Lecture Notes in Networks and Systems 1001

David Pastor-Escuredo · Imene Brigui · Nishtha Kesswani · Sushanta Bordoloi · Ashok Kumar Ray *Editors*

The Future of Artificial Intelligence and Robotics

Proceedings of 5th International Conference on Deep Learning, Artificial Intelligence and Robotics, (ICDLAIR) 2023 - Progress in AI-Driven Business Decisions & Robotic Process Automation

Springer

Lecture Notes in Networks and Systems 1001

The series "Lecture Notes in Networks and Systems" publishes the latest developments in Networks and Systems—quickly, informally and with high quality. Original research reported in proceedings and post-proceedings represents the core of LNNS.

Volumes published in LNNS embrace all aspects and subfields of, as well as new challenges in, Networks and Systems.

The series contains proceedings and edited volumes in systems and networks, spanning the areas of Cyber-Physical Systems, Autonomous Systems, Sensor Networks, Control Systems, Energy Systems, Automotive Systems, Biological Systems, Vehicular Networking and Connected Vehicles, Aerospace Systems, Automation, Manufacturing, Smart Grids, Nonlinear Systems, Power Systems, Robotics, Social Systems, Economic Systems and other. Of particular value to both the contributors and the readership are the short publication timeframe and the worldwide distribution and exposure which enable both a wide and rapid dissemination of research output.

The series covers the theory, applications, and perspectives on the state of the art and future developments relevant to systems and networks, decision making, control, complex processes and related areas, as embedded in the fields of interdisciplinary and applied sciences, engineering, computer science, physics, economics, social, and life sciences, as well as the paradigms and methodologies behind them.

Indexed by SCOPUS, EI Compendex, INSPEC, WTI Frankfurt eG, zbMATH, SCImago.

All books published in the series are submitted for consideration in Web of Science.

For proposals from Asia please contact Aninda Bose (aninda.bose@springer.com).

David Pastor-Escuredo · Imene Brigui ·
Nishtha Kesswani · Sushanta Bordoloi ·
Ashok Kumar Ray
Editors

The Future of Artificial Intelligence and Robotics

Proceedings of 5th International Conference on Deep Learning, Artificial Intelligence and Robotics, (ICDLAIR) 2023 - Progress in AI-Driven Business Decisions & Robotic Process Automation

Springer

Editors
David Pastor-Escuredo
UNHCR
Geneva, Switzerland

Nishtha Kesswani
Department of Computer Science
Central University of Rajasthan
Tehsil Kishangarh, Rajasthan, India

Ashok Kumar Ray
NERIST, North Eastern Regional Institute
of Science and Technology
Nirjuli, India

Imene Brigui
Emlyon Business School
Écully, France

Sushanta Bordoloi
National Institute of Technology Mizoram
Mizoram, India

ISSN 2367-3370 ISSN 2367-3389 (electronic)
Lecture Notes in Networks and Systems
ISBN 978-3-031-60934-3 ISBN 978-3-031-60935-0 (eBook)
https://doi.org/10.1007/978-3-031-60935-0

Contents

A Comparative Study of Machine Learning Algorithms for Predicting Cardiovascular Disease

Anu Lohachab(✉) and Kuldeep Kumar

Department of Computer Engineering, National Institute of Technology, Kurukshetra, Kurukshetra, India
anulohachab@gmail.com, kuldeepkumar@nitkkr.ac.in

Abstract. Cardiovascular diseases (CVDs) are one of the primary causes of global morbidity and mortality, presenting significant healthcare challenges. The critical need for accurate prediction of CVD risk is paramount for timely and proactive intervention, yet it remains a challenge. To enhance CVD risk prediction, this study delves into a range of machine learning algorithms, encompassing supervised, and ensemble algorithms. Furthermore, recognizing the literature's limitations, our focus has been on enhancing model performance through hyperparameter tuning, implementing robust feature selection methods, and conducting thorough model evaluations. Besides, for feature selection, we utilize chi-squared tests and correlation analysis to ensure the relevance and significance of the features. Moreover, our comprehensive evaluation, spanning three diverse datasets, assesses both supervised and ensemble learning algorithms for their accuracy and generalizability. The results indicate that the K-Nearest Neighbors-based model excels, achieving 97.82% accuracy. By enhancing predictive accuracy and model robustness, our study not only contributes to improved patient-specific interventions but also aids in shaping more effective and efficient public health strategies in cardiovascular care.

Keywords: Machine Learning in Healthcare · Predictive Modeling for CVD · Supervised Learning Algorithms · Ensemble Learning Models · Healthcare Data Analysis · Patient Care Strategies

1 Introduction

Cardiovascular disease (CVD), one of the leading causes of morbidity and mortality worldwide, poses significant challenges in healthcare [2]. The complexity of CVD stems from its multifactorial nature, where factors such as age, gender, smoking habits, and genetic predispositions interplay to influence an individual's risk. Indeed, recent advances in data mining and neural network applications have paved the way for more sophisticated assessments of CVD risks [1,3]. However, accurate risk quantification is complicated by the heterogeneity of these

D. Pastor-Escuredo et al. (Eds.): ICDLAIR 2023, LNNS 1001, pp. 1–11, 2024.
https://doi.org/10.1007/978-3-031-60935-0_1

contributing factors, underscoring the need for more nuanced and robust predictive models that can integrate and analyze a broad spectrum of risk indicators.

Nevertheless, building upon the intricate dynamics of cardiovascular disease, the emphasis on early detection and the creation of predictive models is pivotal in reducing CVD-related fatalities. To this, employing artificial intelligence, particularly in cardiac disease detection systems, significantly bolsters the efficacy of established models for CVD detection and prediction [15]. For instance, longitudinal studies using multivariate regression analysis are instrumental in constructing these risk prediction models [13]. Furthermore, in the literature (e.g., [6,10,12,14]), numerous studies have explored various classification techniques, spanning supervised, unsupervised, hybrid, and deep learning methodologies for CVD prediction. Despite these, a critical analysis reveals significant areas for improvement, particularly concerning hyperparameter optimization, the implementation of more effective feature selection strategies, and the thoroughness of model evaluation processes. These underlying issues are key contributors to the reduced accuracy observed in existing predictive models, underscoring a need for methodological enhancements in this domain. Specifically, such improvements are essential for achieving higher precision in CVD risk prediction and advancing the overall effectiveness of these models.

Accordingly, in this study, a variety of machine learning algorithms are utilized to predict CVD risk from historical data. To ensure the robustness and generalizability of our findings, our analysis includes three distinct datasets: the Statlog Heart Disease Database [11], the UCI Cleveland Database [5], and the Kaggle Database. The selection of each dataset is based on its unique attributes and relevance to CVD risk factors, facilitating a thorough evaluation of the predictive capabilities of our models in diverse clinical contexts. The primary contributions of this paper are delineated as follows:

- This paper presents a comparative study that evaluates a diverse range of machine learning algorithms across three distinct datasets. Such a holistic approach aims to provide insights into the strengths and weaknesses of each algorithm in different clinical contexts and suggest an approach for relatively more robust CVD risk prediction models.
- This study addresses the complexities of CVD risk factors, resulting in an enhancement of the accuracy of the underlying algorithms; specifically, with the K-Nearest Neighbors (KNN)-based model, we achieved an accuracy of 97.82%, a relatively better performance.

The remainder of this paper is organized as follows: Sect. 2 reviews literature in the field of cardiovascular disease prediction. Section 3 details the methodology and study design for CVD prediction. In Sect. 4, we discuss experimental outcomes, highlighting the significant improvements achieved in model accuracy and, subsequently, predictive capabilities. Finally, Sect. 5 concludes the paper and outlines potential future directions.

2 Related Works

This section briefly discusses some of the existing works dealing with CVD prediction, encompassing supervised, unsupervised, and hybrid approaches. A brief comparison is provided in Table 1.

2.1 Supervised Machine Learning-Based Approaches

Supervised machine learning algorithms have been widely employed for heart disease prediction, demonstrating promising results. Shah et al. [12] investigated the effectiveness of various supervised learning algorithms, including Random Forest, KNN, Decision Trees, and Naive Bayes, using the Cleveland database from the UCI Machine Learning Repository and selected a subset of 14 attributes. Among the employed algorithms, KNN exhibited the highest accuracy rate of 90.78%. Katarya et al. [6] summarized supervised machine learning algorithms, including Naive Bayes, Random Forest, Support Vector Machines (SVMs), and Decision Trees, and emphasized feature selection before prediction. In another study, Tasnim et al. [14] employed two datasets, the UCI Cleveland dataset, and the Statlog Hungarian dataset, to evaluate the performance of supervised machine learning algorithms. They implemented Principal Component Analysis (PCA) and Chi-Square feature selection techniques to reduce the dimensionality of the data. The combination of the Random Forest Classifier and PCA yielded the most favorable accuracy, reaching 92.85%. Mohan et al. [8] investigated the performance of supervised machine learning algorithms. They applied a range of algorithms, including K-Nearest Neighbors, Random Forest, Naive Bayes, and Logistic Regression(LR). LR emerged as the relatively better algorithm, achieving an accuracy rate of 90.2%.

2.2 Unsupervised Machine Learning-Based Approaches

Though notable unsupervised machine learning-based approaches for CVD prediction exist, limited accessibility restricts our review to a select few. In their work, Islam et al. [4] addressed the challenge of dataset dimensionality reduction using PCA and further enhanced their approach by incorporating Genetic Algorithms with unsupervised k-means clustering. Their experimentation, conducted on the UCI Machine Learning heart disease dataset, yielded an accuracy of 94.06% . In a separate study, Rajalakshmi et al. [10] proposed a novel approach to cardiovascular disease prediction. Their methodology combined K-Means Clustering, Weighted Associative Classifier, and Decision Tree C5.0 algorithms applied to a carefully curated dataset containing 11 features. The outcome of their research demonstrated a high accuracy of 94.54% in predicting cardiovascular disease.

2.3 Hybrid Machine Learning-Based Approaches

Several studies have explored hybrid machine learning approaches for cardiovascular disease prediction. Kavitha et al. [7] implemented machine learning

Table 1. A Brief Comparative Analysis of ML Approaches for CVD Prediction

Reference	Dataset	Algorithms	Feature Selection/Engineering	No. of Features	Accuracy
Shah et al. [12]	UCI Cleveland	Random Forest, KNN, Decision Trees, Naive Bayes	Not specified	14	90.78%,
Tasnim et al. [14]	UCI Cleveland, Statlog	NB, KNN, SVM, LR, DT, Xgboost	PCA, Chi-Square	14 & 12 respectively	92.85%
Islam et al. [4]	UCI heart disease	Hybrid Genetic algorithm with K-Means	PCA	2(after transformation)	94.06%
Rajalakshmi et al. [10]	UCI Cleveland	K-Means with Decision Tree C5.0 and Weighted Associative Classifier	Not specified	11	94.54%
Kavitha et al. [7]	UCI Cleveland	Random Forest, Decision Tree and a Hybrid Model	Not specified	14	88.7%
S.Mohan et al. [9]	UCI Cleveland	HRFLM (Random Forest and Linear Model)	Not specified	13	88.7%
Our Study	UCI Cleveland, Statlog & Kaggle	Supervised Models(Random Forest, KNN, Logistic Regression, Decision Tree) & Ensemble Models(AdaBoost, CatBoost)	Correlation Analysis& Chi Square test	13, 11 & 11 clinical feautures respectively	97.82%

algorithms using the UCI Cleveland dataset, employing decision trees, random forests, and a hybrid model combining random forests and decision trees. The hybrid model exhibited an accuracy rate of 88.7% in predicting cardiovascular disease. Mohan et al. [9] also employed machine learning-based models on the UCI Cleveland dataset, utilizing a hybrid HRFLM approach that combines random forest (RF) and linear model (LM). This hybrid approach demonstrated an accuracy of 88.7% in predicting heart disease.

Nevertheless, a brief review of the selected works suggests that there is potential for further exploration in various aspects, including hyperparameter optimization, more effective feature selection techniques, and more comprehensive model evaluation.

3 Methodology and Study Design

In this section, we elaborate on the essential phases of our study, as depicted in Fig. 1. These phases align with the typical stages found in machine learning-based research. However, our study features distinct adaptations, notably in hyperparameter tuning, dataset diversity, and a wide range of algorithm considerations, making it distinctive from the highlighted existing works (Table 1).

1. **Data Acquisition:** In this phase, we collect datasets from various sources, including the UCI Cleveland dataset [5], Statlog heart disease datasets [11], and the Kaggle database.
2. **Data Preprocessing:** In this phase, the datasets undergo preprocessing. The Interquartile Range (IQR) method is utilized for outlier detection and removal. This method segments the data into quartiles. The first quartile (Q1) indicates the value below which 25% of the data lies, while the third quartile (Q3) marks the value below 75% of the data. The IQR is calculated as:

$$IQR = Q3 - Q1 \tag{1}$$

Subsequently, the lower and upper bounds for identifying outliers are computed using Eqs. 2 and 3, respectively.

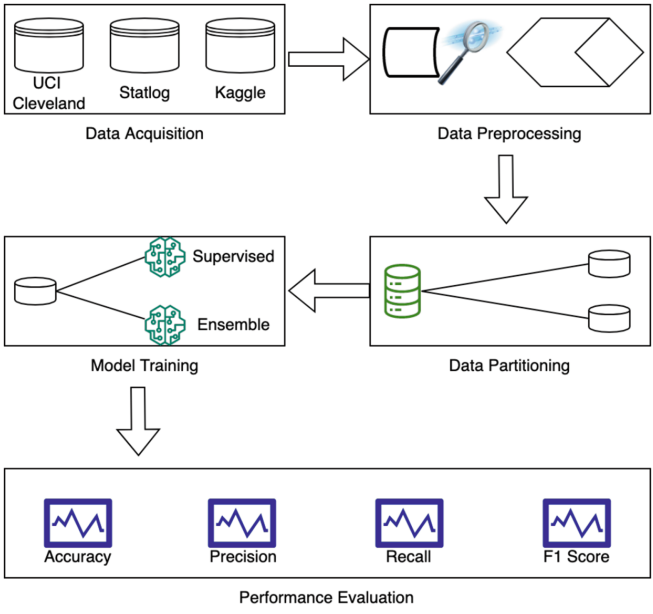

Fig. 1. Overview of Study Methodology Phases

$$\text{Lower\_limit} = Q1 - 1.5 \times IQR \tag{2}$$

$$\text{Upper\_limit} = Q3 + 1.5 \times IQR \tag{3}$$

For feature selection, correlation analysis, and the Chi-Square test are employed. These methods assess the dependency between attributes and the output class label. A higher Chi-Square value suggests a stronger association. The most relevant features are selected based on these methods.

3. **Data Partitioning:** The typical 80:20 split ratio is chosen to provide a substantial amount of data for training while reserving a significant portion for model validation. Ensuring this representative split helps mitigate the risks of overfitting and underfitting, thus enhancing the model's ability to generalize to unseen data.
4. **Model Training:** The training phase involves instructing machine learning algorithms-based models, which include both supervised and ensemble learning methods, to learn from the training data. The supervised methods employed include:
   - Random Forest (RF): An ensemble of decision trees, employing bagging to enhance model robustness against overfitting.
   - K Nearest Neighbours (KNN): A non-parametric method that classifies data points based on the majority vote of their neighbors, with k representing the number of neighbors considered.
   - Logistic Regression (LR): A regression model where the dependent variable is categorical, ideal for binary classification tasks.
   - Decision Trees (DT): A model that segments the data into subsets based on feature values, forming a tree-like structure of decisions.

   Similarly, ensemble learning algorithms such as AdaBoost and CatBoost are included. These methods aggregate multiple base models to enhance predictive performance, often resulting in improved accuracy and stability. A typical formulation of an ensemble-based model can be written as follows:

$$\text{Ensemble Model} = \sum_{i=1}^{N} \alpha_i \times \text{Base Model}_i \tag{4}$$

   where α_i are the weights assigned to each base model, determined through the ensemble learning algorithm. These weights reflect the importance or confidence of each base model within the ensemble.
5. **Performance Evaluation:** The evaluation phase is critical in assessing the effectiveness of the trained models. This involves analyzing various metrics, each providing different insights into the model's predictive capabilities. The metrics employed in this study are:

Table 2. Attributes Availability Across Datasets

Attribute	Cleveland Dataset	Statlog Dataset	Kaggle Dataset
ID (Patient's ID)	×	×	✓
Age (Patient's Age)	✓	✓	✓
cp (Chest Pain Type)	✓	✓	×
Sex (Patient's Gender)	✓	✓	✓
Hypertension (presence of Hypertension)	×	×	✓
oldpeak (Exercise induced ST Depression)	✓	✓	×
Heart Disease (History of Heart Disease)	×	×	✓
trestbps (Blood Pressure while Resting)	✓	✓	×
Ever_married (marital status)	×	×	✓
thalach (Maximum Heart Rate attained)	✓	✓	×
work_type (Employement status)	×	×	✓
chol (Serum Cholesterol)	✓	✓	×
exang (Exercise-Induced Angina)	✓	✓	×
Residence_type (Rural or Urban)	×	×	✓
restecg (Resting Electrocardiographic Results)	✓	✓	×
avg_glucose_level (average glucose level in blood)	×	×	✓
fbs (Blood Sugar Level in the time of Fasting)	✓	✓	×
thal (Blood Disorder i.e. Thalassemia)	✓	×	×
bmi (body mass index)	×	×	✓
Slope (ST segment's slope during peak exercise)	✓	✓	×
smoking_status	×	×	✓
ca (Major Vessel's count)	✓	×	×
Target/Class/Stroke (CVD Prediction)	✓	✓	✓

- Precision: It is calculated as the proportion of correctly predicted positive observations (TP) to the total predicted positives (TP + FP).
- Recall (Sensitivity): Recall quantifies the fraction of TP that the model successfully identifies from all actual positive cases (TP + FN).
- Accuracy: It assesses the overall accuracy of the model by comparing the sum of correct predictions (TP and TN) with the total number of predictions made.
- F1-Score: It is the harmonic mean of precision and recall and becomes particularly insightful for datasets with imbalanced class distributions or when the impact of false positives and negatives is distinct.

Table 3. Performance Metrics of Supervised and Ensemble ML Models on Various Datasets

Dataset	Model Type	Model Used	Accuracy	Recall	Precision	F1-Score
UCI Cleveland	Supervised	KNN	0.978	0.94	0.97	0.98
UCI Cleveland	Supervised	DT	0.82	0.94	0.96	0.83
UCI Cleveland	Supervised	RF	0.87	0.87	0.93	0.87
UCI Cleveland	Supervised	LR	0.86	0.87	0.93	0.87
UCI Cleveland	Ensemble	AdaBoost	0.89	0.94	0.96	0.89
UCI Cleveland	Ensemble	CatBoost	0.80	0.94	0.93	0.91
Statlog Heart	Supervised	KNN	0.95	0.95	0.94	0.95
Statlog Heart	Supervised	DT	0.94	0.94	0.94	0.94
Statlog Heart	Supervised	RF	0.96	0.97	0.96	0.97
Statlog Heart	Supervised	LR	0.92	0.92	0.89	0.92
Statlog Heart	Ensemble	AdaBoost	0.932	0.95	0.92	0.93
Statlog Heart	Ensemble	CatBoost	0.938	0.94	0.94	0.94
Kaggle	Supervised	KNN	0.95	0.95	0.96	0.95
Kaggle	Supervised	DT	0.93	0.94	0.93	0.94
Kaggle	Supervised	RF	0.96	0.96	0.98	0.97
Kaggle	Supervised	LR	0.92	0.92	0.96	0.92
Kaggle	Ensemble	AdaBoost	0.92	0.93	0.95	0.93
Kaggle	Ensemble	CatBoost	0.93	0.94	0.94	0.94

Overall, our approach in this study begins with the meticulous collection of data, followed by its systematic partitioning. We then train a diverse range of machine learning-based models, emphasizing both accuracy and robustness. Finally, these models are rigorously evaluated using established metrics, ensuring their reliability and effectiveness.

4 Experiments and Discussions

4.1 Experiments: Setup and Datasets

Due to resource limitations, we utilized Google Colab notebooks for our experimental setup. Besides, as stated before, our study focuses on analyzing three distinct datasets, each aimed at detecting cardiac disease. For a comparison of attribute distribution and availability, refer to Table 2, which provides an overview of each dataset's attributes. Basically, we employed the UCI Cleveland Dataset, which comprises 13 clinical features and a binary target variable, to signify the presence (1) or absence (0) of cardiac disease. Similarly, the Statlog Heart Disease Dataset utilizes 11 clinical attributes, along with a binary target

variable that follows the same coding scheme as the UCI dataset. Finally, the dataset sourced from Kaggle also employs 11 clinical features and a binary target variable, consistent with the previous datasets.

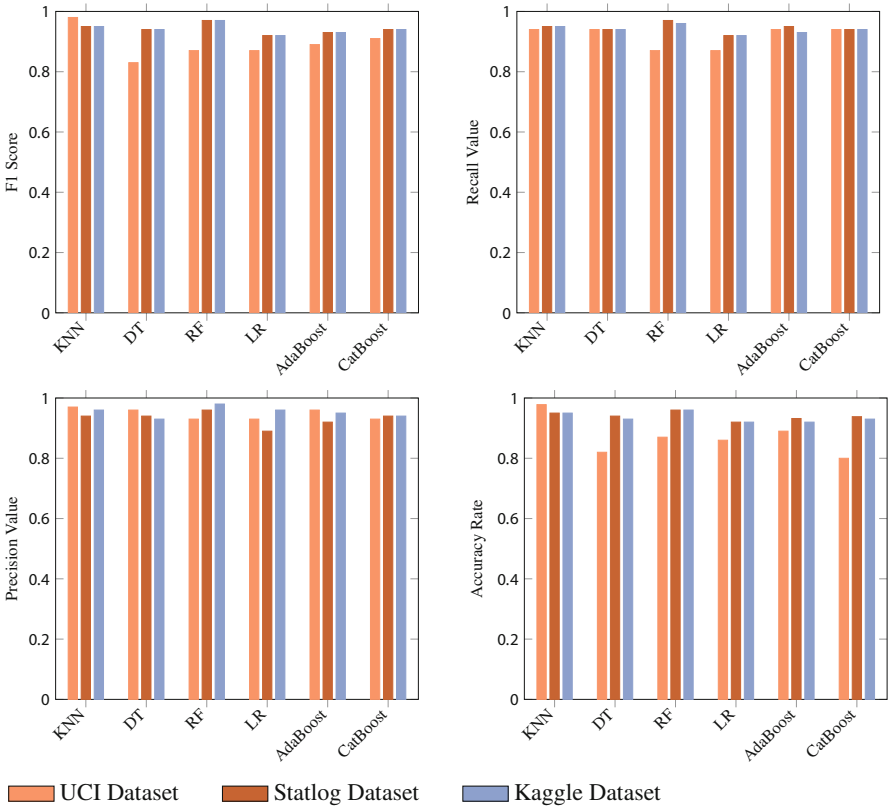

Fig. 2. Comparison of Machine Learning Models: F1 Score, Recall, Precision, and Accuracy Across Multiple Datasets

4.2 Discussions: Performance Evaluation

The metrics considered for evaluating the performance of our study are detailed in Sect. 3. Based on these metrics, Table 3 presents the results of our analysis. A key finding is that the KNN-based model, specifically configured with a leaf size of 45, Minkowski distance metric (with $p = 1$), uniform weights, and considering 5 neighbors, achieved the highest accuracy in the UCI Cleveland Dataset. This suggests that for this dataset, the KNN-based model's specific configuration plays a crucial role in its predictive performance. In contrast, the Random Forest Model initialized with a random seed value of 760, demonstrated superior accuracy on the Statlog and Kaggle Heart Disease datasets. This highlights the

model's robustness and effectiveness in handling different types of heart disease data. Further, in the realm of Ensemble Learning Models, the AdaBoost-based model, with a base Random Forest Classifier consisting of 100 trees, achieved an 89% accuracy rate on the UCI Cleveland dataset. However, it was the CatBoost-based model, trained with a learning rate of 0.9, that outperformed AdaBoost, reaching an impressive accuracy rate of 93.8% on the Statlog and 93% on the Kaggle dataset. This indicates that CatBoost's advanced algorithms and learning rate optimization offer significant advantages in predictive accuracy. Figure 2 provides the comparative analysis of the various Machine Learning Models under consideration. According to our results, KNN-based model achieves the highest accuracy for the UCI Cleveland Dataset. Additionally, this model also demonstrates the best precision for the same dataset. The highest F1 Score in the UCI Cleveland dataset context is also attributed to the KNN-based model. On the other hand, the Random Forest-based model shows its strength in the Statlog Heart Disease dataset by achieving the highest recall value. To sum up, it can be said that these findings provide important insights into the performance of different models across various datasets, highlighting the need for careful model selection based on specific performance metrics.

5 Conclusion

In this paper, we conducted a thorough analysis of CVD prediction, leveraging widely accepted machine learning models. Our study encompassed three diverse datasets, enabling a detailed assessment of the robustness and generalizability of our predictive models. Among the various machine learning-based models tested, the KNN-based model demonstrated notable effectiveness, achieving an accuracy rate of 97.82%. From the Ensemble Learning Models standpoint, the CatBoost-based model also exhibited significant efficacy, with an accuracy rate of 93.8%. These results underscore the potential of our models as highly effective tools in supporting clinical decision-making, contributing valuable insights to the field of CVD prediction. However, it is also important to acknowledge that our study was conducted under certain assumptions and with relatively clean data sets. While the findings indicate a promising direction, they may not fully capture the complexities encountered in more realistic clinical scenarios. Therefore, further investigation might be needed. Future research should aim to explore more diverse and less curated datasets and consider the impact of real-world variables and conditions on the performance of these models. Additionally, future efforts may focus on feature engineering and the integration of additional clinical data sources to enhance model performance and utility. This approach will not only validate the effectiveness of the algorithms in practical settings but also refine their applicability in predicting CVD with greater accuracy and reliability. Our study, therefore, serves as an exploratory step towards more nuanced and comprehensive cardiovascular disease prediction research, acknowledging the ongoing journey toward achieving truly robust and generalizable models.

References

1. Durairaj, M., Revathi, V., et al.: Prediction of heart disease using back propagation MLP algorithm. Int. J. Sci. Technol. Res. **4**(8), 235–239 (2015)
2. Franco, M., Cooper, R.S., Bilal, U., Fuster, V.: Challenges and opportunities for cardiovascular disease prevention. Am. J. Med. **124**(2), 95–102 (2011)
3. Gandhi, M., Singh, S.N.: Predictions in heart disease using techniques of data mining. In: 2015 International Conference on Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE), pp. 520–525. IEEE (2015)
4. Islam, M.T., Rafa, S.R., Kibria, M.G.: Early prediction of heart disease using PCA and hybrid genetic algorithm with k-means. In: 2020 23rd International Conference on Computer and Information Technology (ICCIT), pp. 1–6. IEEE (2020)
5. Janosi, A., Steinbrunn, W., Pfisterer, M., Detrano, R.: Heart disease. https://archive.ics.uci.edu/dataset/45/heart+disease, Accessed July 2023. https://doi.org/10.24432/C52P4X
6. Katarya, R., Srinivas, P.: Predicting heart disease at early stages using machine learning: a survey. In: 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), pp. 302–305. IEEE (2020)
7. Kavitha, M., Gnaneswar, G., Dinesh, R., Rohith Sai, Y., Sai Suraj, R.: Heart disease prediction using hybrid machine learning model. In: 2021 6th International Conference on Inventive Computation Technologies (ICICT), pp. 1329–1333. IEEE (2021)
8. Mohan, N., Jain, V., Agrawal, G.: Heart disease prediction using supervised machine learning algorithms. In: 2021 5th International Conference on Information Systems and Computer Networks (ISCON), pp. 1–3. IEEE (2021)
9. Mohan, S., Thirumalai, C., Srivastava, G.: Effective heart disease prediction using hybrid machine learning techniques. IEEE Access **7**, 81542–81554 (2019)
10. Rajalakshmi, K., Nirmala, K.: Heart disease prediction with mapreduce by using weighted association classifier and k-means. Indian J. Sci. Technol. **9**(19), 1–7 (2016)
11. UCI Machine Learning Repository. Statlog (Heart). https://archive.ics.uci.edu/dataset/145/statlog+heart, Accessed July 2023. https://doi.org/10.24432/C57303
12. Shah, D., Patel, S., Bharti, S.K.: Heart disease prediction using machine learning techniques. SN Comput. Sci. **1**, 1–6 (2020)
13. Stevens, D., et al.: Modelling of longitudinal data to predict cardiovascular disease risk: a methodological review. BMC Med. Res. Methodol. **21**(1), 1–24 (2021)
14. Tasnim, F., Habiba, S.U.: A comparative study on heart disease prediction using data mining techniques and feature selection. In: 2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), pp. 338–341. IEEE (2021)
15. Yan, Y., Zhang, J.-W., Zang, G.-Y., Jun, P.: The primary use of artificial intelligence in cardiovascular diseases: what kind of potential role does artificial intelligence play in future medicine? J. Geriatr. Cardiol. JGC **16**(8), 585 (2019)

Deep W-Net: DNN for Spatial Saliency Prediction in Video Frames

Gaurav Sharma(✉) and Maheep Singh

Computer Science and Engineering, NIT Uttarakhand, Srinagar, India
{gauravsharma.phd2020,maheepsingh}@nituk.ac.in

Abstract. Deep neural networks (DNN) have recently been utilized to recognize saliency in images and movies. Various models have been put forth to anticipate spatial, temporal, and spatiotemporal saliency. This paper proposes a new approach for estimating spatial saliency in video frames. Our suggested approach, which achieves excellent results on various measuring parameters, is inspired by U-net architecture. We tested our model using video frame extracts and achieved satisfactory outcomes. Results on data sets with still imagery are also examined. The proposed model does not utilize any transfer learning approaches during any part of training or testing.

Keywords: U-net · Spatial saliency · Video data sets · Transfer learning · Deep Neural Network

1 Introduction

Video saliency is identifying the area that draws human attention for observation. While some pixels are in the background, most conspicuous components are concentrated in foreground regions. In image/video compression, analysis, summarization, Etc., traditional image/video saliency prediction/ recognition is utilized. While visual saliency varies on the area of the subject and may sometimes be different for everyone, detecting it in films and images has proven difficult.

Our work presents a DNN trained on frame level using a range of datasets. The model receives an associated frame for each frame. The classic U-Net architecture utilized for medical image segmentation served as an inspiration for this work's architecture. The notion of U-net architecture was intended to be applied to border domain [1,2]. To improve the precision and efficiency of video saliency detection in dynamic and complex visual environments, our work is motivated by the need to understand and model human visual attention using deep learning, as well as to advance various multimedia applications, improve user experiences, and contribute to advancements in fields like autonomous systems, security, and healthcare.

The paper is structured as follows. In Sect. 2, related works are reviewed. The proposed architecture is explained in Sect. 3. The experimental findings and data sets used for the training and testing portion are given in Sect. 4. The conclusion is presented in Sect. 5.

D. Pastor-Escuredo et al. (Eds.): ICDLAIR 2023, LNNS 1001, pp. 12–21, 2024.
https://doi.org/10.1007/978-3-031-60935-0_2

2 Related Work

For still pictures, a method for predicting saliency by comparing a multi-scale center-surrounding feature comparison with color, orientation, and intensity was developed by Itti et al. [3]. Harel et al. [4] applied graph theory to forecast picture saliency, following Itti et al. Background and foreground clues are crucial for predicting saliency. Yang et al. [5] employed graph-based manifold ranking to determine how comparable different image elements were using background and foreground signals. Their connection to the seed determined the saliency. It is advantageous to use image details and region-based attributes to determine picture saliency. Zhang et al. employed a model based on a multiple graph where the graph was constructed using picture attributes, and visual rarity was calculated for saliency assessment [6]. Zhang et al. [7] a model in which a local tree-structured low-rank representation technique obtained background. To improve salient object detection, some studies attempt to create universal frameworks from frameworks that already exist [8]. Wang et al. [9] mention where attention stimuli were utilized to examine saliency in stereoscopic images using fixation maps to identify salient objects.

Video saliency prediction considers motion information compared to saliency prediction in still images. Liu et al. created a super pixel-based model, also known as a saliency tree for video sequences [10]. According to Leboran et al. [11] computational approach, sensitive traits can be described by high-order statistical structure. Fang et al. proposed a video saliency prediction model that combined spatial and temporal saliency utilizing statistical uncertainty indicators [12]. To determine spatiotemporal saliency, Kin et al. used a steady-state distribution of walkers to represent random walks [13]. Li et al. [14] calculated the video saliency in videos by calculating the multivariate Gaussian data's lossy encoding length. Seo et al. employed a space-time local steering kernel to extract features from a video sequence [15]. Motion and spatial edges were considered by Wang et al. [16] for saliency assessment and treated as characteristics. Using object segmentation, Tu et al. model [17] included the object signature they discovered.

There are two aspects to the video saliency challenge: semantic segmentation and salient object identification. Semantic segmentation finds numerous things in videos and treats each equally, but salient object detection only finds one object in a video. Shen et al. [18] introduced a trajectory clustering approach employing sub-modular optimization for video saliency prediction; later, Shen et al. [19] employed Taylor expansion to create a lower-order function for segmenting images.

Many studies have been conducted due to the rapid progress of deep learning techniques in computer vision. Zhao et al. [20] created a multi-context DNN that considered local and global context information and used them to model a unified network. A fully convolutional neural network-based multitask deep learning network was proposed by Li et al. [21]. By extending their technique for films with handcrafted low-level features and deep contrast characteristics, Li et al. [22]. A DNN model that learned from both local and global features was

proposed by Wang et al. [23]. Zang et al. [24] developed an approach by expressly penalizing the inaccuracy of the localization. For segmenting videos, Li et al. presented a neighborhood reversible flow and a complementary convolutional neural network [25]. The author of [26] introduced a methodology for employing visual saliency to address the issue of photo cropping. Wang et al. [27] employed hierarchical saliency information to address the saliency prediction issue.

2.1 Recent Work

A DNN was introduced by Wong et al. [28] to account for spatial and temporal saliency. Using an auto-encoder, Li et al. [29] suggested a DNN framework. RNN with ConvolutionalLSTM was utilized by Song et al. [30] to learn spatiotemporal information. The authors of [31] suggested an RNN that is flow-guided for video saliency detection. Saliency identification in images utilizing object proposal and omitting long-term temporal information was introduced by Guo et al. [32]. Shi et al. [33] mention a bi-directional method for extracting context information. Yang et al. [34] presented a bi-directional FCN to gather skeleton context information. W-Net architecture proposed by [35] resembles the English alphabet "W" and it uses a multi layer encoder and decoder for spatial saliency detection. The proposed model's idea and these investigations significantly differed based on diverse tasks. Our study suggests a Deep W-Net architecture influenced by conventional U-net architecture to determine spatial saliency in video frames. To learn spatial saliency, we created multi-layer encoders and decoders. In the first layer, all of the frame's pixels were used as features, and transfer learning was not used. In Sect. 4, experimental findings demonstrated that our model performs satisfactorily over various measuring parameters and data sets.

3 Proposed Model

This Section outlines the proposed Deep W-Net for predicting spatial saliency in videos. The proposed architecture is based on U-net [1] and W-Net [35]; nevertheless, we made several structural adjustments and evaluated the experimental findings to develop the proposed model. Google Colab is used for implementation, while Python was used on a local machine for frame extraction, resizing, and renaming of videos and image processing. Without applying transfer learning, every video frame pixel was considered input in our model. On any testing frame, no training has been conducted. Before modeling, testing videos were manually segregated.

3.1 Proposed Architecture

We employ an encoder and decoder to down-sample and up-sample the frames under U-net [1]. In contrast to U-net's concatenation, which has been performed between up-sampling and second down-sampling, multiple steps of down-sampling and up-sampling are performed similarly. Two stages of down-sampling

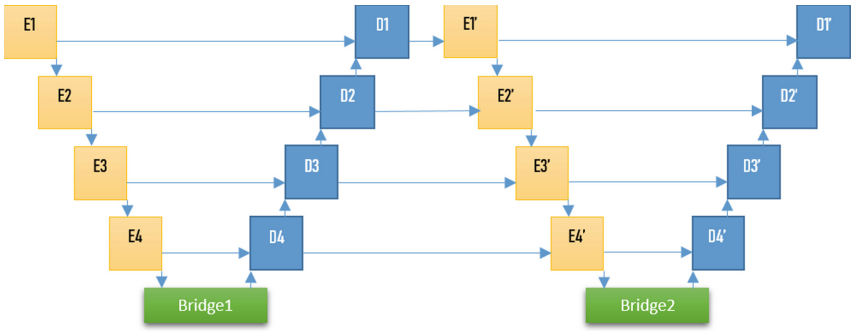

Fig. 1. Proposed Architecture (Deep W-Net)

and two of up-sampling make up the suggested model. E1, E2, and so on stand for encoder blocks, and the D series stands for decoder blocks, as seen in Fig. 1. Bridge1 is the convolution of the output of the E4 encoder performed by convBlock, as seen in Fig. 2. The convolution performed by E4's encoder is Bridge2. Convolutional transpositions of Bridges 1 and 2 are taken by D4 and D4', respectively. The convolution results of convBlock's (Fig. 2) output are concatenated to the convolution transpose output of D blocks; convolution from D blocks are concatenated to convolution from E1'–E4'.

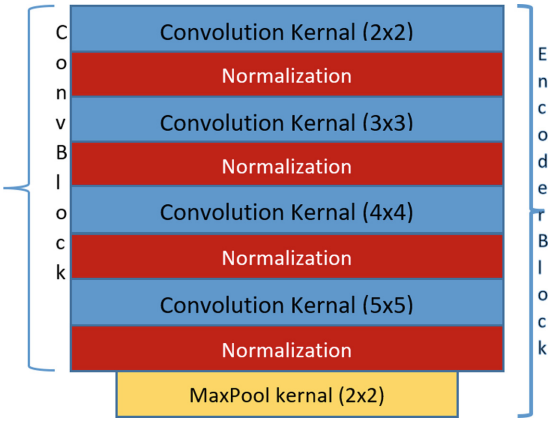

Fig. 2. Encoder Block

3.2 Encoder Block

Six encoder blocks are present in the suggested model. Four convolution layers and a normalizing layer follow each convolution in the encoder block. Figure 2 shows a convolution block. Each convBlock is followed by a Maxpooling layer to downsample. The input for the first convolution layer, C1, is 224 × 224 pictures (frames), and the kernel size is 2×2. C2 receives input from C1 and has a 3×3 kernel. Using a 4×4 kernel size in C3, the output of C2 is used as the input. C4 accepts the features from C3 as input. In C4, the kernel measures 5×5. The ability to extract low-level to high-level features is aided by kernel size variation. For all convolutional layers, padding is the same, but the stride is 1. Bridge1 is accomplished from E4; however, there is no Maxpooling. Similarly, Bridge2 is produced by E4's output. In various phases of our model, 2 × 2 kernel size max pooling is performed.

3.3 Decoder Block

The up-sampling decoder component is shown in Fig. 3. Convolutional transposition layer, concatenation, and convBlock are included in this block. ConvBlock is the encoder block's convBlock. Concatenation is used to map feature information from the subsequent convolutions of the encoder block when convolutional transpose is used for up-sampling. Convolutional transposition uses a 2 X 2 pixel kernel with a 2 pixel stride.

3.4 Training Process

The proposed model was trained using the DAVSOD, SegTrack, and ViSal data sets. 80 % data is used for training, while 20 % is used for testing. Learning parameters involves minimizing the loss function. The loss function is the binary cross entropy. The proposed model, Deep W-Net, uses the Adam optimizer to identify the optimized network parameter while minimizing the loss function. End-to-end training is conducted using a batch size of 3 and a learning rate 0.001.

4 Experimental Results

4.1 Evaluation Methodology

We conducted experiments on DAVSOD [36], SegTrack [37], and ViSal [38] datasets to assess the performance of our model Deep W-Net. Two hundred twenty-six videos with 23938 frames can be found in the DAVSOD data set. 1000 frames make up SegTrack, while 17 videos from various sources are found in ViSal. F-measure, S-measure, and Mean Absolute Error (MAE) are utilized for performance comparison. F-measure is calculated as in Eq. 1.

$$F - measure = 2 \times P \times R \div (P + R) \tag{1}$$

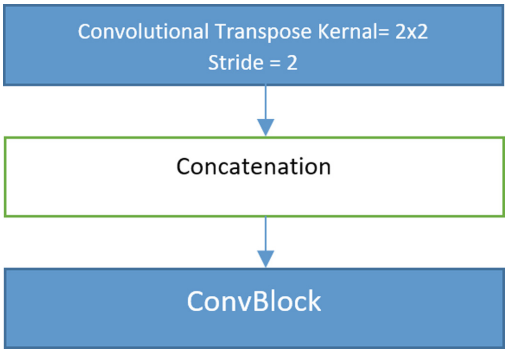

Fig. 3. Decoder Block

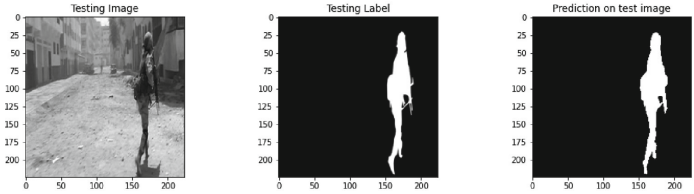

Fig. 4. Visual Comparison

where Precision is calculated as in Eq. 2.

$$Precision(P) = TP/(TP + FP) \tag{2}$$

where FP stands for false positive and TP for true positive. Recall is calculated as in Eq. 3.

$$Recall(R) = TP/(TP + FN) \tag{3}$$

where FN stands for false negative and TP for true positive. The region-aware structural similarity (S-measure) metric is calculated as:

$$Sr = \sum_{k=1}^{K} w_k * ssim(k) \tag{4}$$

wk is the weight assigned to each block, K is the total number of blocks in the sub-image, and ssim is the structural similarity metric. MAE is calculated as,

$$MAE = 1/n \sum_{i=1}^{n} |x_i - x| \tag{5}$$

where n is number of samples(frames) and absolute error is
$|x_i - x|$ where x and x_i are ground truth and model's result respectively.

Table 1. Quantitative Comparisons Between Our Model and Other Sota Vsod Models Using The F-Max, S-Measure, and Mae Metrics. The Top Three Results are Respectively Highlighted in RED, GREEN, and BLUE.

Dataset	SegTrackV2			ViSal			DAVSOD		
Metric / Model	F-Max	S-Measure	MAE	F-Max	S-Measure	MAE	F-Max	S-Measure	MAE
Ours	0.871	0.901	0.015	0.892	0.924	0.021	0.688	0.762	0.066
ESTI 2021	0.860	0.891	0.017	0.952	0.952	0.013	0.651	0.746	0.086
MGA 2019	0.821	0.865	0.030	0.933	0.936	0.017	0.640	0.738	0.084
CPD 2019	0.778	0.841	0.023	0.941	0.94	0.016	0.608	0.724	0.092
DLVS 2018	-	-	-	0.851	0.881	0.048	0.521	0.657	0.121
SCOM 2018	0.764	0.815	0.030	0.831	0.762	0.122	0.464	0.599	0.220
SFLR 2017	0.745	0.804	0.037	0.779	0.814	0.062	0.478	0.624	0.132
SGSP 2017	0.673	0.681	0.124	0.677	0.706	0.165	0.426	0.577	0.207
STBP 2017	0.640	0.735	0.061	0.622	0.629	0.163	0.410	0.568	0.160
MSTM 2016	0.526	0.643	0.114	0.673	0.749	0.095	0.344	0.532	0.211
GFVM 2015	0.592	0.699	0.091	0.683	0.757	0.107	0.334	0.553	0.167
SAGM 2015	0.634	0.719	0.081	0.688	0.749	0.105	0.370	0.565	0.184
MB+M 2015	0.554	0.618	0.146	0.692	0.726	0.129	0.342	0.538	0.228
RWRV 2015	0.438	0.583	0.162	0.440	0.595	0.188	0.283	0.504	0.245
SPVM 2014	0.618	0.668	0.108	0.700	0.724	0.133	0.358	0.538	0.202
TIMP 2014	0.573	0.644	0.116	0.479	0.612	0.170	0.395	0.563	0.195
SIVM 2010	0.581	0.605	0.251	0.522	0.606	0.197	0.298	0.486	0.288

4.2 Comparative Research Employing Techniques that Are Already Proposed

The proposed model is contrasted with other cutting-edge models in this section. Comparison with ESTI [50], MGA [49], CPD [39], DLVS [28], SCOM [40], SFLR [41], SGSP [42], STBP [43], MST [44], GFVM [38], SAGM [16], MB+M [45], RWRV [13], SPVM [46], TIMP [47], SIVM [48] is shown in Table.1. In this work, F-measure, S-measure, and MAE are used to compare experimental results. Figure 4 presents our results for comparison between prediction and ground truth. The proposed model Deep W-Net performs best on DAVSOD and SegTrack Dataset with F-Max as 0.688 and 0.871, S-Measure as 0.762 and 0.901, and MAE as 0.066 and 0.015, respectively. Overall, the effectiveness of the proposed model, Deep W-net, is convincingly evidenced by the F-Max, S-measure, and MAE values shown in Table 1.

5 Conclusion

This paper presented a novel approach (Deep W-Net) for detecting salient video objects. The proposed model uses an encoder and decoder block to map information from the encoder to the decoder and from the decoder to the encoder. We used each pixel from the normalized frames as a feature to train the proposed model. Extensive experiments have been done on three benchmark datasets, and