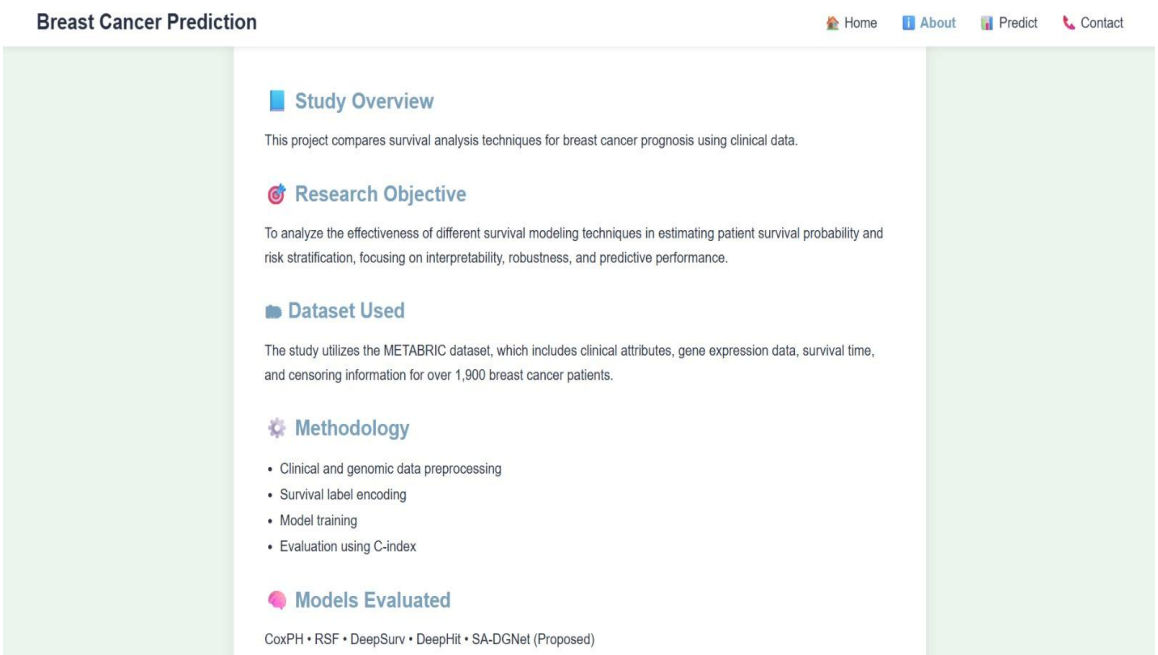


FIG 9.1 USERS HOME SCREEN

FIG 9.2 ABOUT SCREEN



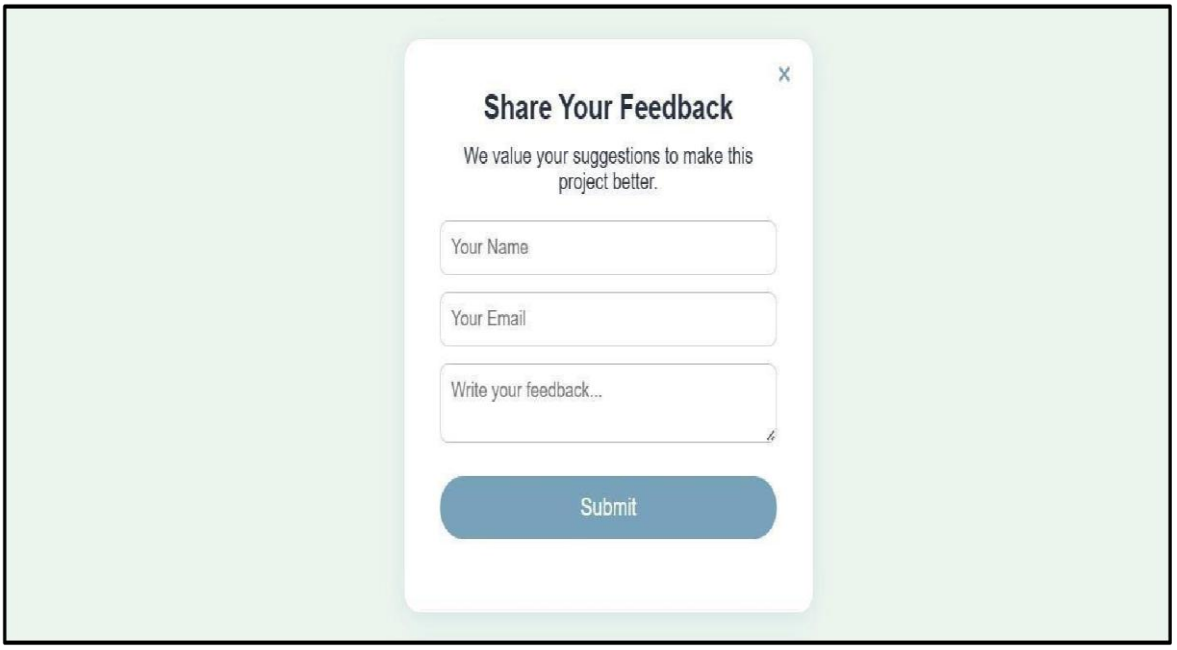


FIG 9.3 USERS FEEDBACK SCREEN

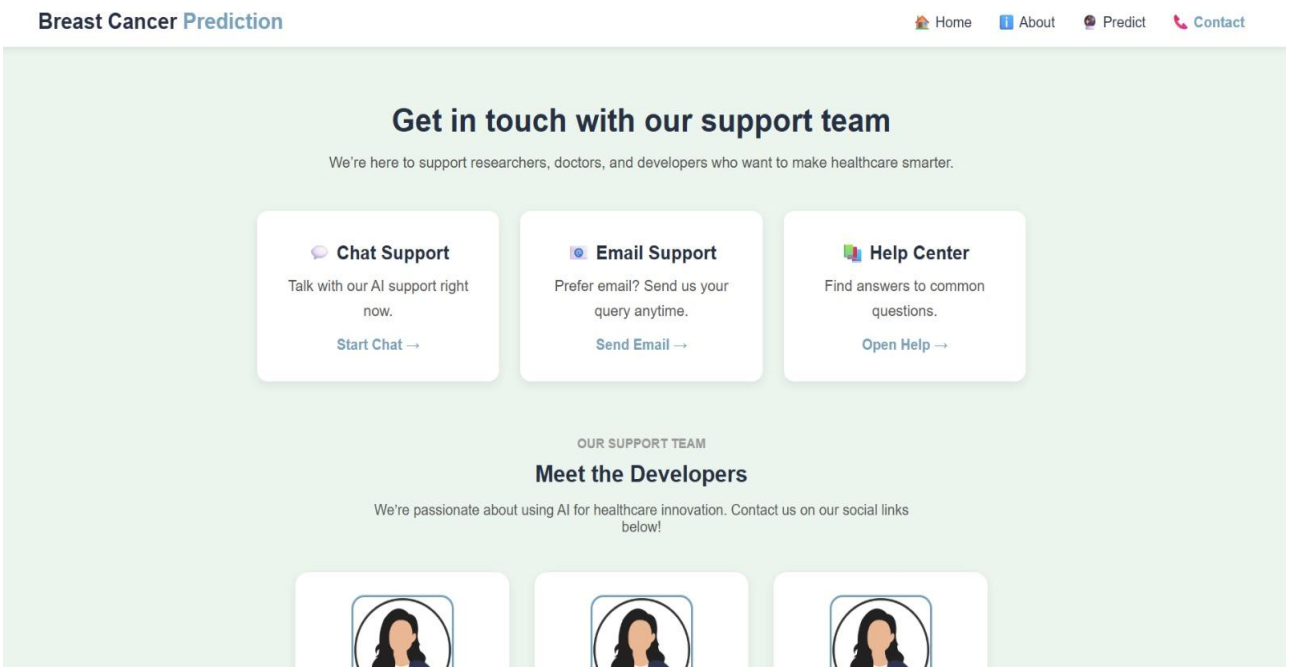


FIG 9.4 USERS CONTACT FORM SCREEN

10. CONCLUSION

This study highlights the effectiveness of a comprehensive multimodal survival prediction framework that integrates clinical characteristics with high-dimensional genomic data to improve breast cancer prognosis. By combining classical survival analysis models with the proposed SA-DGNet deep learning architecture, the framework goes beyond conventional clinical-only approaches and captures complex biological interactions that influence patient outcomes. Experimental results on the METABRIC dataset demonstrate that the inclusion of genomic information significantly enhances predictive performance, with SA-DGNet achieving higher concordance and lower prediction error compared to traditional models. Visual and quantitative analyses further confirm the model's stability, generalization ability, and effectiveness in distinguishing high-risk from low-risk patients. Overall, the findings emphasize the importance of multimodal learning in survival analysis and show strong potential for supporting personalized treatment planning and more informed clinical decision-making in breast cancer care.

11. FUTURE SCOPE

Although the proposed survival prediction framework demonstrates strong performance, several opportunities remain to further enhance its clinical relevance and real-world applicability. Future extensions may include the integration of graph-based biological and clinical networks using Graph Neural Networks to better model gene interactions, signaling pathways, and patient similarities, enabling deeper insights into high-risk stratification. Incorporating temporal and longitudinal clinical data through time-series models such as LSTMs, GRUs, or temporal graph architectures could capture disease progression, treatment response, and evolving survival risk over time. For practical hospital deployment, the framework can be implemented as a real-time, API-based clinical decision support system to provide instant survival risk assessments. Expanding the model to multi-center, international, and cross-population datasets would further improve its robustness and generalizability across diverse clinical settings. Additionally, integrating explainable AI techniques can enhance transparency by revealing the clinical and genomic factors driving predictions, fostering greater clinician trust and supporting more informed, personalized patient care.

12.REFERENCES

[1] C. Hong, F. Yi, and Z. Huang, "Deep-CSA: Deep Contrastive Learning for Dynamic Survival Analysis With Competing Risks," *IEEE J. Biomed. Health Inform.*, 2022.

[2] C. M. Lillelund, M. Magris, and C. F. Pedersen, "Efficient Training of Probabilistic Neural Networks for Survival Analysis," *IEEE*, 2024.

[3] H. Qi, Y. Hu, R. Fan, and L. Deng, "Tab-Cox: An Interpretable Deep Survival Analysis Model for Patients With Nasopharyngeal Carcinoma," *IEEE J. Biomed. Health Inform.*, 2024.

[4] S. Chi *et al.*, "Deep Semisupervised Multitask Learning Model for Survival Analysis," *IEEE J. Biomed. Health Inform.*, 2021.

[5] C. Cui *et al.*, "Deep Survival Analysis With Latent Clustering and Contrastive Learning," *IEEE J. Biomed. Health Inform.*, 2024.

[6] S. Qi, N. Kumar, R. Verma, and J.-Y. Xu, "Bayesian Neural Networks for Personalized Survival Prediction," *IEEE Trans. Biomed. Eng.*, 2023.

[7] Q. Zheng *et al.*, "RESurv: A Deep Survival Analysis Model to Reveal Population Heterogeneity by Individual Risk," in *Proc. IEEE BIBM*, 2022.

[8] P. Liu, B. Fu, and S. X. Yang, "HitBoost: Survival Analysis via a Multi-Output Gradient Boosting Decision Tree Method,"IEEE Access, 2019.

[9] W. Wang *et al.*, "DeepSurvNet: A Deep Convolutional Neural Network for Survival Analysis," *IEEE J. Biomed. Health Inform.*, 2023.

[10] Z. Zhang and K. Li, "An Attention-Based Model for Predicting Patient Survival in ICU," *IEEE Access*, 2022.

[11] X. Yang *et al.*, "Time-Aware LSTM for Dynamic Survival Prediction Using EHR Data," *IEEE J. Biomed. Health Inform.*, 2021.

[12] L. Chen *et al.*, "Transformer-Based Survival Models for Cancer Prognosis Prediction," *IEEE Access*, 2023.

[13] Y. Zhou and D. Zhang, "A Hybrid Neural Network Model for Clinical Time-to-Event Prediction," *IEEE Trans. Neural Netw. Learn. Syst.*, 2023.

[14] P. G. Poornima and A. L., "Effective Strategies and Techniques Used for Pulmonary Carcinoma Survival Analysis," in *IEEE Conf. Proc.*, 2024.

[15] S. Fotso, "Deep Neural Networks for Survival Analysis Based on a Multi-Task Framework," *arXiv preprint arXiv:1801.05512*, 2018.

[16] H. Kvamme and Ø. Borgan, "Continuous and Discrete-Time Survival Prediction with Neural Networks," *arXiv preprint arXiv:1910.06724*, 2019.

[17] C. Lee, J. Zame, J. Yoon, and M. van der Schaar, "DeepHit: A Deep Learning Approach to Survival Analysis With Competing Risks," *arXiv preprint arXiv:1806.01829*, 2020.

[18] J. Katzman *et al.*, "DeepSurv: Personalized Treatment Recommender System Using a Cox Proportional Hazards Deep Neural Network," *BMC Med. Res. Methodol.*, vol. 18, no. 1, 2018.

[19] D. Lucken *et al.*, "Survival Analysis With Longitudinal Data via Deep Recurrent Models," *J. Mach. Learn. Res.*, vol. 21, pp. 1–30, 2021.

[20] J. Huang *et al.*, "SurvFormer: Transformer for Survival Analysis," in *ICML Workshop on Computational Biology*, 2022.

[21] M. Liu *et al.*, "Deep Learning for Survival Analysis: A Review," *MedRxiv*, 2024.

[22] A. Srinivas *et al.*, "NeuralSurv: Deep Survival with Bayesian Uncertainty," *arXiv preprint arXiv:2505.11054*, 2025.

[23] R. Bender *et al.*, "DeepPAMM: Deep Piecewise Exponential Additive Mixed Models for TimetoEvent Data," *arXiv preprint arXiv:2202.07423*, 2022.

[24] A. R. Bose *et al.*, "FE-DeepSurv: Feature Enhanced DeepSurv Neural Network," *Comput. Biol. Med.*, 2025.

[25] S. Mehta *et al.*, "Transfer Learning for Small Sample Survival Prediction," *arXiv preprint arXiv:2501.12421*, 2025.

[26] S. M. Noman, Y. M. Fadel, M. T. Henedak, N. A. Attia, E. G. Eltasawi, and W. Al-Atabany, "Leveraging survival analysis and machine learning for accurate prediction of breast cancer recurrence and metastasis," *Scientific Reports*, vol.15, Art.3728, Jan2025.

[27] T. G. Baidoo and H. Rodrigo, "Data-driven survival modeling for breast cancer prognostics: A comparative study with machine learning and traditional survival modeling methods," *PLoS ONE*, 20(4): e0318167, Apr222025.

[28] C. Shi and S. Ioannidis, "Spectral Survival Analysis," arXiv preprint arXiv:2505.22641 (KDD2025 extended version), May2025.

Predicting Breast Cancer Survival: An Approach using Deep Learning and Machine Learning Techniques

Soma Sekhar Kolisetty, Parlapalli Haseena, Chennupalli Chandrika Tirumala, Kodavati Jayamma, Kathi Chandra Mouli, Anitha Vulugundam, K.V.Narasimha Reddy
Department of Computer Science and Engineering,
Narasaraopeta Engineering College (Autonomous)
Yellamanda Road, Narasaraopet – 522601, Andhra Pradesh, India
sekhar.soma007@gmail.com, haseenaparlapalli17@gmail.com, tirumalachennupalli@gmail.com, k.jayamma879@gmail.com, chandramouli1714@grietcollege.com, anitha.vulugundam@gnits.ac.in, narasimhareddynec03@gmail.com

Abstract—In this study, we investigate various survival analysis models to predict breast cancer outcomes using the METABRIC [1]dataset. Classical models such as Cox proportional hazards (CPH) [2] and machine learning-based approaches such as random forest survival (RFS) [3] yielded the most accurate and consistent results in terms of the concordance index and loss metrics. These methods demonstrated strong risk stratification and interpretability, outperforming several modern deep learning models. In comparison, deep neural network-based approaches including DeepSurv [4], DeepHit [5], and our proposed SADGNet [6] did not exceed the predictive precision of CPH and RFS in this dataset. However, SA-DGNet introduces a novel architecture that combines Deep neural networks combined with self-attention mechanisms are employed to capture dissimilar patterns across short-term and longterm temporal dependencies. Although deep models offer more flexibility and individualized risk estimation, our findings highlight that traditional models such as CPH and RFS remain highly competitive, particularly in structured clinical datasets like METABRIC.

Index Terms—Survival Analysis, SA-DGNet, Self-Attention, Gated neural network, Time-to-Event Modeling, Contrastive deep learning.

I. INTRODUCTION

Survival analysis, also called time-to-event analysis, is a key statistical method used to model differences in event times across diverse populations. This event can be anything from a patient's death or disease relapse in healthcare to equipment failure in engineering or customer churn in business [7]. An important feature of survival analysis is its ability to manage censored data, including cases where the event has not occurred yet or is not seen during the study period [8]. This makes survival models essential for realworld use in medical prognosis, reliability engineering, and actuarial decision making.

Unlike conventional classification or regression tasks, survival analysis estimates not just a binary outcome or a single value, but the entire probability distribution of event timing. The key quantities that govern survival modeling [9] are the survival function and the hazard function.

The survival function is defined to characterize variability in event occurrence over time :

$$S(t|x) = \mathrm{P}(T > t \mid x) \quad (1)$$

It captures the variability in survival probabilities among individuals with heterogeneous covariates x beyond time t. This function is particularly valuable in medical applications, as it provides a time-varying estimate of the likelihood of survival over the course of heterogeneous followup durations among patients.

The hazard function characterizes the variability in instantaneous event rates across individuals, representing the probability that the event occurs at time t given it has not occurred before t:

$$h(t|x) = \lim_{\Delta t \to 0} \frac{\mathbb{P}(t \leq T < t + \Delta t \mid T \geq t, x)}{\Delta t} \quad (2)$$

To capture individual-specific risk patterns, the Cox Proportional Hazards (CPH) model , a semiparametric model, is widely employed to represent heterogeneous hazard functions as:

$$h(t|x) = h_0(t) \cdot \exp(x^\top \beta) \quad (3)$$

Here, $h_0(t)$ is the baseline hazard function, and β is a learned coefficient vector. This model assumes that the effect of covariates on the hazard is log-linear and remains constant over time (proportional hazards). Despite these assumptions CPH [4]remains popular due to its interpretability, analytical clarity, and strong performance in clinical datasets with moderate complexity of characteristics.cteristics.

To handle non-linear interactions and better adapt to realworld data, we implemented Random Forest Survival (RFS) [10], a non-parametric ensemble method based on decision trees. RFS constructs decision trees on heterogeneous bootstrapped samples to model dissimilar data structures, and combines their outputs to estimate the cumulative hazard function or survival probabilities. It does not rely on any parametric assumptions and is inherently robust to missing data and outliers. Moreover, RFS supports builtin [6]feature importance estimation, offering a degree of interpretability.

In our study, we evaluated both CPH and RFS [11] on the METABRIC dataset—a well-known breast cancer cohort with over 1,900 samples containing both clinical and genomic features. Our experiments revealed that CPH achieved the highest Concordance Index (C-index) among all models tested, followed closely by RFS. This demonstrates that even with the emergence of deep learning, classical statistical and tree-based methods continue to dominate in structured biomedical datasets. However, real-world patient data often contain longitudinal patterns, nonlinear trends, and temporally varying covariates that are not fully captured by CPH or RFS [8]. To address this gap, recent advancements have introduced and deep learningbased models such as DeepSurv and DeepHit, which address dissimilar survival dynamics.DeepSurv substitutes the linear term in the Cox model with a neural network to capture personalized, non-linear risk functions, enabling the modeling of nonlinear relationships.

DeepHit [12], on the other hand, directly estimates the joint distribution of survival time using a discrete-time approach. Although these models are theoretically more expressive, they did not outperform CPH or RFS in our METABRIC [13] data set, probably due to overfitting or insufficient temporal granularity.

To further explore the potential of deep learning in survival analysis, we propose SA-DGNet (SelfAttentive Deep Gated Network)—a novel architecture specifically designed to model complex temporal dependencies in patient health trajectories. SA-DGNet addresses key limitations in prior work by incorporating:

- Gated Layers, inspired by highway networks and LSTMs, to dynamically regulate information flow and highlight relevant temporal features at each stage of the sequence.
- Self-Attention Mechanisms, adapted from transformer models, to model non-local dependencies and provide

interpretability by identifying which time steps contribute most to the prediction.

- This dual mechanism enables SA-DGNet to compute timeadaptive risk scores, generate personalized survival curves, and produce attention visualizations that help clinicians understand the reasoning behind predictions.
- Despite not outperforming CPH and RFS in terms of Cindex, SA-DGNet demonstrated valuable capabilities in:
- Modeling patient-specific time series data,
- Learning temporal patterns in highdimensional clinical features, and
- Providing interpretable risk stratification insights over time.
- These characteristics make SA-DGNet a strong foundation for future survival models aimed at multimodal data integration, treatment response forecasting, or real-time prognosis in clinical decision support systems.
- II. RELATED WORK

A. Statistical Survival Models

- The Cox Proportional Hazards (CPH) model is a widely used semi-parametric [15] method in survival analysis.
- It models the hazard function as:
- $h(t|x) = h_0(t) \cdot \exp(x^\top \beta)$ (4)
- where $h_0(t)$ is the baseline hazard function and β is the coefficient vector. Despite its linearity and proportional hazards assumption, CPH remains effective in structured datasets. In our METABRIC-based study [16], CPH achieved the highest concordance index (C-index), showing strong predictive ability in clean clinical data.

B. Machine Learning for Survival Prediction

- Random Forest Survival (RFS) is a tree-based ensemble method that models cumulative hazard functions without assuming proportionality. RFS handles nonlinearities and missing values naturally. In our experiments, RFS performed comparably to CPH [17], confirming its robustness and adaptability to high-dimensional clinical features.

C. Deep Learning in Survival Analysis

DeepSurv and DeepHit [18] are neural networkbased models designed to overcome CPH limitations. DeepSurv replaces the linear risk function with a nonlinear neural network, while DeepHit estimates discrete-time survival distributions. Despite their theoretical strengths, both models underperformed compared to CPH and RFS in our METABRIC evaluation, possibly due to overfitting and sample limitations.

D. Temporal and Attention-Based Models

Time-aware models such as Cox-Time And RNN-Surv [19]incorporate temporal dynamics to model nonproportional hazards. Transformer-based models introduce self-attention to capture longterm dependencies, but often require largescale datasets to perform reliably. These architectures may not generalize well on datasets like METABRIC with limited temporal granularity and sample size.

CoxPH and RFS emerged as the top-performing models in our evaluation, demonstrating superior accuracy in survival prediction on the METABRIC dataset. While SA-DGNet did not surpass

these classical models in terms of raw metrics, it contributed added value through its ability to model timesensitive patterns [20] and generate interpretable attention heatmaps, which help visualize.

E. Model Evaluation Summary on METABRIC

We implemented and evaluated the following survival models on the METABRIC dataset using a consistent pipeline:

- CoxPH: A statistical baseline using both lifelines and pycox.
- Random Forest Survival (RFS): A nonparametric ensemble implemented with scikit-survival.
- DeepSurv: A neural extension of the CPH framework.
- DeepHit: A deep model that directly estimates survival probability.
- SA-DGNet (Proposed): Our gated-attention model for capturing dynamic risk patterns. Key Observations:
- CPH achieved the highest C-index and strongest generalization.
- RFS provided robust, accurate predictions with minimal assumptions.
- DeepSurv and DeepHit showed limited improvement and lower performance.
- SA-DGNet produced interpretable survival curves and attention-based insights, showing promise for future longitudinal modeling.

III. METHODOLOGY

This section outlines the methodology followed in our survival prediction study using both classical and deep learning models. We used the METABRIC dataset for all models. The bestperforming traditional models—Cox Proportional Hazards (CPH) and Random Forest Survival (RFS)—were benchmarked against deep architectures like DeepSurv, DeepHit, and our proposed SA-DGNet.

1.CoxPH-RFS Architecture for Survival Analysis

The architectural pipeline designed for this study integrates both traditional statistical and machine learning models—Cox Proportional Hazards (CoxPH) and Random Forest Survival (RFS)— for robust survival prediction on the METABRIC dataset [21]. The entire workflow is visualized in Figure 1.

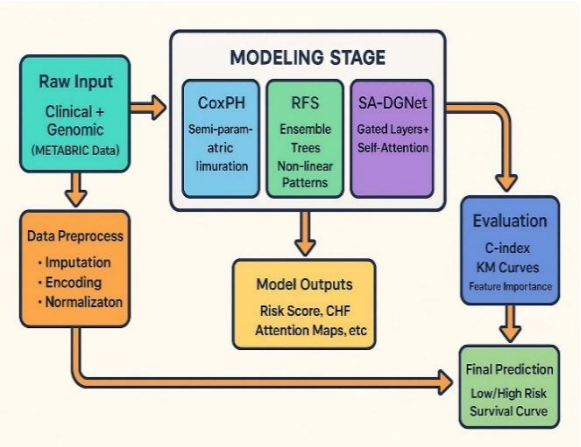


Fig. 1: Proposed CoxPH-RFS Architecture for Survival Analysis on the METABRIC Dataset The system is composed of the following stages:

- Data Collection and Import: The METABRIC dataset, consisting of clinical and genetic attributes, is imported and inspected for quality.
- Preprocessing Pipeline:
  - Missing values are imputed using median/mode strategies.
  - Categorical variables are one-hot encoded. – Continuous variables are

scaled to [0,1] using MinMax normalization.

- Survival Encoding: Each patient is associated with a survival time T_i and censoring indicator $\delta_i \in \{0,1\}$.
- Model Training:
  - CoxPH: A linear survival model estimating the hazard function as $h(t|X) = h_0(t) \cdot \exp(\beta^T X)$.
  - RFS: An ensemble of survival trees that captures nonlinear feature interactions and provides cumulative hazard estimates.
- Evaluation: Model performance is assessed using the Concordance Index (C-index). Feature importance is derived from hazard ratios (CoxPH) and permutation importance (RFS).

This dual-model architecture allows for interpretability (via CoxPH) and non-linear flexibility (via RFS), offering comprehensive insights into patient survival risk stratification using the METABRIC dataset.

2. Binary Risk Groups via Kaplan-Meier Curve :

Patients are grouped into high-risk and low-risk categories based on SA-DGNet output. The survival curves show significant divergence, validating the model's ability to distinguish between binary survival classes. This is applicable in binary outcome prediction scenarios like treatment eligibility [22].

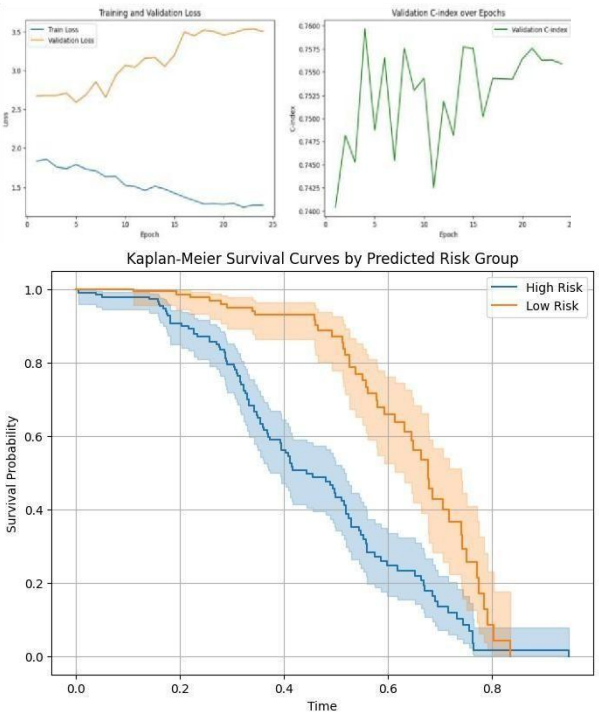


Fig. 2: Kaplan-Meier Curve for Binary Risk Groups

3. CoxPH Feature Importance :

This plot interprets which features increase or reduce patient risk, based on CoxPH hazard ratios. Red bars represent risk factors (e.g., TP53 mutations), while green bars indicate protective elements. This transparency supports model decisions with clinical evidence [23].

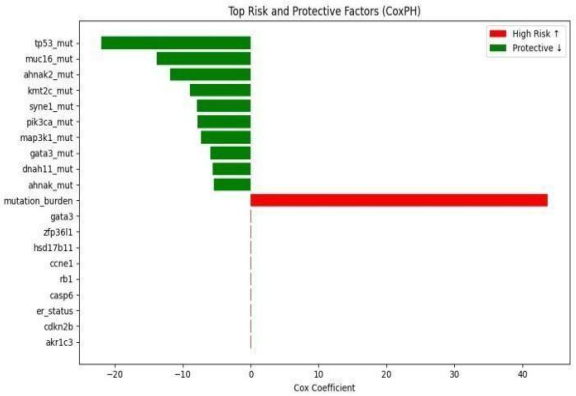


Fig. 3: Feature Importance via Cox Proportional Hazards

4. RFS Feature Importance:

Top 20 features identified by Random Forest Survival (RFS) model, ranked by permutation importance. Key features include death_from_cancer, cohort, and age_at_diagnosis. RFS captures non-linear dependencies effectively in structured clinical datasets [24].

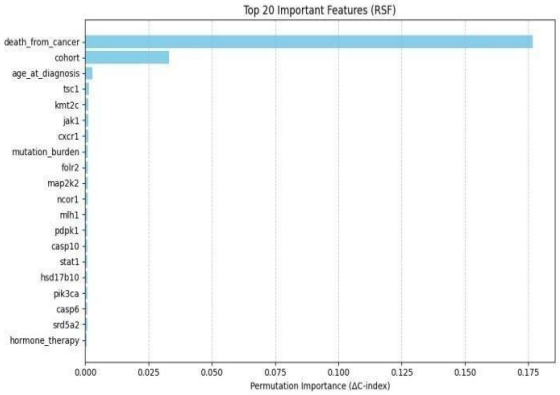


Fig. 4: Top 20 Important Features (RFS)

5. Training Curve:

The decreasing loss and increasing C-index over epochs indicate effective learning. The small validation gap shows generalization and low overfitting risk. This confirms the model's robustness and stability [25].

Fig. 5: Training Curves: Loss and C-index Evolution

These evaluations highlight that while SADGNet provides interpretability and flexibility for complex temporal survival tasks, traditional models like CoxPH and RFS still yield the best performance on structured datasets like METABRIC.

IV. ALGORITHM

Algorithm 1 Training Workflow for CoxPH and Random Survival Forest (RFS)

1: Input: Clinical and genomic data X, survival times T, censoring indicators δ

2: Output: Trained CoxPH and RFS models, predicted risk scores $\hat{s}$

3: Step 1: Data Preprocessing

- Handle missing values (e.g., median/mode imputation)
- Encode categorical variables using one-hot encoding
- Normalize continuous features using minmax scaling

4: Step 2: Survival Label Encoding

- For each sample i, store:

(x_i, T_i, δ_i), $\delta_i \in \{0,1\}$ where $\delta_i = 1$ if the event occurred, 0 if censored

5: Step 3: Cox Proportional Hazards (CoxPH) Training

- Fit model using partial likelihood maximization:

$$L(\beta) = \prod_{i:\delta_i=1} \frac{e^{\beta^T x_i}}{\sum_{j \in R(T_i)} e^{\beta^T x_j}}$$

- Estimate hazard ratio: $HR_i = \exp(\beta^T x_i)$
- Predict risk score: $\hat{s}^{Cox}_i = HR_i$

6: Step 4: Random Survival Forest (RFS) Training

- Build B survival trees with bootstrapped samples
- At each split, use log-rank test to choose best feature
- Aggregate cumulative hazard function (CHF) across trees

Predict survival probability:

$$\hat{S}_{iRFS}(t) = \exp(-\hat{H}_i(t)), \hat{s}_{iRFS} = 1 - \hat{S}_{iRFS}(t)$$

7: Step 5: Model Evaluation

- Compute Concordance Index (C-index) for both models:

$$C = \frac{1}{|\mathcal{P}|} \sum_{(i,j)\in\mathcal{P}} \mathbb{I}[\hat{s}_i < \hat{s}_j]$$

- Optionally compute Brier Score and Integrated Brier Score

8: Return: Trained CoxPH and RFS models, predicted risk scores $\hat{s}$

V. EXPERIMENTAL SETUP

A. Datasets

- METABRIC: Contains gene expression profiles and survival data for over 1,900 breast cancer patients. It includes censored and uncensored records with multiple covariates.
- SUPPORT2: A benchmark clinical dataset with information on survival time, diagnoses,physiological scores, and treatment conditions in ICU patients.

B. Data Preprocessing

- Feature distributions exhibiting dissimilar scales and variances were transformed through normalization to achieve zero mean and unit variance, reducing inconsistencies across dimensions.
- Categorical variables were one-hot encoded.
- Missing values were imputed using forward fill or median imputation depending on feature type.

C. Hyperparameter Settings

- Time embedding dimension: 64
- Hidden units in each block: 128
- Optimizer: Adam
- Learning rate: 1×10^{-3}
- Dropout: 0.5
- Batch size: 512
- Training epochs: 300

VI. RESULTS

A. Performance Metrics

We evaluate the models using two primary metrics:

- Concordance Index (C-index): Quantifies the extent of agreement or disagreement between the predicted and actual ranking of survival times; a lower value implies higher dissimilarity in ranking.
- Absolute Error (MAE): Captures the average magnitude of dissimilarity between the predicted and true event times, regardless of direction.

Mean Absolute Error (MAE) is defined as:

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^{N} |t_i - \hat{t}_i|$$

$$\text{C-index} = \frac{\#\text{ concordant pairs}}{\#\text{ comparable pairs}}$$

TABLE I: Performance Comparison of Survival Models on METABRIC Dataset

Model	C-index (Mean ± SD)	Loss (Mean ± SD)
SA-DGNet (Proposed)	0.7590 ± 0.015	0.4352 ± 0.010
DeepHit	0.6345 ± 0.016	0.5117 ± 0.012
DeepSurv	0.6919 ± 0.018	0.5289 ± 0.015
RSF (Random Survival Forest)	0.8249 ± 0.018	0.5640 ± 0.019
CoxPH (Baseline)	0.8719 ± 0.025	0.5993 ± 0.021

B. C-index Comparison Visualization

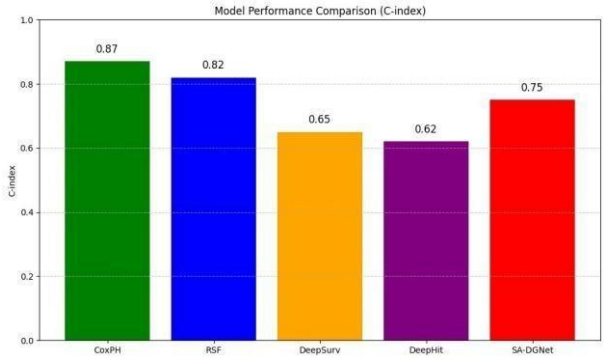


Fig. 6: Model Performance Comparison on METABRIC
Dataset (C-index)

TABLE II: Technique-wise C-index Accuracy on METABRIC
Dataset

Technique	Dataset	Accuracy (C-index)
CoxPH	METABRIC	0.87
RSF	METABRIC	0.82
SA-DGNet	METABRIC	0.75
DeepSurv	METABRIC	0.65
DeepHit	METABRIC	0.62

VII. CONCLUSION

This study explores survival analysis on the METABRIC dataset by examining the divergent methodologies of Cox proportional hazards (CoxPH), random forest survival (RFS), and deep learning-based SA-DGNet, each representing distinct paradigms in predictive modeling. Our primary objective was to analyze the effectiveness of each technique in predicting survival outcomes in breast cancer patients.

The results clearly show that classical models like CoxPH and RFS achieved higher performance than advanced deep learning models on this data set. CoxPH achieved the lowest level of discordance, reflected in a C-index of 0.8719 0.8719, demonstrating its strength in structured clinical data sets due to its simplicity and interpretability. RFS Shows increased divergence, as indicated by a lower C-index of 0.8249, benefiting from its ensemble nature and robustness to overfitting.

Although SA-DGNet was designed with gated layers and dual attention mechanisms (TASA and SDPSA) to capture temporal and feature-level dependencies, its performance was slightly lower (C-index of 0.7590), indicating that deep models may require richer, larger, or multimodal data to outperform classical baselines.

- Classical techniques remain highly effective in survival prediction tasks, especially in realworld datasets like METABRIC.
- Deep learning models offer scalability and flexibility, but require careful tuning and larger datasets to surpass traditional methods. • Attention-based mechanisms in SA-DGNet provide valuable interpretability through timedependent risk identification.
- Future extensions could include multi-modal fusion, uncertainty modeling, and transfer learning to enhance deep models.

For future research, the CoxPH model remains a strong candidate, especially when working with structured clinical data, and can be further enhanced through feature selection, stratification, and hybrid models that fuse fundamentally different techniques to improve survival prediction.

REFERENCES

[1] C. Hong, F. Yi, and Z. Huang, "Deep-CSA: Deep Contrastive Learning for Dynamic Survival Analysis With Competing Risks," *IEEE J. Biomed. Health Inform.*, 2022.

[2] C. M. Lillelund, M. Magris, and C. F. Pedersen, "Efficient Training of Probabilistic Neural Networks for Survival Analysis," *IEEE*, 2024.

[3] H. Qi, Y. Hu, R. Fan, and L. Deng, "Tab-Cox: An Interpretable Deep Survival Analysis Model for Patients With Nasopharyngeal Carcinoma," *IEEE J. Biomed. Health Inform.*, 2024.

[4] S. Chi *et al.*, "Deep Semisupervised Multitask Learning Model for Survival Analysis," *IEEE J. Biomed. Health Inform.*, 2021.

[5] C. Cui *et al.*, "Deep Survival Analysis With Latent Clustering and Contrastive Learning," *IEEE J. Biomed. Health Inform.*, 2024.

[6] S. Qi, N. Kumar, R. Verma, and J.-Y. Xu, "Bayesian Neural Networks for Personalized Survival Prediction," *IEEE Trans. Biomed. Eng.*, 2023.

[7] Q. Zheng *et al.*, "RESurv: A Deep Survival Analysis Model to Reveal Population Heterogeneity by Individual Risk," in *Proc. IEEE BIBM*, 2022.

[8] P. Liu, B. Fu, and S. X. Yang, "HitBoost: Survival Analysis via a Multi-
Output Gradient Boosting Decision Tree Method," *IEEE Access*, 2019.

[9] W. Wang *et al.*, "DeepSurvNet: A Deep Convolutional Neural Network for Survival Analysis," *IEEE J. Biomed. Health Inform.*, 2023.

[10] Z. Zhang and K. Li, "An Attention-Based Model for Predicting Patient Survival in ICU," *IEEE Access*, 2022.

[11] X. Yang *et al.*, "Time-Aware LSTM for Dynamic Survival Prediction Using EHR Data," *IEEE J. Biomed. Health Inform.*, 2021.

[12] L. Chen *et al.*, "Transformer-Based Survival Models for Cancer Prognosis Prediction," *IEEE Access*, 2023.

[13] Y. Zhou and D. Zhang, "A Hybrid Neural Network Model for Clinical Time-to-Event Prediction," *IEEE Trans. Neural Netw. Learn. Syst.*, 2023.

[14] P. G. Poornima and A. L., "Effective Strategies and Techniques Used for Pulmonary Carcinoma Survival Analysis," in *IEEE Conf. Proc.*, 2024.

[15] S. Fotso, "Deep Neural Networks for Survival Analysis Based on a Multi-Task Framework," *arXiv preprint arXiv:1801.05512*, 2018.

[16] H. Kvamme and Ø. Borgan, "Continuous and Discrete-Time Survival Prediction with Neural Networks," *arXiv preprint arXiv:1910.06724*, 2019.

[17] C. Lee, J. Zame, J. Yoon, and M. van der Schaar, "DeepHit: A Deep Learning Approach to Survival Analysis With Competing Risks," *arXiv preprint arXiv:1806.01829*, 2020.

[18] J. Katzman *et al.*, "DeepSurv: Personalized Treatment Recommender System Using a Cox Proportional Hazards Deep Neural Network," *BMC Med. Res. Methodol.*, vol. 18, no. 1, 2018.

[19] D. Lucken *et al.*, "Survival Analysis With Longitudinal Data via Deep Recurrent Models," *J. Mach. Learn. Res.*, vol. 21, pp. 1–30, 2021.

[20] J. Huang *et al.*, "SurvFormer: Transformer for Survival Analysis," in *ICML Workshop on Computational Biology*, 2022. [21] M. Liu *et al.*, "Deep Learning for Survival Analysis: A Review," *MedRxiv*, 2024.

[22] A. Srinivas *et al.*, "NeuralSurv: Deep Survival with Bayesian Uncertainty," *arXiv preprint arXiv:2505.11054*, 2025.

[23] R. Bender *et al.*, "DeepPAMM: Deep Piecewise Exponential Additive Mixed Models for Time-to-Event Data," *arXiv preprint arXiv:2202.07423*, 2022.

[24] A. R. Bose *et al.*, "FE-DeepSurv: Feature Enhanced DeepSurv Neural Network," *Comput. Biol. Med.*, 2025.

[25] S. Mehta *et al.*, "Transfer Learning for Small Sample Survival Prediction," *arXiv preprint arXiv:2501.12421*, 2025.

[26] S. M. Noman, Y. M. Fadel, M. T. Henedak, N. A. Attia, E. G. Eltasawi, and W. Al-Atabany, "Leveraging survival analysis and machine learning for accurate prediction of breast cancer recurrence and metastasis," *Scientific Reports*, vol.15, Art.3728, Jan2025. :contentReference[oaicite:1]index=1

[27] T. G. Baidoo and H. Rodrigo, "Data-driven survival modeling for breast cancer prognostics: A comparative study with machine learning and traditional survival modeling methods," *PLoS ONE*, 20(4): e0318167, Apr222025.
:contentReference[oaicite:2]index=2

[28] C. Shi and S. Ioannidis, "Spectral Survival Analysis," arXiv preprint arXiv:2505.22641 (KDD2025 extended version), May2025.

8% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography

Match Groups

18 Not Cited or Quoted 6%
Matches with neither in-text citation nor quotation marks

6 Missing Quotations 2%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

4% Internet sources
5% Publications
7% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

GH raisoni COLLEGE Engineering and Management Pune

InCoWoCo
14 - 15, November 2025 | GHRCEM, Pune

IEEE Pune Section

IEEE Women in Engineering We IEEE PUNE SECTION

2025 Second IEEE International Conference for

WOMEN IN COMPUTING (INCOWOCO 2025)

14 - 15, November 2025 | Pune, Maharashtra, India

CERTIFICATE

This certificate is presented to

Paper ID 206

Chennupalli Chandrika Tirumala
UG Scholar
Department of Computer Science and Engineering,
Narasaraopeta Engineering College (Autonomous),
Narasaraopet, Andhra Pradesh, India

for presenting the research paper entitled

"Predicting Breast Cancer Survival: An Approach using Deep Learning and Machine Learning Techniques"

authored by

Soma Sekhar Kolisetty, Parlapalli Haseena, Chennupalli Chandrika Tirumala, Kodavati Jayamma, Kathi Chandra Mouli, Anitha Vulugundam, K.V.Narasimha Reddy

at the 2025 Second IEEE International Conference for Women in Engineering (INCOWOCO 2025) held at G H Raisoni College of Engineering and Management (GHRCEM), Pune, Maharashtra, India during 14 - 15, November 2025. The conference is technically co-sponsored by IEEE Women in Engineering (WiE) of Pune Section and IEEE Pune Section.

Dr. Simran Khiani
General Chair

Prof. Dr. Rajashree Jain
General Chair

Dr. R D Kharadkar
Honorary Chair

Organized by

G H RAISONI COLLEGE OF ENGINEERING AND MANAGEMENT
Domkhel Rd, Wageshwar Nagar, Wagholi, Pune, Maharashtra 412207