computers

MDPI

Article

Cervical Cancer Diagnosis Using Stacked Ensemble Model and Optimized Feature Selection: An Explainable Artificial Intelligence Approach

Abdulaziz AlMohimeed [1], Hager Saleh [2,*], Sherif Mostafa [2], Redhwan M. A. Saad [3] and Amira Samy Talaat [4]

[1] College of Computer and Information Sciences, Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh 13318, Saudi Arabia
[2] Faculty of Computers and Artificial Intelligence, South Valley University, Hurghada 84511, Egypt
[3] College of Informatics, Midocean University, Moroni 8722, Comoros
[4] Computers and Systems Department, Electronics Research Institute, Cairo 12622, Egypt; amtalat@yahoo.com
* Correspondence: hager.saleh@fcih.svu.edu.eg

Abstract: Cervical cancer affects more than half a million women worldwide each year and causes over 300,000 deaths. The main goals of this paper are to study the effect of applying feature selection methods with stacking models for the prediction of cervical cancer, propose stacking ensemble learning that combines different models with meta-learners to predict cervical cancer, and explore the black-box of the stacking model with the best-optimized features using explainable artificial intelligence (XAI). A cervical cancer dataset from the machine learning repository (UCI) that is highly imbalanced and contains missing values is used. Therefore, SMOTE-Tomek was used to combine under-sampling and over-sampling to handle imbalanced data, and pre-processing steps are implemented to hold missing values. Bayesian optimization optimizes models and selects the best model architecture. Chi-square scores, recursive feature removal, and tree-based feature selection are three feature selection techniques that are applied to the dataset For determining the factors that are most crucial for predicting cervical cancer, the stacking model is extended to multiple levels: Level 1 (multiple base learners) and Level 2 (meta-learner). At Level 1, stacking (training and testing stacking) is employed for combining the output of multi-base models, while training stacking is used to train meta-learner models at level 2. Testing stacking is used to evaluate meta-learner models. The results showed that based on the selected features from recursive feature elimination (RFE), the stacking model has higher accuracy, precision, recall, f1-score, and AUC. Furthermore, To assure the efficiency, efficacy, and reliability of the produced model, local and global explanations are provided.

Keywords: cervical cancer; stacking ensemble; explainable AI (XAI)

check for updates

Citation: AlMohimeed, A.; Saleh, H.; Mostafa, S.; Saad, R.M.A.; Talaat, A.S. Cervical Cancer Diagnosis Using Stacked Ensemble Model and Optimized Feature Selection: An Explainable Artificial Intelligence Approach. *Computers* **2023**, *12*, 200. https://doi.org/10.3390/computers12100200

Academic Editor: Paolo Bellavista

Received: 16 August 2023
Revised: 24 September 2023
Accepted: 26 September 2023
Published: 7 October 2023

1. Introduction

The World Health Organization in 2020 estimated that there were 604,000 new cervical cancer (CC) cases, and it was the fourth most common malignancy among women. Nearly ninety percent of the 342,000 deaths from cervical cancer in 2020 were from low- and middle-income countries. Cervical cancer is six times more likely in HIV-positive women than in HIV-negative women [1]. In low- and middle-income nations, these preventative measures are not widely accessible, and cervical cancer is typically not discovered until it has advanced and symptoms appear. Additionally, access to cancer surgery, radiation, and chemotherapy for malignant lesions may be restricted in these countries, which raises the mortality rate from cervical cancer. Effective interventions at different times of life could help lower the high death rate from cervical cancer worldwide (the standard age rate for women in 2020 is 13.3 per 100,000) [1]. Cervical cancer disease can be diagnosed based on a variety of features. Cervical cancer can be cured if detected early and treated promptly. Doctors may need help to diagnose it quickly and accurately. As a result, it is critical to use computerized technology in cervical cancer disease diagnostics to help doctors diagnose patients more quickly and precisely.

Artificial intelligence (AI) is revolutionizing the diagnosis and therapy of diseases [2]. AI systems can also observe patients over time, giving doctors helpful information about possible therapies and allowing more accurate treatments. Overall, AI technology can transform cervical cancer treatment and detection, delivering more efficient and superior care for people who are ill.

While developing a predictive model [3], feature selection approaches reduce the amount of input variables. A crucial stage in machine learning (ML) is feature selection algorithms, whereby we determine what are the most essential features from a database to be included in a model [4,5]. Filter methods use statistical measures to rank the features based on their relevance to the target variable [6]. Wrapper approaches evaluate the utility of features through evaluation of the classifier's performance [7].

In order to improve overall performance and prediction accuracy, ensemble machine learning which utilizes multiple models could be used [8]. Combining numerous individual models is the fundamental idea behind ensemble learning. Ensemble learning comes in a variety of forms: stacking [9], boosting [10] and stacking, and bagging [11]. Stacking is a learning technique that integrates multiple models using a meta-model to enhance prediction accuracy and robustness. There are multiple forms of stacking, which include homogeneous stacking, which employs base models of the same type [12]. Contrarily, heterogeneous stacking employs models of several types as its foundation models [12].

In the ML field, an essential compromise exists between the difficulty level and the efficacy of the built models. While basic models such as linear regression offer greater interpretability and ease of explanation, complex DL and ML models [13] need more transparency. Consequently, the ability to provide a clear and understandable explanation for these complex models becomes crucial in establishing trust in their outcomes. The motivation of explainability is the lack of transparency exhibited by complex models, commonly known as black-box models, which affects trust in the model's performance and outcomes. The concept of explainability, also known as explainable AI (XAI), aims to enhance the reliability, interpretability, and understanding of model predictions. Explainability can be approached at two primary levels: global and local. Global explainability pertains to explaining the overall decision-making process across all data points [14]. On the other hand, local explainability focuses on explaining individual instances, providing a more precise and accurate explanation. The main goals of this paper are to study the effect of applying feature selection methods with stacking models for the prediction of cervical cancer, proposing stacking ensemble learning that combines different models with meta-learners to predict cervical cancer, explore the black-box of the stacking model with the best-optimized features using explainable artificial intelligence (XAI).

Our work encompasses the following key contributions:

- Examining the impact of combining feature selection techniques alongside machine learning models towards cervical cancer prediction.
- Proposing stacking ensemble learning that combines different ML with meta-learner with SMOTETomek method to predict cervical cancer.
- Assessing the effectiveness of the provided model and evaluating it against different ensemble models and conventional ML models applying different evaluation matrices.
- When compared to other models, the performance of the suggested model gave better performed with REF.
- Enhancing the proposed approach through using techniques to deliver a comprehensible explanation.

The remainder of the paper is structured as follows: Section 2 analyzes the related literature to inform about previous work and recent models established to predict cervical cancer. Section 3 outlines the methodology and how our research was conducted. Section 4 describes the results. Finally, Section 6 summarizes and provides concluding remarks on the research.

2. Related Work

In [2], the authors used RFE and the least absolute shrinkage and selection operator (LASSO) to determine the most significant attributes for cervical cancer prediction. SMOTE-Tomek handled unbalanced data. In the results, the best performance was recorded by DT with RFE.

To predict cervical cancer, the authors of [15] employed machine learning algorithms comprised of DT, LR, SVM, and KNN boosting. RF and boosting received the best performance ratings. Also, the authors used various statistical analyses on the data, such as computational complexity analysis (CCA), exploratory cervical data analysis (ECDA), and empirical consequence reports (ECP). Based on the Risk Factors dataset, the authors of [16] utilized an improved DT classifier to categorize patients with cervical cancer. No feature selection strategy was used on the proposed DT classifier for categorizing cervical cancer patients in order to choose an ideal collection of features. The authors of [17] investigated the impact of using various over-sampling, under-sampling, and over-and-under-sampling techniques to handle imbalanced data. Methods for selecting features using filters and wrappers were utilized. The findings indicate that DT had the best accuracy. In [18], the authors classified the cervical cancer diagnostic dataset as healthy or cancerous using KNN, C4.5 DT, NB, RF, and MLP. The data were subjected to data imputation to address missing values and an over-sampling technique called SMOTE to address the imbalanced dataset. The authors of [19] employed feature transformation techniques, like the sine function, log, and Z-score, together with RF and RT, on their datasets. The modified datasets were subjected to a variety of feature selection techniques in order to recognize and rank noteworthy risk variables.

In [20], datasets have been classified using the Softmax classification using the stacked autoencoder deep learning technique. Models such as SVM, DT, kNN, rotation forest, and feed forward NN use stacked autoencoder to study the softmax classification's classification results. The voting classifier integrated three prediction classifiers, DT, LG, and RF, in [21]. In addition to using the PCA approach to remove dimensions that have no impact on the model's accuracy, the SMOTE was used to solve the issue of an unbalanced dataset. In [22], the authors used an RF model with SMOTE and two feature reduction methods, recursive feature elimination and PCA. In [23], the authors applied ML on data with 23 attributes, which were collected from Shohada Hospital Tehran, Iran, during 2017–2018.

There are studies that applied imbalanced strategies to handle imbalanced classification issues. Wang et al.'s [24] proposal of a data expansion approach based on the idea of the normal distribution was an attempt to address the classification problem of imbalanced datasets. By applying a linear interpolation method based on the normal distribution between the minority sample points and the minority center, the approach enhances the minority data while avoiding the marginalization problem. The results suggest that increasing the five imbalanced datasets in this manner may lead to more accurate classification than applying the original SMOTE method's requirements. Due to the presence of multiclass data with unbalanced distributions in the majority of Industrial Internet of Things (IIoT) datasets, The deterioration in the ML-based IDS model's accuracy for identifying attacks was addressed by Le et al. [25]. In order to solve this issue, the authors proposed an intrusion detection system (IDS) for IIoT imbalanced datasets using the (XGBoost) model. According to the instance difficulty, Yu et al. [26] suggested that the instance-level rebalancing technique dynamically modifies the sample probabilities for the distribution of classes and unlabeled subclass instances. The authors described the instance difficulty as a measurement based on the instance's learning rate, which was motivated by the human learning process.

Some studies applied ensemble learning. For example, Jiang et al. [27] presented an ensemble XAI model to produce a more trustworthy factor importance ranking approach. The ensemble XAI model used the SHAP values and model performances to illustrate how multiple machine-learning models certified for six datasets performed efficiently, providing a more accurate factor ranking. Large IoT-based IDS datasets could be used, according to

Le et al. [28], to explain the predictions derived from machine learning (ML) models and boost IDS's performance for detecting attacks. The decision tree (DT) and random forest (RF) classifiers used were assessed on the two datasets, NF-BoT-IoT-v2 and NF-ToN-IoT-v2. To explain and understand the categorization decisions, authors implemented Shapley additive explanations (SHAP) alongside the explainable AI (XAI) technique. Chakir et al. [29] developed models for empirical evaluation involving homogeneous and heterogeneous ensemble techniques enabling web-based threat detection in Industry 5.0. The authors produced a heterogeneous ensemble comprising three best-performing ML algorithms using max voting and stacking methods.

3. Methodology

Figure 1 shows the main steps of prediction cervical cancer. Data processing is critical for ML analysis that includes filling missing values, dataset encoding, and min-max normalization. SMOTE-Tomek is used to handle the imbalance of the dataset and Bayesian optimization (BO) to optimize ML models. ML refers to ensemble models.

3.1. Data Description

We used from UC Irvine Machine Learning Repository (UCI) that was collected from 'Caracas University Hospital' in Caracas, Venezuela [30]. Data on 858 patients includes demographics, habits, and historical medical records and 36 features. Table 1 shows Features name and Abbreviation for cervical cancer dataset.

3.2. Data Processing

Data processing is critical for ML analysis to guarantee data quality and reduce noise, outliers, and class imbalance [31].

- Filling missing values in the cervical cancer dataset is crucial for the accuracy and reliability of ML models [32]. Methods include mode, mean, and median imputation. Mean imputation substitutes the mean value of the corresponding feature for missing data, and median imputation substitutes missing values with the median value. We remove features with 70% missing values, reduce the number of features from 36 to 32, and remove rows with more than 50% null values; the number of rows reduces from 858 to 763. Then, we applied median for filling missing values for numerical features and the mode for filling in missing values for categorical features.
- Dataset encoding converts non-numerical or categorical data into a numerical format. Encoding labels involves assigning a number value to each category of categorical data.
- We used min-max normalization, which scales the features to a fixed range.

3.3. Optimization Models

Optimization is determining the best combination of hyperparameters. It also finds the best model architecture and training parameters to maximize the model's performance on validation and testing datasets. By combining training sets and cross-validation procedures, Bayesian optimization (BO) is used to optimize ML models. The optimal collection of hyperparameters in Bayesian optimization is denoted by x^*, as illustrated by the formula [33]:

$$x^* = \arg\min_{c \in \chi} f(x) \tag{1}$$

where $f(x)$ is the objective function and c represents any value in the domain of χ.

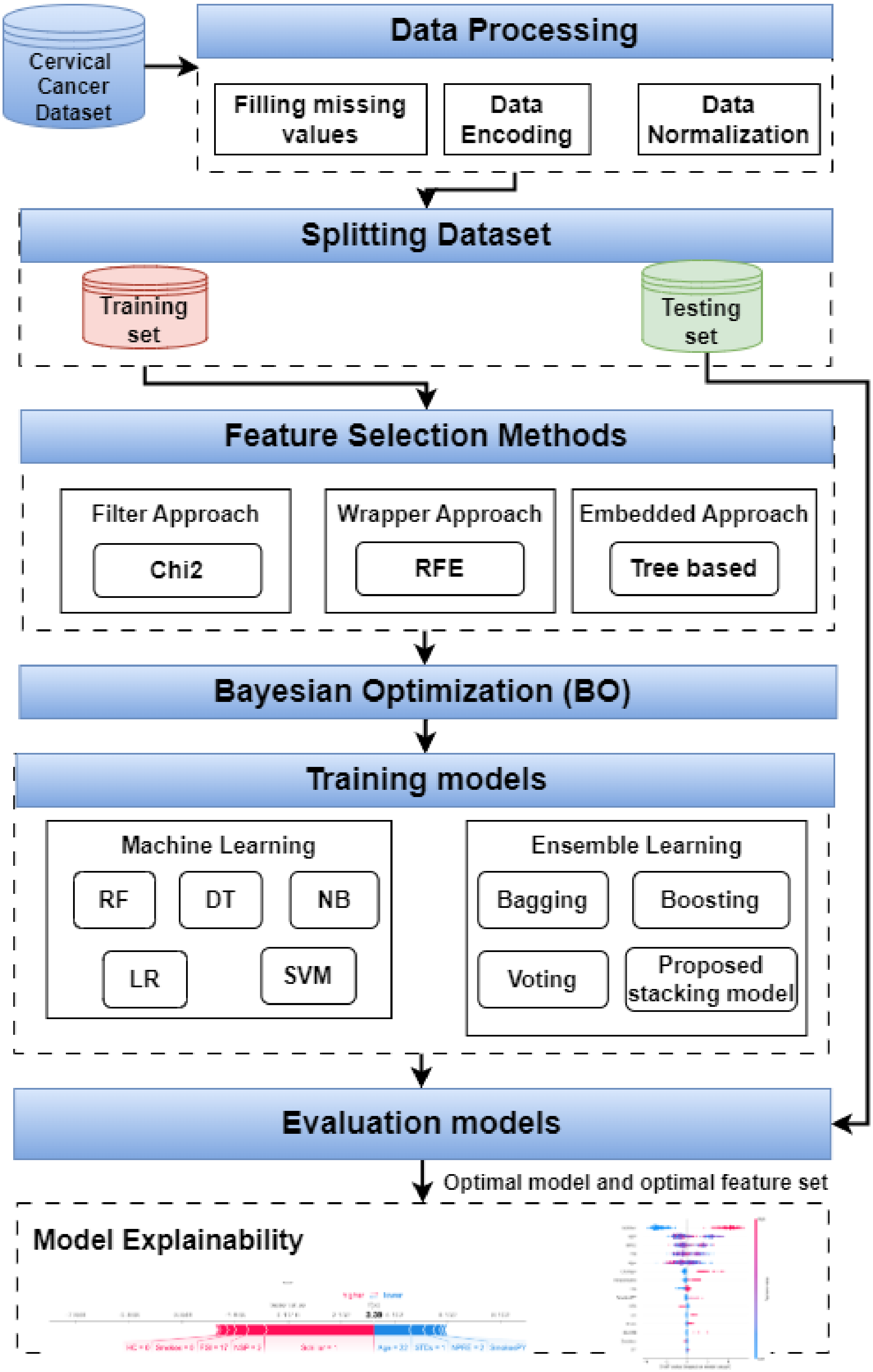

Figure 1. The main steps for predicting cervical cancer.

Table 1. Features name and abbreviation for cervical cancer dataset.

Features Name	Abbreviation
Smokes	Smokes
Hormonal Contraceptives	HC
IUD	IUD
STDs	STDs
STDs:condylomatosis	CON
STDs:vaginal condylomatosis	VC
STDs:vulvo-perineal condylomatosis	VPC
STDs:syphilis	SYP
STDs:pelvic inflammatory disease	PID
STDs:genital herpes	GH
STDs:molluscum contagiosum	MC
STDs:HIV	HIV
STDs:Hepatitis B	HB
STDs:HPV	STDs:HPV
Dx:Cancer	Dx:Cancer
Dx:CIN	Dx:CIN
Dx:HPV	Dx:HPV
Dx	Dx
Hinselmann	Hinselmann
Schiller	Schiller
Citology	Citology
Biopsy	Biopsy
Age	Age
Number of sexual partners	NSP
First sexual intercourse	FSI
Num of pregnancies	NPRE
Smokes (years)	SY
Smokes (packs/year)	SmokesPY
Hormonal Contraceptives (years)	HC
IUD (years)	IUD (years)
STDs (number)	STDs (number)
STDs: Number of diagnosis	ND
Smokes	Smokes
Hormonal Contraceptives	HC
IUD	IUD
STDs	STDs

Bayesian optimization (BO) is a model-based sequential optimization technique that efficiently optimizes costly black-box operations. The BO method involves defining the search space, selecting a surrogate model [34], determining an acquisition function, initializing the surrogate model, iterating the optimization loop, evaluating the objective function, updating the surrogate model, checking termination criteria, and returning the optimized solution. To explore the search space successfully, BO uses the surrogate and acquisition function [35]. Cross-validation (CV) uses to evaluate model performance and fine-tune hyperparameters. It involves several key steps. The dataset is split into k folds to ensure an appropriate distribution of samples. The model is then trained on k-1 folds, with one fold reserved for validation. This procedure is carried out k times. Each fold serves as a validation dataset. Using metric, such as accuracy and performance of the model is evaluated. These metrics are averaged across all k folds to estimate the overall performance. Finally, a separate test dataset is used to check the model has not overfit the training data.

3.4. Class-Imbalanced Resampling Techniques

Class imbalance is a typical problem in ML when there are far fewer members of one class than the other(s). Because ML models are inclined towards the majority class, this imbalance might have a negative impact on their performance [36]. In random over-sampling, this method replicates examples from the minority class at random until the required balance is reached [37]. SMOTE (synthetic minority over-sampling technique) creates and builds minority class samples by finding current minority class samples that are close together in feature space and then interpolates between them [38]. It chooses an instance, finds its KNN, and builds new instances along the line segments that connect them. SMOTE aids in overcoming the overfitting issues associated with random over-sampling. In our work, SMOTE-Tomek takes advantage of the synthetic minority over-sampling technique (SMOTE) to over-sample, and Tomek Link to under-sample [39,40].

3.5. Feature Selection Methods

Feature selection methods are an essential step in ML that selects the most significant features from a database to use in a model. Figure 2 shows feature selection techniques [4,5]:

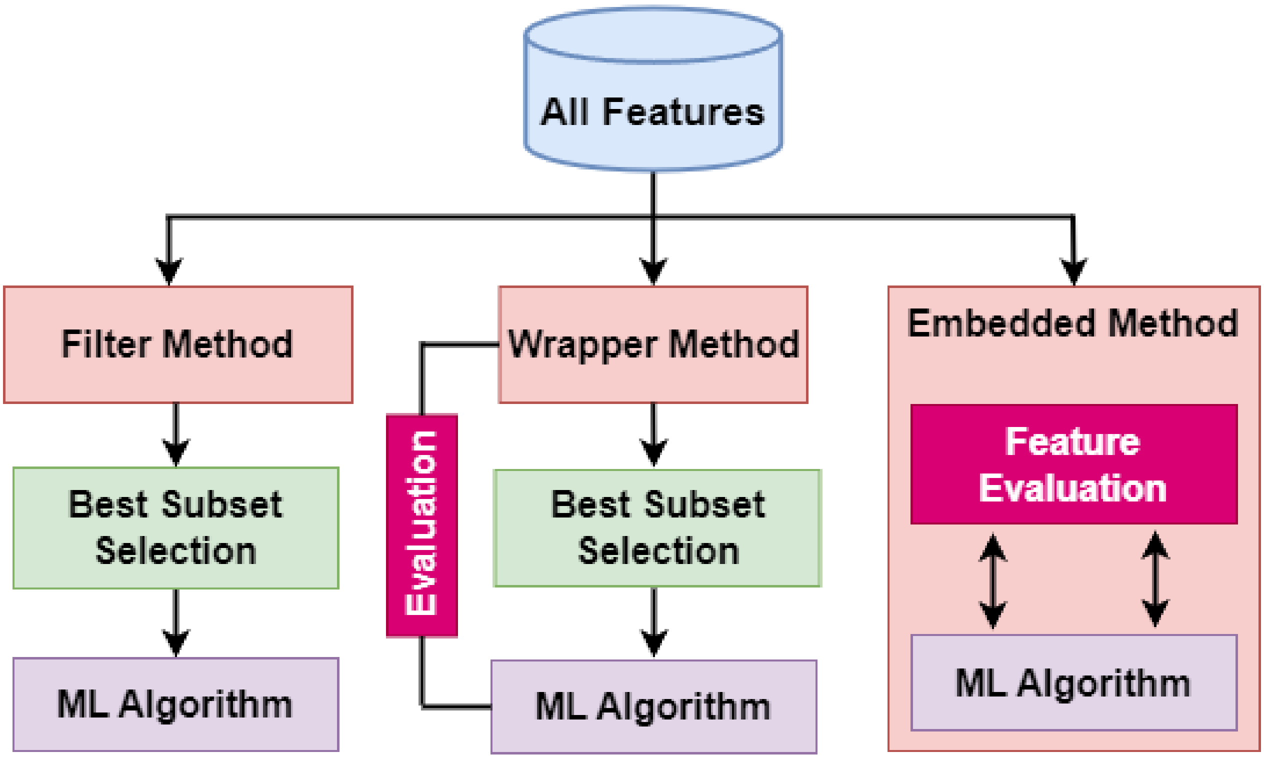

Figure 2. Feature selection techniques.

3.5.1. Filter Approach

The filter technique ranks the features according to the degree to which they are relevant to the target variable using statistical measurements. [6]. It includes the chi-square test [41] and mutual information [42]. In our study, we applied the chi-square test to select essential features. With the use of statistics, the chi-square test (Chi2) assesses the correlation between features and the target then selects the ideal number of features with the best chi-square scores. It ranks the features based on the best chi-square scores [41].

3.5.2. Wrapper Approach

Wrapper methods assess features' usefulness by evaluating the classifier's performance. The selected feature subsets serve as estimators of the model's performance, providing an indication of its predictive capability. Based on the observed model performance, the feature selection algorithm iteratively adjusts the feature subset by adding or removing features to converge upon the optimal feature subset [7]. Recursive feature elimination (RFE) [7] and forward/backward selection [43] are examples of the wrapper methods.

RFE is a feature selection algorithm that leverages an ML model to assess the importance of each feature within a given dataset. RFE operates through the following steps [7]:

- Initially, the ML model is trained using the complete set of features available in the dataset.
- Importance scores are calculated for each feature based on the feature importance attribute. These scores quantify the relative significance of each feature concerning the model's predictive performance.
- The feature with the lowest importance score is eliminated from the feature set.
- A new model is then trained using the remaining features.
- Steps 2–4 are iteratively repeated until the optimal number of features is selected.
- RFE can be used with any ML model that provides feature importance scores, such as DT, SVM, and LR.

3.5.3. Embedded Approach

The embedded approach employs a combination of ensemble and hybrid learning techniques to facilitate the process of feature selection. It operates by dynamically selecting the most optimal features throughout the learning phase. This selection is made strategically to maximize computational efficiency. By leveraging a collective decision-making mechanism, this approach outperforms both the filter and wrapper methods in terms of computational cost and classification accuracy [44]. Several embedded methods for feature selection are available, including random forest (RF), which generates several decision trees and aggregates their predictions to provide ultimate predictions. The bootstrap sampling method is used to build each decision tree in the RF, which randomly chooses a subset of features for each split. As a result of the random feature selection, the trees become more diverse and overfitting becomes diminished. The output of random forest is the class chosen by the majority of trees [45].

3.6. Class-Imbalanced Resampling Techniques

Class imbalance is a typical problem in ML when the number of instances in one class is much smaller than the number of instances in the other class(es). Because ML models are inclined towards the majority class, this imbalance might have a negative impact on their performance. To address this issue, different class unbalanced resampling strategies for rebalancing the dataset and improving model performance have been devised [36].

3.7. Machine Learning Models

Different ML techniques are used:

- Logistic Regression (LR) refers to a statistical technique that estimates outcomes according to a collection of input features. It is a generalized linear model that employs a logistic function (sigmoid function) to model the relationship between input features and the probability of result [46].

- Support Vector Machines (SVMs) operate by locating the hyperplane that effectively separates the data into two classes. The hyperplane is selected to maximize the margin, which refers to the distance between the closest points from each class and the hyperplane. The decision boundary is defined as the set of points that lie on the hyperplane. The data is transformed into a higher dimensional space using kernel functions [47].
- Decision Tree (DT) works by dividing the data into groups depending on the characteristics values. The objective is to build a tree that effectively separates the data into several classes or predicts the target variable. Every node in the tree indicates a feature, and the branches represent the potential values of the feature. The leaves represent the class label or the predicted values [48].
- Random Forest is an ensemble technique which employs many decision trees for improving the model's accuracy and robustness. Using a randomly selected subset of features and data, it functions by creating a forest of decision trees. The predictions of all trees in the forest are combined to make the final prediction [49].

3.8. Ensemble Learning Models

Ensemble learning is an ML technique that merges many models to increase overall performance and prediction accuracy [8]. The basic concept behind ensemble learning is the merging of many individual models. Here are some popular ensemble learning methods:

- Bagging is the process of simultaneously training multiple models on different subsets of the training data then combining their predictions to generate the final prediction [9]. Multiple models are independently trained on distinct subsets of the training data, and their predictions are merged to get the best possible outcome [9]. Bagging minimizes prediction variance. It averages predictions from numerous models trained on different data subsets.
- Boosting, a sequential training technique, involves sequentially training multiple models. Each model is trained to rectify the errors made by the preceding models [50]. The fundamental foundation of boosting lies in the amalgamation of weak models, which serves as the cornerstone of this technique. Through this combination, a robust and accurate ensemble model is generated. The primary advantage of boosting resides in its ability to mitigate prediction bias by iteratively adjusting the weights assigned to misclassified instances; boosting emphasizes challenging or misclassified samples during subsequent rounds of model training. This adaptive approach enables the ensemble model to enhance performance and progressively achieve more precise predictions [50].

3.9. Proposed Stacking Ensemble Model

Stacking is a learning technique integrating multiple models using a meta-model to enhance performance and robustness. The stacking technique uses the training data to train many base models, which then feed their predictions into a meta-model for generating the final prediction. There are several kinds of stacking, involving homogeneous stacking, which is used with templates of the same type as base models [12]. Heterogeneous stacking uses different types of models as base models [12]. The suggested stacking ensemble model in this work operates on two levels, as shown in Figure 3: Level 1 (multiple base learners) and Level 2 (meta-learner). Level 1: Stacking of several model outputs (training and testing stacking). Training stacking employed in Level 2 to train meta-learners (SVM, LR, and RF). Testing stacking metrics are used to evaluate meta-learners (SVM, LR, and RF).

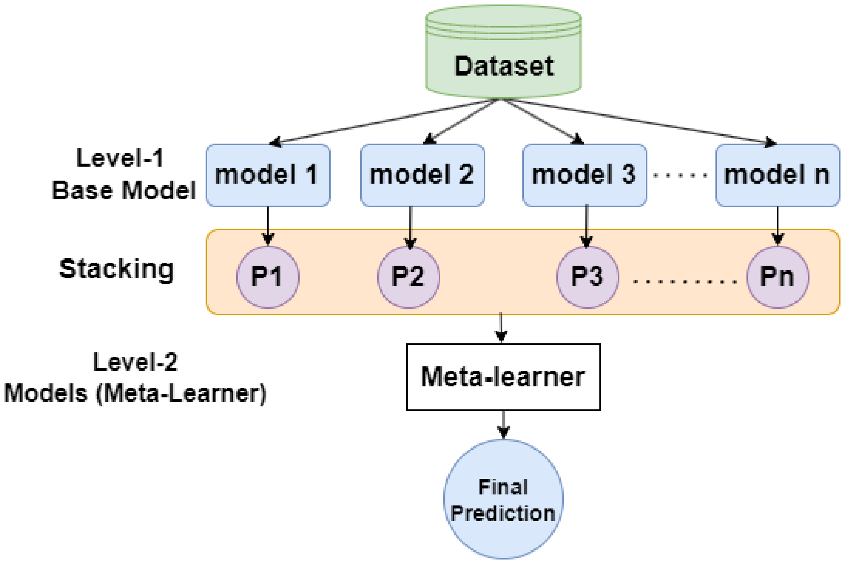

Figure 3. The proposed stacking model for predicting cervical cancer.

3.10. Explainable Artificial Intelligence XAI

XAI is an emerging field that aims to develop ML models. It can provide transparent and interpretable explanations for their predictions [51]. The two primary levels of the concept of explainability are local and global. Global explainability refers to the final decision made after considering all data points. It provides a global fidelity causal analysis. But it is limited in that it only discusses the significance of the instance level. But local explainability can explain all samples, providing a more accurate explanation [51].

XAI techniques are Shapley additive explanations that use Shapley values to explain various model outputs [52]. SHAP can be used with any black-box model. It works well when applied to certain model classes, such as tree ensembles, because of these models' inherent structure and properties. This efficiency allows for more tractable and practical computation of the Shapley values, facilitating the interpretation and understanding of model predictions. SHAP can be employed for global and local interpretability purposes [53].

3.11. Evaluating Models

Models are evaluated with many methods:

- Accuracy, precision, recall, and F1-score among are the more frequently employed metrics to measure classification performance. These metrics calculations are explained by Equations (1) to (4).
  True negative (TN) indicate that the individual is healthy and the test is negative, whereas true positive (TP) indicate if the individual is sick and the test is positive. A false positive (FP) happens when a test results in a positive result while the subject is healthy. A false negative (FN) occurs when a test yields a negative result despite the fact that the person was sick.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}. \tag{2}$$

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

$$F1\text{-}score = \frac{2 \cdot precision \cdot recall}{precision + recall} \tag{5}$$

- A receiver operating characteristic curve (ROC curve) provides a graph which depicts the efficacy of a classification model over all classification levels. The true-positive rate and false-positive rate [54] are plotted on this curve. Because the true-positive rate (TPR) is a synonym for recall, it is defined as follows:

$$TPR = \frac{TP}{TP + FN} \tag{6}$$

The following is how the false-positive rate (FPR) is determined:

$$FPR = \frac{FP}{FP + TN} \tag{7}$$

An ROC curve is generated while plotting both TPR and FPR at various categorization levels. More items are classified as positive when the classification threshold is lowered, which raises the number of both false positives and true positives. A typical ROC curve is depicted in the following figure [54]. The acronym AUC stands for “area under the ROC curve”. It measures the complete two-dimensional region below the entire ROC curve, from (0,0) to (1,1), distinguishing between classes measured by this parameter. Models with a higher AUC are better at predicting values [54].

4. Experiment Results

This subsection investigates the results of applying feature selection methods, the performance of ML and ensemble learning, and the proposed stacking models with selected features.

4.1. Experimental Setup

ML and stacking models were implemented using Scikit-learn [55]. SHAP explainers [56] was used to interpret the model that provides local and global explanations. Matplotlib.pyplot [57] library was used to plot ROC curve. SMOTE-Tomek [40] was used to combine under-sampling and over-sampling to handle imbalanced data A stratified sampling method was used to divide the dataset into two parts: 70% training and 30% testing. ML models are optimized and trained using the training set. Evaluation metrics are used to evaluate ML models, comparing stacking models to several ML models, including bagging, boosting, and voting ensemble models as well as RF, LR, DT, SVM, and NB. The results of each feature selection technique used to choose the 15 best features are presented. Results of models being applied to features chosen via REF, Chi2, and based-tree feature selection are shown.

4.2. Feature Selection Results

The experiments explore the essential features of applying feature selection methods to the cervical cancer dataset.

4.2.1. Feature Scores Based on Chi2

In Figure 4, the scores for each feature are shown after applying Chi2 to the dataset. We can see that Schiller is the most crucial feature, with 150.36 scores. Hinselmann and Citology have the second important features at 70.550 and 60.490, respectively. Dx, IUD. ND, VC, and HC.1 feature with scores between 4 and 3. HC, NPRE, Age, SmokesPY, FSI, IUD (years), and NSP are the lowest impacted features with scores below 1. ML models were applied to the 15 highest features.

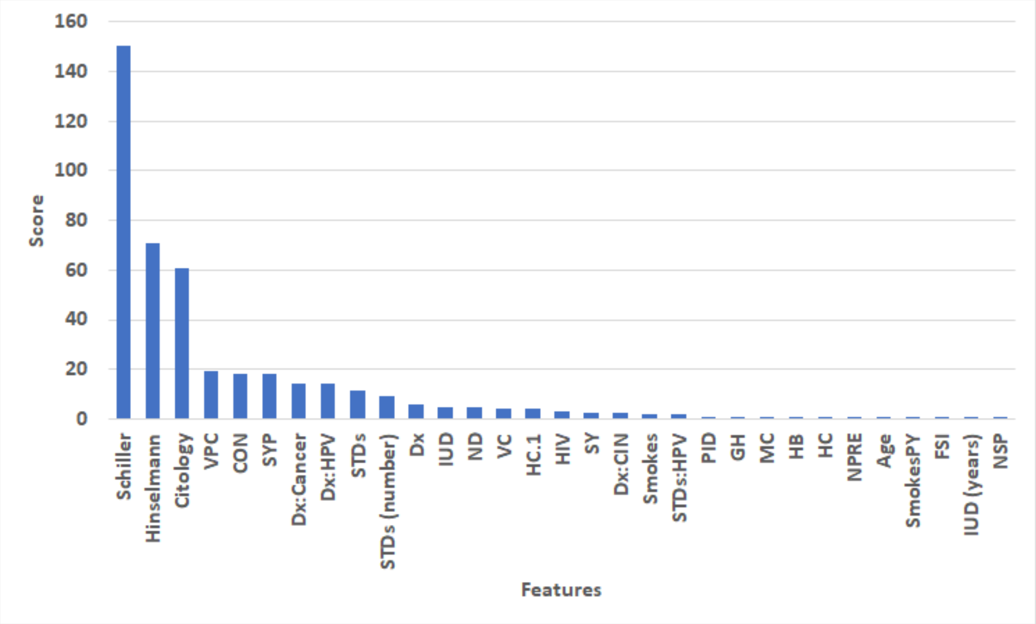

Figure 4. Feature scores based on Chi2.

4.2.2. Importance of Selected Features by Based Tree

As shown in the Figure 5, features based on based trees are ranked in importance. We can see that Schiller is the most crucial feature, with 0.293634763 importance. NPRE and Hinselmann have the second important features at 0.098876738 and 0.081442887, respectively. HIV, VC, PID, HB, STDs:HPV, and MC are the lowest impacted features with importance below 0.00003. ML models were applied to the 15 highest features.

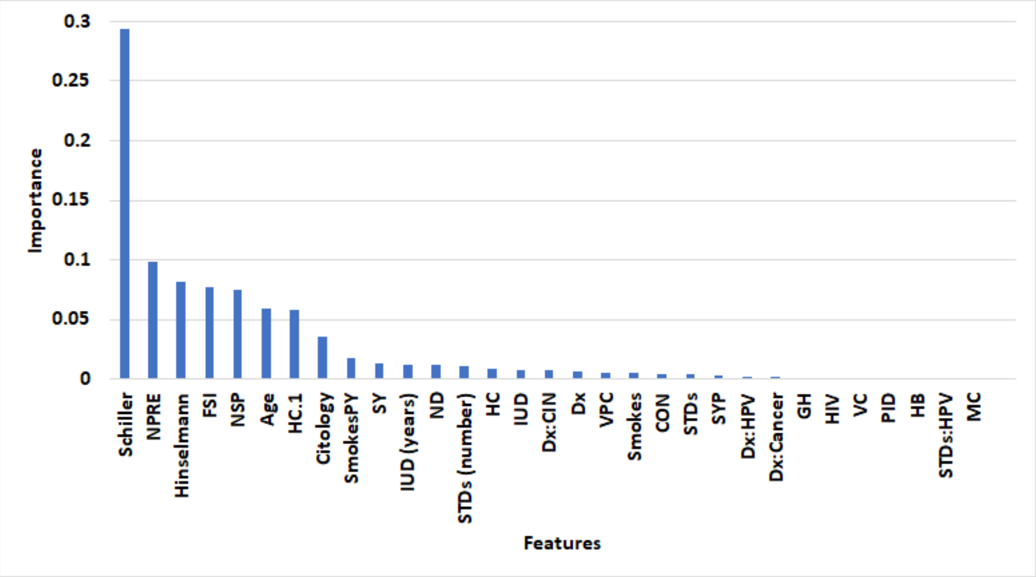

Figure 5. Importance of selected features by based tree.

4.2.3. The Ranking of Selected Features According to RFE

Figure 6 shows the ranking of features according to REF. There are 15 top features ranked 1 by REF, including SY, STDs (number), STDs, SmokesPY, Smokes, Schiller. The worst features are STDs:HPV and HB, which are ranked 7. ML models were applied to the 15 highest features.

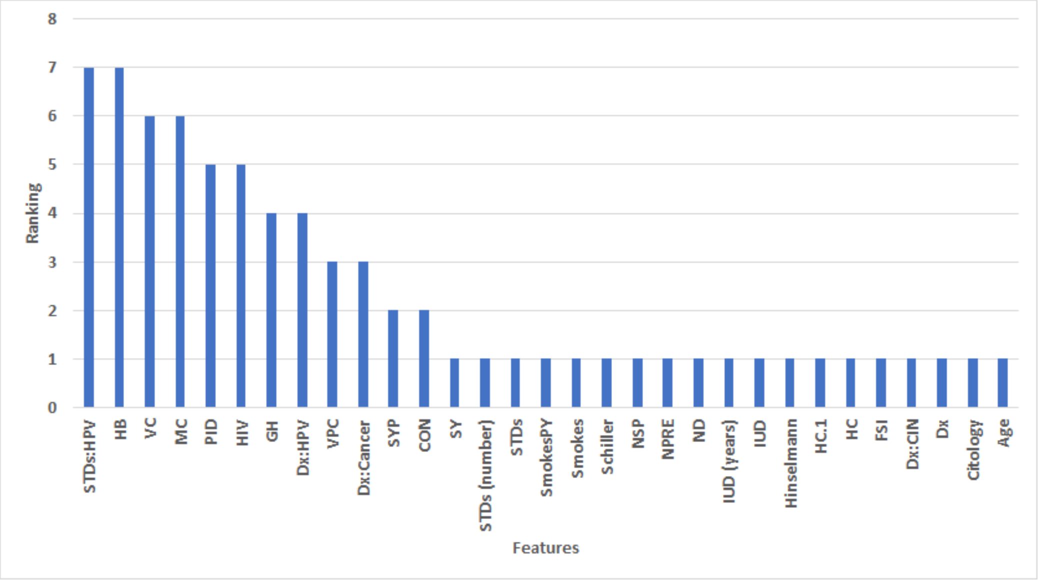

Figure 6. The ranking of selected features according to RFE.

4.3. ML Results

This section presents the results of ML models.

Results of Chi2 for Models with Selected Features

This subsection investigates the performance of ML and ensemble learning, as well as the proposed stacking models with selected features using the Chi2 algorithm.

All models are evaluated based on several metrics as shown in Table 2. The following can be observed: Regrading ML, LR and NB recorded similar performance (accuracy = 90.11, precision = 90.40, recall = 90.11, and F1-score = 90.10). The performance of the RF has improved by 2% above LR and NB. RF showed the best performance (accuracy = 92.66, precision = 92.85, recall = 92.66, and F1-score = 92.65).

Table 2. Results of Chi2 for models with selected features.

Approaches	Models	Accuracy	Precision	Recall	F1-Score
ML models	RF	92.66	92.85	92.66	92.65
	LR	90.11	90.40	90.11	90.10
	DT	91.81	91.92	91.81	91.80
	SVM	90.96	90.98	90.96	90.96
	NB	90.11	90.40	90.11	90.10
Ensemble models	AdaBoost	90.68	90.84	90.68	90.67
	Bagging	94.07	94.14	94.07	94.07
	Voting	92.09	92.23	92.09	92.08
	Stacking	96.05	96.07	96.05	96.04

Based on a comparison of ensemble models, AdaBoost showed the worst performance (accuracy = 90.68, precision = 90.84, recall = 90.68, and F1-score = 90.67). The proposed stacking model performed the best performance (accuracy = 96.05, precision = 96.07, recall = 96.05, and F1-score = 96.04) compared to other models.

The findings depicted in Figure 7 provide a comprehensive overview of the ROC curve and AUC obtained from the ML models and ensemble models, utilizing Chi2 feature selection methods. The stacking model emerges as the top performer, exhibiting an outstanding AUC of 96.045. Following closely, the bagging model achieved second the highest

AUC at 94.068. In contrast, the DT and NB model demonstrates the lowest AUC among the evaluated models, reaching 90.113. The stacking model is the most accurate, surpassing the models in classification accuracy.

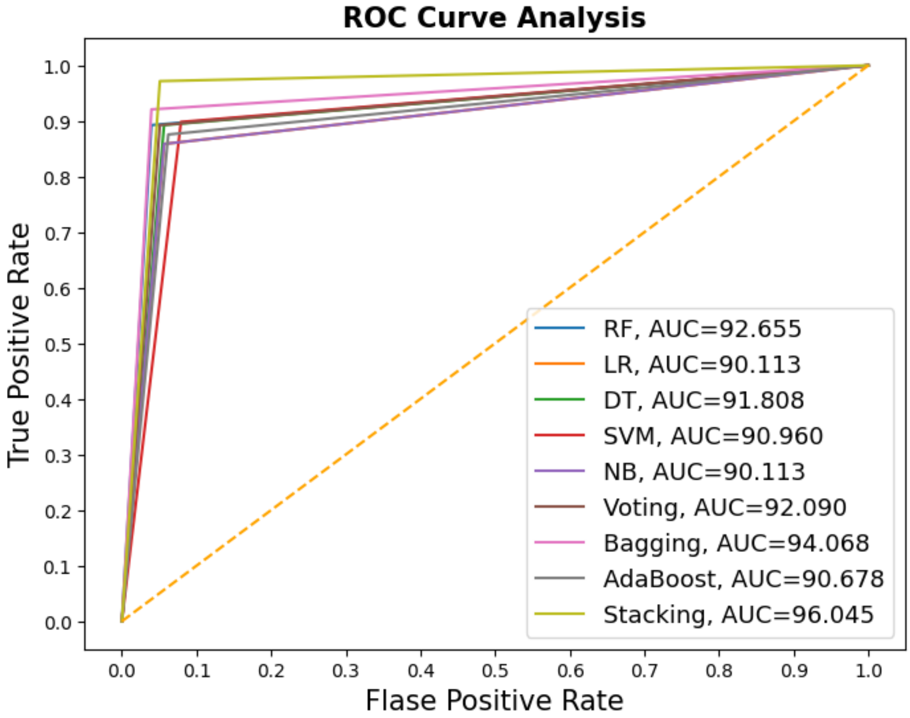

Figure 7. ROC of REF chi2 models with selected features.

4.4. Results of REF for Models with Selected Features

The effectiveness of ML and ensemble learning is examined in this subsection, along with the suggested stacking of models using particular features using the REF algorithm. As stated in Table 3, a number of measures are used to evaluate each model.

Table 3. Results of REF for models with selected features

Approaches	Models	Accuracy	Precision	Recall	F1-Score
ML models	RF	98.87	98.87	98.87	98.87
	LR	89.83	89.08	89.83	89.81
	DT	94.92	94.92	94.92	94.92
	SVM	92.09	92.18	92.09	92.09
	NB	90.11	90.40	90.11	90.10
Ensemble models	AdaBoost	94.63	94.65	94.63	94.63
	Bagging	97.46	97.53	97.46	97.46
	voting	94.92	94.92	94.92	94.92
	Stacking	99.44	99.44	99.44	99.44

The following can be observed: Regrading ML, LR recorded the lowest performance (accuracy = 89.83, precision = 89.08, recall = 89.83, and F1-score = 89.81). The performance of the RF has improved by 8% above LR. RF showed the best performance (accuracy = 98.87, precision = 98.87, recall = 98.87, and F1-score = 98.87).

Based on a comparison of ensemble models, AdaBoost showed the worst performance (accuracy = 94.63, precision = 94.65, recall = 94.63, and F1-score = 94.63). The proposed stacking model performed the best performance (accuracy = 99.44, precision = 99.44, recall = 99.44, and F1-score = 99.44) compared to other models.

The findings depicted in Figure 8 provide a comprehensive overview of the ROC curve and AUC obtained from the ML models and ensemble models, utilizing REF feature selection methods. The stacking model emerges as the top performer, exhibiting an outstanding AUC of 99.435. Following closely, the RF model achieved second the highest AUC at 98.870. In contrast, the LR model demonstrates the lowest AUC among the evaluated models, reaching 89.831. The stacking model is the most accurate, surpassing the models in classification accuracy.

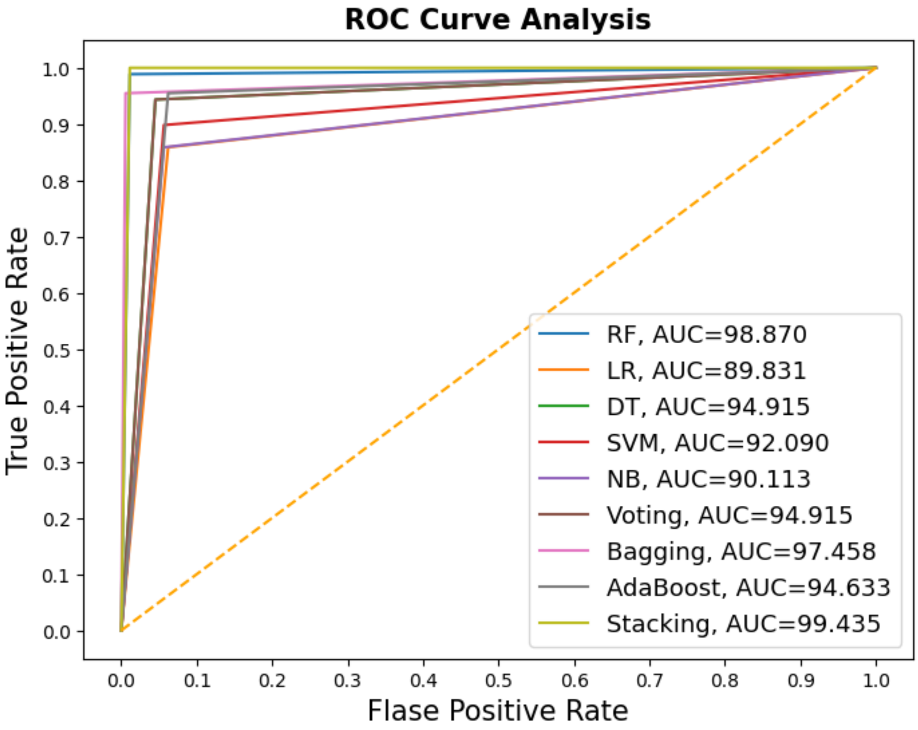

Figure 8. ROC of REF for models with selected features.

Results of Tree-Based Models with Selected Features

This subsection investigates the performance of ML and ensemble learning, as well as the proposed stacking models with selected features using tree-based models.

All models are evaluated based on several metrics as shown in Table 4. The following can be observed: Regrading ML, LR and NB recorded a minor similar performance (accuracy = 90.15, precision = 90.30, recall = 90.15, and F1-score = 90.30). The performance of RF has improved by 5% above LR and NB. RF showed the best performance (accuracy = 97.74, precision = 97.75, recall = 97.74, and F1-score = 97.74).

Table 4. Results of three-based for models with selected features.

Approaches	Models	Accuracy	Precision	Recall	F1-Score
ML models	RF	98.87	98.87	98.87	98.87
	LR	89.83	89.08	89.83	89.81
	DT	94.92	94.92	94.92	94.92
	SVM	92.09	92.18	92.09	92.09
	NB	90.11	90.40	90.11	90.10

Table 4. *Cont.*

Approaches	Models	Accuracy	Precision	Recall	F1-Score
Ensemble models	AdaBoost	94.63	94.65	94.63	94.63
	Bagging	97.46	97.53	97.46	97.46
	voting	94.92	94.92	94.92	94.92
	Stacking	99.44	99.44	99.44	99.44

Based on a comparison of ensemble models, voting showed the lowest performance (accuracy = 94.35, precision = 94.36, recall = 94.63, and F1-score = 94.63). The proposed stacking model performed the best performance (accuracy = 98.31, precision = 98.31, recall = 98.31, and F1-score = 98.31).

The findings depicted in Figure 9 provide a comprehensive overview of the ROC curve and AUC obtained from the ML models and ensemble models, utilizing tree-based feature selection methods. The stacking model emerges as the top performer, exhibiting an outstanding AUC of 98.305. Following closely, the RF model achieved second the highest AUC at 97.740. In contrast, the LR and NB models demonstrate the lowest AUC among the evaluated models, reaching 90.113. The stacking model is the most accurate, surpassing the models in classification accuracy.

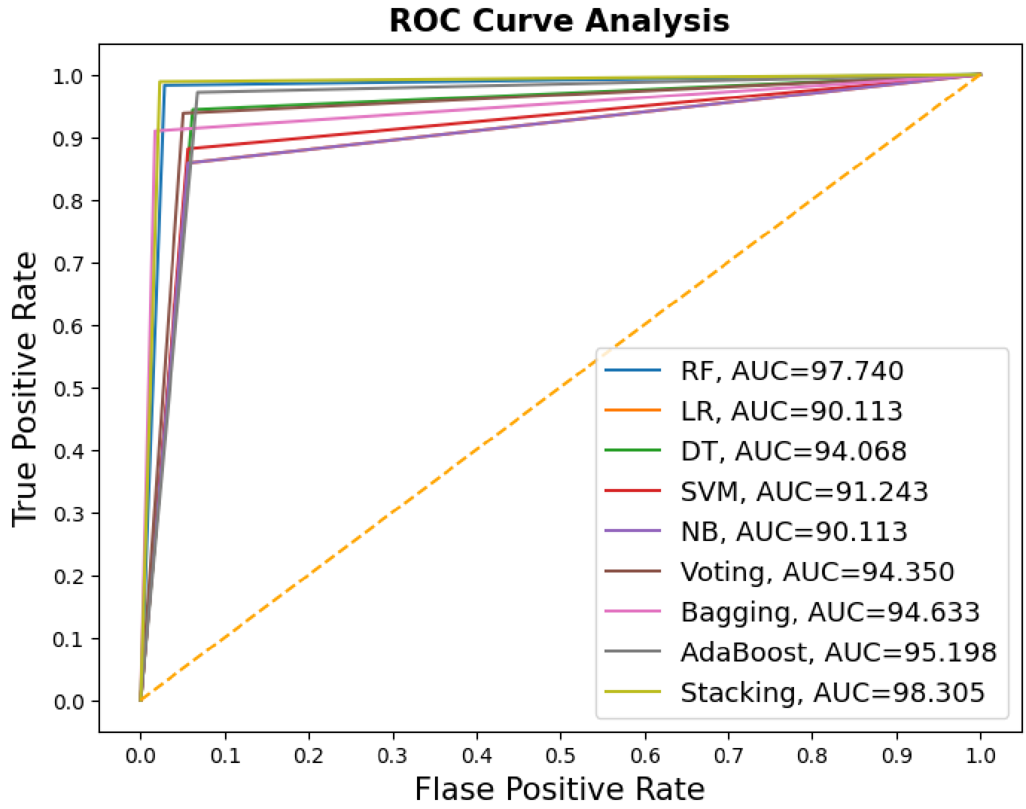

Figure 9. ROC of tree-based models with selected features.

5. Discussion

5.1. The Best Models

In this paper, we investigate the performance of ML models, the ensemble model, and the proposed stacking model with chi2, REF, and tree-based methods for feature selection. From Figure 10, it is clear that the stacking model using REF achieves the best results in comparison to chi2 and tree-based models. Stacking with chi2 was the least efficient model.

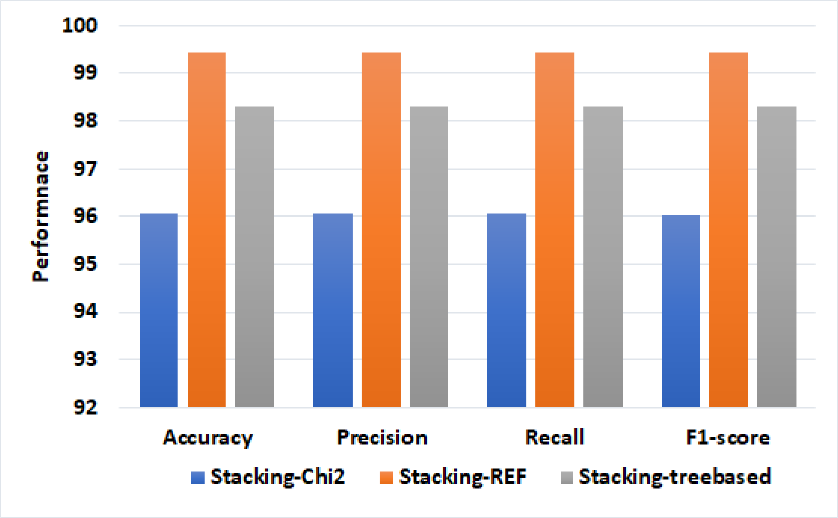

Figure 10. The best models.

5.2. Explainable Artificial Intelligence (XAI)

This section addresses global explainable (dataset level) and local explainable (instance level) output decisions. We use SHAP explainers to interpret the model provides local and global explanations. The experimental results prove that the stacking model with REF feature selection has the highest performance. We use SHAP explainers to interpret the model. The SHAP summary plot of the developed model is shown in the Figure 11. Several significant features of the entire dataset are important to the decision-making process, as shown in Figure 11. Each feature is represented by a horizontal line to show how it affects the output: blue means it causes the model to move in the lower direction, and red means it moves in the upper direction. YES or NO determination is significantly influenced by Schiller. Age, FSI, HC, and Hinselmann are considered the second essential features. STDs and DX are the lowest features that impact model decisions. Additionally, the Figure 12 shows the mean impact of each feature. Each bar represents the importance of a feature, and its length corresponds to the feature's name on the X-axis. We can see that the Schiller feature most significantly impacts model decisions.

Local explainability methods seek to clarify how the AI model arrived at a specific cause or prediction in response to a given input. This enables people to comprehend and trust the AI system's decisions. Local explainable explains why the model arrived at a particular decision for a specific sample. Therefore, we used the SHAP force plot to present the local explainability for two causes for class 0 and class 1, as shown in Figure 13A,B. The plots show the base_value based on the whole dataset, predict_proba_value, which indicates the probability according to the specific cause, feature values shown on the left, and feature contributions indicated through arrows. A probability is calculated for each instance by adding the base value and the contribution of each feature.

For example, instance with class 0 is shown in Figure 13A, which presents the force plot of cause for class 0 to present the features that are contributed for this model prediction. Based on the model's average over the entire dataset, the predicted value's base value is 2.205, and predict_proba is −1.87. The arrows represent the value of the features (NSP = 1, SmokesPY = 12, Age = 10, Schiller = 0, Smokes = 1, NPRE = 1, HC = 1). Features with a big arrow have a significant impact, while features with a short arrow have a smaller impact. Features with red arrows contribute to higher scores, and blue arrows contribute to lower scores. For example, a cause with class 1 is shown in Figure 13B due to the features contributing to the model prediction. Based on the model's average over the entire dataset, the predicted value's base value is 0.2049, and predict_proba is 9.14. The arrows represent the value of the features (NSP = 1, NPRE = 3, FSI = 17, Age = 32, Schiller = 1, HC = 1).

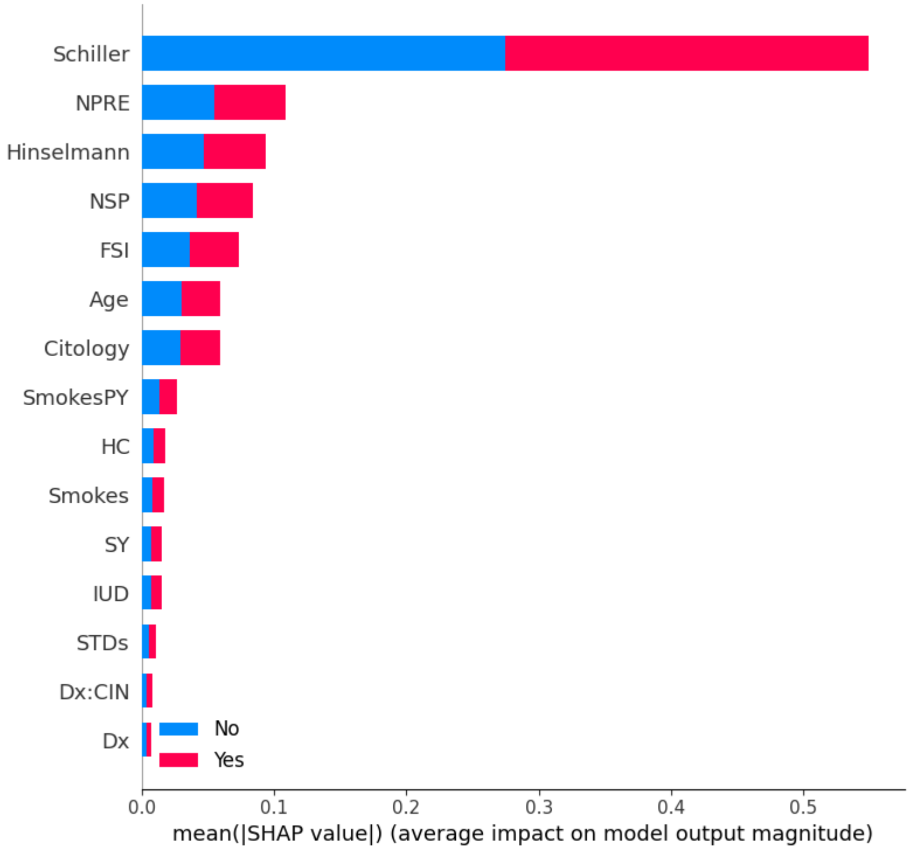

Figure 11. Global explanation of stacking model according to SHAP explainer.

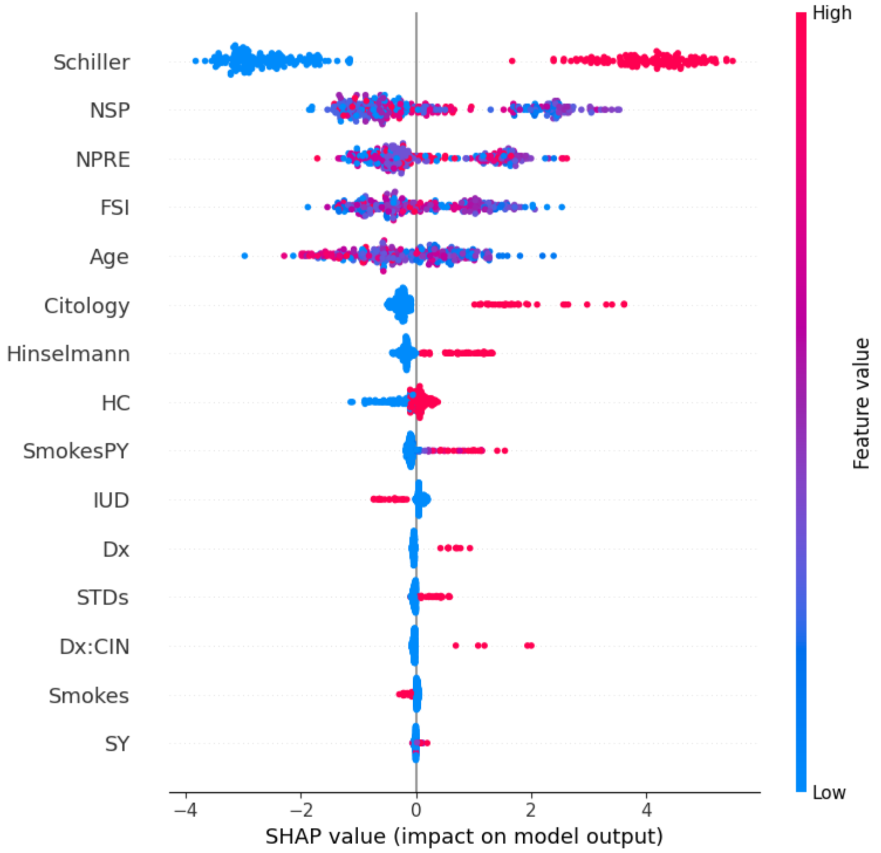

Figure 12. SHAP summary plot according to mean SHAP values.

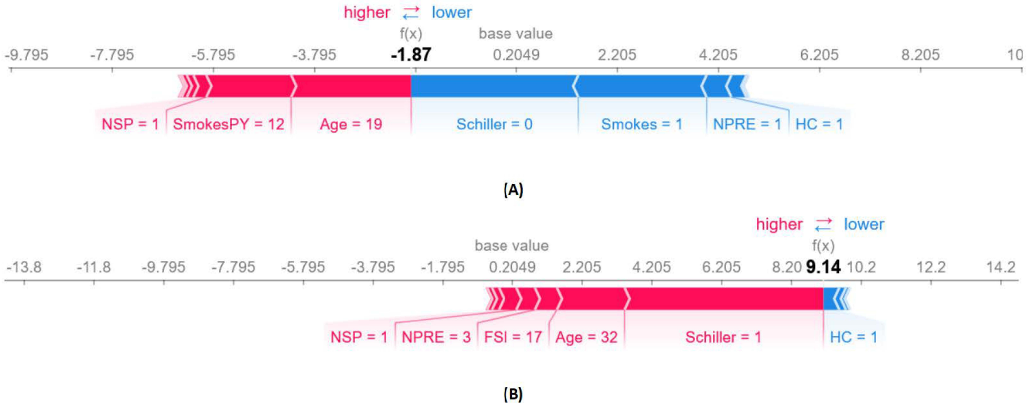

Figure 13. Force plot for specific instance. (**A**) instance with class 0, (**B**) instance with class 1.

5.3. Model Results Comparison with the Literature

Table 5 compares the proposed model with other models. The authors in [2] applied DT with REF and registered an accuracy of 98.72. RF recorded 99 accuracies in [15]. In [18], DT with wrapper feature selection approach recorded 97.5. In [18], the accuracy of RF was registered at 96.4. RF with different feature selection methods recorded an accuracy of 98.33. In [21], the authors used a voting model with PCA that recorded 97.44 accuracy. In [22], the authors used an RF model with PCA that recorded 96.06 accuracy. Our work recorded the highest accuracy with 99.44 for applying stacking with REF; also, our work provided local and global explanations of the developed model to ensure its efficiency, effectiveness, and trustworthiness.

Table 5. Comparing the proposed model with other models.

Papers	Models	Feature Selection Methods	Performance
[2]	DT	REF	Accuracy = 98.7
[15]	RF	—	Accuracy = 99
[17]	DT	wrapper	Accuracy = 97.5
[18]	RF		Accuracy = 96.4
[19]	RF	FST	Accuracy = 98.33
[21]	Voting	PCA	Accuracy = 97.44
[22]	RF	PCA	Accuracy = 96.06
Our work	Stacking	REF	Accuracy = 99.44

6. Conclusions

The main goals of this paper are to study the effect of applying feature selection methods with stacking models for the prediction of cervical cancer, proposing stacking ensemble learning that combines different models with meta-learners to predict cervical cancer. The min steps of prediction cervical cancer include pre-processing step, feature selection methods, handling imbalanced data, optimization models, and training models. A cervical cancer dataset from UCI that is highly imbalanced and contains missing values is used. The pre-processing of data steps consists of removing columns that have 70% missing values, filling missing values by mean for numerical data and mode for categorical data, and dataset encoding. SMOTE-Tomek was used to handle imbalanced data. REF, chi2, and tree-based feature extraction methods were used to extract the most important features that affect cervical cancer prediction. Stacking models are extended to multiple levels: Level 1 (multiple base learners) and Level 2 (meta-learner). In Level 1: the output of multi-base models is combined in stacking (training stacking and testing stacking). In level 2, training stacking is used to train meta-learner (SVM, LR, and RF). Testing stacking is used to evaluate meta-learner (SVM, LR, and RF) using different evaluation metrics. Bayesian optimization optimizes models and selects the best model architecture. Based on the selected features from RFE, the stacking model has higher accuracy, precision, recall, f1-score, and AUC, 99.44, 99.44, 99.44, and 99.44, respectively. To ensure that the model is efficient, effective, and trustworthy, we provide local and global explanations. The main limitation is an unbalanced dataset can lead to biased models that perform poorly on the minority class. The model is trained on a dataset that comprises a majority class and a minority class, which could lead to inaccurate predictions for the minority class due to the model favoring the majority class. Minority classes are challenging to learn since fewer examples exist for training. Therefore, the model may not capture the underlying patterns in minority class data. Overfitting may occur when models are trained on imbalanced data. As a result, new, unseen data may not generalize as well. To address these limitations, we used SMOTE-Tomek to handle unbalanced data, and various evaluation matrixes, such as precision, recall, and F1-score, and stacking models are used to improve predictive performance, model diversity, and reduce overfitting. In future work, the proposed model will be validated using different datasets and diseases to ensure its generalization abilities. In addition to structured data, they will use unstructured data from datasets such as clinical notes and images. We aim to gain valuable information and improve our model's overall performance. To ensure the model's generalizability, we plan to aggregate more data. In addition, we will analyze the model from the perspective of computing complexity. In addition, the developed model will be deployed in a real clinical environment to evaluate its performance.

Author Contributions: Methodology, A.A., H.S., S.M., R.M.A.S. and A.S.T.; Software, A.A. and H.S.; Validation, H.S.; Formal analysis, H.S.; Investigation, A.A., S.M. and A.S.T.; Writing—original draft, A.A., H.S., S.M., R.M.A.S. and A.S.T.; Writing—review & editing, A.A., H.S., S.M., R.M.A.S. and A.S.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research project was supported by a grant from Midocean University.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Data of cervical cancer availability statement: dataset was obtained from the open-access cervical cancer (Risk Factors) dataset database of Cervical Cancer Risk Factors for Biopsy and are available at 10 August 2023 https://archive.ics.uci.edu/ml/datasets/Cervical+cancer+%28Risk+Factors%29.

Acknowledgments: The authors would like to thank the support by a grant from Midocean University.

Conflicts of Interest: All authors declare that they have no conflict of interest.

References

1. World Health Organization. Cervical-Cancer. 2023. Available online: https://www.who.int/news-room/fact-sheets/detail/cervical-cancer (accessed on 5 August 2023).
2. Tanimu, J.J.; Hamada, M.; Hassan, M.; Kakudi, H.; Abiodun, J.O. A machine learning method for classification of cervical cancer. *Electronics* **2022**, *11*, 463. [CrossRef]
3. Venkatesh, B.; Anuradha, J. A review of feature selection and its methods. *Cybern. Inf. Technol.* **2019**, *19*, 3–26. [CrossRef]
4. Gu, Q.; Li, Z.; Han, J. Generalized fisher score for feature selection. *arXiv* **2012**, arXiv:1202.3725.
5. Lin, X.; Li, C.; Zhang, Y.; Su, B.; Fan, M.; Wei, H. Selecting feature subsets based on SVM-RFE and the overlapping ratio with applications in bioinformatics. *Molecules* **2017**, *23*, 52. [CrossRef] [PubMed]
6. He, Y.; Yu, H.; Yu, R.; Song, J.; Lian, H.; He, J.; Yuan, J. A correlation-based feature selection algorithm for operating data of nuclear power plants. *Sci. Technol. Nucl. Install.* **2021**, *2021*, 9994340. [CrossRef]
7. Guyon, I.; Weston, J.; Barnhill, S.; Vapnik, V. Gene selection for cancer classification using support vector machines. *Mach. Learn.* **2002**, *46*, 389–422. [CrossRef]
8. Rokach, L. Ensemble-based classifiers. *Artif. Intell. Rev.* **2010**, *33*, 1–39. [CrossRef]
9. Breiman, L. Bagging predictors. *Mach. Learn.* **1996**, *24*, 123–140. [CrossRef]
10. Schapire, R.E. A brief introduction to boosting. *Ijcai* **1999**, *99*, 1401–1406.
11. Saleh, H.; Mostafa, S.; Alharbi, A.; El-Sappagh, S.; Alkhalifah, T. Heterogeneous ensemble deep learning model for enhanced Arabic sentiment analysis. *Sensors* **2022**, *22*, 3707. [CrossRef]

12. Rajagopal, S.; Kundapur, P.P.; Hareesha, K.S. A stacking ensemble for network intrusion detection using heterogeneous datasets. *Secur. Commun. Netw.* **2020**, *2020*, 4586875. [CrossRef]
13. Rudin, C. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nat. Mach. Intell.* **2019**, *1*, 206–215. [CrossRef] [PubMed]
14. Lee, H.; Yune, S.; Mansouri, M.; Kim, M.; Tajmir, S.H.; Guerrier, C.E.; Ebert, S.A.; Pomerantz, S.R.; Romero, J.M.; Kamalian, S.; et al. An explainable deep-learning algorithm for the detection of acute intracranial haemorrhage from small datasets. *Nat. Biomed. Eng.* **2019**, *3*, 173–182. [CrossRef] [PubMed]
15. Al Mudawi, N.; Alazeb, A. A model for predicting cervical cancer using machine learning algorithms. *Sensors* **2022**, *22*, 4132. [CrossRef] [PubMed]
16. Fatlawi, H.K. Enhanced classification model for cervical cancer dataset based on cost sensitive classifier. *Int. J. Comput. Tech.* **2017**, *4*, 115–120.
17. Choudhury, A.; Wesabi, Y.; Won, D. Classification of cervical cancer dataset. *arXiv* **2018**, arXiv:1812.10383.
18. Razali, N.; Mostafa, S.A.; Mustapha, A.; Abd Wahab, M.H.; Ibrahim, N.A. Risk factors of cervical cancer using classification in data mining. *J. Physics Conf. Ser.* **2020**, *1529*, 022102. [CrossRef]
19. Ali, M.M.; Ahmed, K.; Bui, F.M.; Paul, B.K.; Ibrahim, S.M.; Quinn, J.M.; Moni, M.A. Machine learning-based statistical analysis for early stage detection of cervical cancer. *Comput. Biol. Med.* **2021**, *139*, 104985. [CrossRef]
20. Adem, K.; Kiliçarslan, S.; Cömert, O. Classification and diagnosis of cervical cancer with stacked autoencoder and softmax classification. *Expert Syst. Appl.* **2019**, *115*, 557–564. [CrossRef]
21. Alsmariy, R.; Healy, G.; Abdelhafez, H. Predicting cervical cancer using machine learning methods. *Int. J. Adv. Comput. Sci. Appl.* **2020**, *11*. [CrossRef]
22. Abdoh, S.F.; Rizka, M.A.; Maghraby, F.A. Cervical cancer diagnosis using random forest classifier with SMOTE and feature reduction techniques. *IEEE Access* **2018**, *6*, 59475–59485. [CrossRef]
23. Asadi, F.; Salehnasab, C.; Ajori, L. Supervised algorithms of machine learning for the prediction of cervical cancer. *J. Biomed. Phys. Eng.* **2020**, *10*, 513.
24. Wang, S.; Dai, Y.; Shen, J.; Xuan, J. Research on expansion and classification of imbalanced data based on SMOTE algorithm. *Sci. Rep.* **2021**, *11*, 24039. [CrossRef] [PubMed]
25. Le, T.T.H.; Oktian, Y.E.; Kim, H. XGBoost for imbalanced multiclass classification-based industrial internet of things intrusion detection systems. *Sustainability* **2022**, *14*, 8707. [CrossRef]
26. Yu, S.; Guo, J.; Zhang, R.; Fan, Y.; Wang, Z.; Cheng, X. A re-balancing strategy for class-imbalanced classification based on instance difficulty. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 70–79.
27. Jiang, P.; Suzuki, H.; Obi, T. XAI-based cross-ensemble feature ranking methodology for machine learning models. *Int. J. Inf. Technol.* **2023**, *15*, 1759–1768. [CrossRef]
28. Le, T.T.H.; Kim, H.; Kang, H.; Kim, H. Classification and explanation for intrusion detection system based on ensemble trees and SHAP method. *Sensors* **2022**, *22*, 1154. [CrossRef]
29. Chakir, O.; Rehaimi, A.; Sadqi, Y.; Alaoui, E.A.A.; Krichen, M.; Gaba, G.S.; Gurtov, A. An empirical assessment of ensemble methods and traditional machine learning techniques for web-based attack detection in industry 5.0. *J. King Saud Univ.-Comput. Inf. Sci.* **2023**, *35*, 103–119. [CrossRef]
30. Fernandes, K.C.J.; Fernandes, J. Cervical Cancer (Risk Factors). UCI Machine Learning Repository. 2017. Available online: http://archive.ics.uci.edu/dataset/383/cervical+cancer+risk+factors (accessed on 5 August 2023).
31. Huang, J.; Li, Y.F.; Xie, M. An empirical analysis of data preprocessing for machine learning-based software cost estimation. *Inf. Softw. Technol.* **2015**, *67*, 108–127. [CrossRef]
32. Hartini, E. Classification of missing values handling method during data mining. *Sigma Epsil.-Bul. Ilm. Teknol. Keselam. Reakt. Nukl.* **2018**, *21*.
33. Wu, J.; Chen, X.Y.; Zhang, H.; Xiong, L.D.; Lei, H.; Deng, S.H. Hyperparameter optimization for machine learning models based on Bayesian optimization. *J. Electron. Sci. Technol.* **2019**, *17*, 26–40.
34. Snoek, J.; Larochelle, H.; Adams, R.P. Practical bayesian optimization of machine learning algorithms. *Adv. Neural Inf. Process. Syst.* **2012**, *25*.
35. Brochu, E.; Cora, V.M.; De Freitas, N. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv* **2010**, arXiv:1012.2599.
36. Lemaître, G.; Nogueira, F.; Aridas, C.K. Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. *J. Mach. Learn. Res.* **2017**, *18*, 559–563.
37. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357. [CrossRef]
38. Zeng, M.; Zou, B.; Wei, F.; Liu, X.; Wang, L. Effective prediction of three common diseases by combining SMOTE with Tomek links technique for imbalanced medical data. In Proceedings of the 2016 IEEE International Conference of Online Analysis and Computing Science (ICOACS), Chongqing, China, 28–29 May 2016; pp. 225–228.
39. Khleel, N.A.A.; Nehéz, K. A novel approach for software defect prediction using CNN and GRU based on SMOTE Tomek method. *J. Intell. Inf. Syst.* **2023**, *60*, 673–707. [CrossRef]

40. SMOTETomek. 2023. Available online: https://imbalanced-learn.org/stable/references/generated/imblearn.combine.SMOTETomek.html (accessed on 5 August 2023).
41. McHugh, M.L. The chi-square test of independence. *Biochem. Medica* **2013**, *23*, 143–149. [CrossRef]
42. Germano, M. Turbulence: The filtering approach. *J. Fluid Mech.* **1992**, *238*, 325–336. [CrossRef]
43. Jović, A.; Brkić, K.; Bogunović, N. A review of feature selection methods with applications. In Proceedings of the 2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, Croatia, 25–29 May 2015; pp. 1200–1205.
44. Stańczyk, U. Feature evaluation by filter, wrapper, and embedded approaches. *Feature Sel. Data Pattern Recognit.* **2015**, 29–44. [CrossRef]
45. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
46. LaValley, M.P. Logistic regression. *Circulation* **2008**, *117*, 2395–2399. [CrossRef]
47. Suthaharan, S.; Suthaharan, S. Support vector machine. In *Machine Learning Models and Algorithms for Big Data Classification: Thinking with Examples for Effective Learning*; Springer: Berlin, Germany, 2016; pp. 207–235.
48. Quinlan, J.R. Learning decision tree classifiers. *ACM Comput. Surv. (CSUR)* **1996**, *28*, 71–72. [CrossRef]
49. Rigatti, S.J. Random forest. *J. Insur. Med.* **2017**, *47*, 31–39. [CrossRef] [PubMed]
50. Friedman, J.H. Greedy function approximation: A gradient boosting machine. *Ann. Stat.* **2001**, *29*, 1189–1232. [CrossRef]
51. Holzinger, A.; Biemann, C.; Pattichis, C.S.; Kell, D.B. What do we need to build explainable AI systems for the medical domain? *arXiv* **2017**, arXiv:1712.09923.
52. Lundberg, S.M.; Lee, S.I. A unified approach to interpreting model predictions. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.
53. Albini, E.; Long, J.; Dervovic, D.; Magazzeni, D. Counterfactual shapley additive explanations. In Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency, Seoul, Republic of Korea, 21–24 June 2022; pp. 1054–1070.
54. Narkhede, S. Understanding auc-roc curve. *Towards Data Sci.* **2018**, *26*, 220–227.
55. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
56. SHAP Explainers. 2023. Available online: https://shap.readthedocs.io/en/latest/ (accessed on 5 August 2023).
57. Matplotlib.pyplot. 2023. Available online: https://matplotlib.org/3.5.3/api/_as_gen/matplotlib.pyplot.html (accessed on 5 August 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.