Lecture Notes in Electrical Engineering 786

M. Mitra
Mita Nasipuri
Maitreyi Ray Kanjilal *Editors*

Computational Advancement in Communication, Circuits and Systems

Proceedings of 3rd ICCACCS 2020

Springer

Lecture Notes in Electrical Engineering

Volume 786

The book series *Lecture Notes in Electrical Engineering* (LNEE) publishes the latest developments in Electrical Engineering - quickly, informally and in high quality. While original research reported in proceedings and monographs has traditionally formed the core of LNEE, we also encourage authors to submit books devoted to supporting student education and professional training in the various fields and applications areas of electrical engineering. The series cover classical and emerging topics concerning:

- Communication Engineering, Information Theory and Networks
- Electronics Engineering and Microelectronics
- Signal, Image and Speech Processing
- Wireless and Mobile Communication
- Circuits and Systems
- Energy Systems, Power Electronics and Electrical Machines
- Electro-optical Engineering
- Instrumentation Engineering
- Avionics Engineering
- Control Systems
- Internet-of-Things and Cybersecurity
- Biomedical Devices, MEMS and NEMS

For general information about this book series, comments or suggestions, please contact leontina.dicecco@springer.com.

To submit a proposal or request further information, please contact the Publishing Editor in your country:

China

Jasmine Dou, Editor (jasmine.dou@springer.com)

India, Japan, Rest of Asia

Swati Meherishi, Editorial Director (Swati.Meherishi@springer.com)

Southeast Asia, Australia, New Zealand

Ramesh Nath Premnath, Editor (ramesh.premnath@springernature.com)

USA, Canada:

Michael Luby, Senior Editor (michael.luby@springer.com)

All other Countries:

Leontina Di Cecco, Senior Editor (leontina.dicecco@springer.com)

**** This series is indexed by EI Compendex and Scopus databases. ****

More information about this series at http://www.springer.com/series/7818

M. Mitra · Mita Nasipuri · Maitreyi Ray Kanjilal
Editors

Computational Advancement in Communication, Circuits and Systems

Proceedings of 3rd ICCACCS 2020

Springer

Editors
M. Mitra
Department of Electronics
and Telecommunication
IIEST Shibpur
Howrah, India

Mita Nasipuri
Department of Computer Science
and Engineering
Jadavpur University
Kolkata, India

Maitreyi Ray Kanjilal
Narula Institute of Technology
Kolkata, India

ISSN 1876-1100 ISSN 1876-1119 (electronic)
Lecture Notes in Electrical Engineering
ISBN 978-981-16-4034-6 ISBN 978-981-16-4035-3 (eBook)
https://doi.org/10.1007/978-981-16-4035-3

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

Preface

This book gathers the proceedings of the International Conference on Computational Advancement in Communication Circuits and Systems (ICCACCS 2020), which was organized by Narula Institute of Technology under the patronage of the JIS group, affiliated to Maulana Abul Kalam Azad University of Technology. The book presents peer-reviewed papers that highlight new theoretical and experimental findings in the fields of electronics and communication engineering, advanced computing, pattern recognition and analysis, signal and image processing, etc. The respective papers cover a broad range of principles, techniques and applications in computing and communication, environment-friendly computing, reconfigurable computing, low-power nano technology and VLSI design. The proceedings reflect the conference's strong emphasis on methodological approaches and focus on applications within the domain of computational advancement in communication circuits and systems. It also addresses the emerging technologies in electronics and communication together with the latest practices, issues and trends.

Howrah, India	M. Mitra
Kolkata, India	Mita Nasipuri
Kolkata, India	Maitreyi Ray Kanjilal

Contents

About the Editors

Dr. M. Mitra did his B.Tech., M.Tech. & Ph.D. from the Department of Radiophysics and Electronics under Calcutta University in the year 1982, 1985–1986, and 1995, respectively. He joined as Lecturer in the Department of Electronics and Telecommunication Engineering of IIEST Shibpur (formerly B.E. College) in the year 1995 and became Professor in the same department in the year 2011. He served the department as Head from 2012–2014 and once again from 2016–2018. He had also served the institute at different capacity such as Head of the Department of Purabi Das School of Information Technology (PDSIT) of IIEST, Shibpur. His main area of research is on semiconductor microwave devices, especially IMPATT. He had fabricated and characterized IMPATT diode at Xband for the first time in India. He had published four books such as *Satellite Communication* (2nd Ed.), *Microwave Engineering* (3rd Ed.), *Microwave Semiconductor Devices*, and *Electronic Circuits* from the house of Prentice Hall. So far he had published about 80 research papers in different SCI indexed journals. Till now 15 students have got their Ph.D. degree under his supervision and 5 are continuing. He is Member of different prestigious societies like IEEE, Life Member of IETE, etc.

Prof. Mita Nasipuri received her B.E. and M.E. degrees in Electronics and Telecommunication Engineering from Jadavpur University, Kolkata, India, in 1979 and 1981, respectively, and did her Ph.D. degree in Engineering from Jadavpur University, in 1990. She is currently Professor in the Department of Computer Science and Engineering, Jadavpur University. From July 2004 to July 2006, she was Head of the Computer Science and Engineering Department, Jadavpur University. She has also served as Member of various UG/PG syllabus committees and the Doctoral committee of Engineering faculty of Jadavpur University. Since 2008, she is serving as Coordinator of Centre for Microprocessor Applications for Training Education and Research of Jadavpur University. Her research interest includes biomedical signal processing, image processing, pattern recognition, bioinformatics, medical image analysis, and multi-media system. She has authored or co-authored more than 500 research articles including several book chapters, out of which, more than 100 papers have been published in science citation indexed (SCI/SCIE) technical journals. Two

US patents have been granted on her work. She has supervised more than 22 Ph.D. students. She has been granted sponsored projects by the Government of India with a total amount of around INR 1.5 Crore. She is Fellow of Institution of Engineers (India) and West Bengal Academy of Science and Technology, India. She is also Senior Member of the IEEE, USA, since 1992.

Prof. Maitreyi Ray Kanjilal is currently holding the post of Principal of Narula Institute of Technology and also Professor of Department of Electronics and Communication Engineering. She did her UG and PG from Calcutta University. She has been awarded Ph.D. from Calcutta University in 2000. She has more than 22 years of teaching experience. She has more than 100 published papers in national and international journals and conferences. She has 6 Ph.D. scholars and has published two patents and one is filed. She is the author of two books on Basic Electronics and one book on Analog Electronics. Her subject interest is on high-power and high-frequency (MM-wave—THz region) operation of wide-bandgap semiconductor devices, Group IV-IV, III-V material-based devices, superlattice-based devices, applications of these devices in the medical arena, nanoscale devices: design and develop, heterostructure and heterojunction semiconductor devices, microelectronics fabrication, low power devices, VLSI circuits, nanodevices, spintronics, and quantum computing.

Proper Choice of a Machine Learning Algorithm for Breast Cancer Prediction

Arijit Das, Tanisha Khan, Subhram Das, and D. K. Bhattacharya

Abstract Breast cancer is the most common form of invasive cancer and after lung cancer, it is the second leading cause of cancer death in women. Many statistical models have been used to predict the malignancy of the tumor. Therefore due to the violation of the proportional hazard assumption, a statistical model may fail to predict breast cancer accurately. In the current epoch, machine learning algorithms play a decisive role in predicting the malignancy of a tumor with high accuracy. The primary purpose of this paper is to compare the performance of eleven different machine learning classification techniques for breast cancer prediction. Wisconsin Diagnostic Breast Cancer dataset is utilized to compare these established algorithms based on the k-fold Cross-Validation accuracy score. Additionally, three different feature selection methods have been incorporated to reduce the number of features on the dataset. After the reduction of the features, the same methods are applied again to compare performance based on their accuracy score. It is found that all the algorithms perform very well with more than 93% accuracy score; among these Logistic Regression, Support Vector Classification and Multilayer Perceptron get an accuracy score of over 98%. It is also observed that even after a drastic reduction in the number of features, the result remains satisfactory, and the accuracy score is more than 90% for all the applied algorithms.

Keywords Breast cancer · Machine learning · Feature selection · k-fold Cross-Validation

1 Introduction

Amidst developed nations, breast cancer happens to be the second most common cancer after lung cancer, from which most of the women across the world suffer

A. Das · T. Khan · S. Das (✉)
Computer Science and Engineering, Narula Institute of Technology, Kolkata, India

D. K. Bhattacharya
Pure Mathematics, University of Calcutta, Kolkata, India

M. Mitra et al. (eds.), *Computational Advancement in Communication, Circuits and Systems*, Lecture Notes in Electrical Engineering 786,
https://doi.org/10.1007/978-981-16-4035-3_1

most [1–3]. The percentage of occurrence and transience rate of breast cancer is very high [4–6]. This creates one of the significant and most pressing health problems in today's society.

With the help of mammography tools, breast cancer can be detected in the early stage even before any physical symptoms were developed. However, like any other screening tool mammography has got its loopholes, as it can both underdiagnose and overdiagnoses, subsequently leading to multiple diagnostic fallacies. To resolve this flaw, machine learning techniques and data mining are being used for the last few years to predict cancer diagnosis. Pathological and clinical data [7] are to be used for the computational purpose to assist conventional tools to predict this breast cancer. The more is the accurate data produced, the more is the accurate level of prediction under proper authenticated techniques. The use of computers has significantly improved technology along with the simultaneous advancement of medical database management systems leading to the creation of a huge volume of heterogeneous data and medical databases. Therefore, the intelligent healthcare system combined with various Machine Learning (ML) techniques is a crucial domain for cancer prediction. These current techniques also help doctors to diagnose cancer patients with better perfection [8]. Previously, basic software programs such as Microsoft Excel, SPSS, and STATA [9–11] were used by clinicians to analyze the factors influencing breast cancer survival rate. To demonstrate the advantages and potential of ML, a lot of work is done where different types of conventional statistical methods are compared with traditional ML classification methods. These traditional and statistical methods had various drawbacks, for which they subsequently failed to generate rational and creative visualizations. ML algorithms can be broadly classified into two categories—supervised learning and unsupervised learning. Whether to choose the first category or the second one depends on the types of data and their structures. A current review [12] shows that most of the ML algorithms, which are used in breast cancer diagnosis and prognosis, are of a supervised type. To establish that ML methods could be a reliable classification method for prediction of breast cancer, the classification methods DT and ANN are compared in 2004 with the statistical linear regression method based on a large dataset comprising of more than 200,000 cases. It is found that DT has an accuracy rate of 93.6%, ANN has an accuracy rate of 91.2% and linear aggression has an accuracy rate of 89.2% [13]. Thus both DT and ANN are better than linear regression. Moreover, DT is the best predictor and ANN is the next best. K-fold cross-validation is applied to the model in [14] based on Square Support Vector Machines (LS-SVM) concerning accuracy, sensitivity, and specify matrix. To solve classification problems under WBCD of [15], a new genetically optimized ANN (GOANN) algorithm is proposed in 2015 [16], which uses genetic programming (GP) based on ANN. In 2014, a classification accuracy of 94.56% is obtained on classifying WBCD [17]. This is achieved by applying the J48 decision tree method, which is originally developed by the WEKA project team [18]. Feature selection is also an important element in the field of machine learning as it reduces the number of features in the data, consequently decreasing the training time, noise, and complexity of the model giving us an upper hand of the set of all features. Thus the real challenge is to apply the feature selection method to find the minimum number of features, which can predict breast cancer as efficiently as possible on the WDBC data set using the K-fold method of accuracy. This is the motivation of the present paper. We validate our method on the Wisconsin Diagnostic Breast Cancer dataset. It is found that a combination of the K-fold cross-validation accuracy method and feature selection methods, breast cancer is predicted more efficiently by different ML algorithms. We also compared these different ML algorithms based on their accuracy score and choose the right alternatives. It is also proved that after a drastic reduction of the features results remains satisfactory.

2 Material and Methods

2.1 Data Sets

Wisconsin Diagnostic Breast Cancer (WDBC) data set [19] has been used in our current work. This data set contains information on 569 breast tumors, out of these 357 are benign classes and 212 are malignant classes. This data set consists of 10 different features (Radius, Texture, Perimeter, Area, Smoothness, Compactness, Concavity, Concave points, Symmetry, Fractal Dimension). All the features are computed from a digitized image of an acceptable needle aspirate (FNA) of a breast mass. All these 10 features are present with their 3 different measurements (mean, standard error, and worst measurement). Therefore, the data set comprised of 30 features in total.

2.2 Experiments

2.2.1 ML Algorithms

After pre-processing the dataset, the following eleven different machine learning algorithms have been applied: (a) Logistic Regression, (b) Support Vector Machine (Linear and Radial basis function kernel), (c) Multilayer Perceptron, (d) Adaboost, (e) Gradient Boosting, (f) K-Nearest Neighbors, (g) XG Boost, (h) Random Forest, (i) Linear Discriminant Analysis, (j) Decision Tree, (k) Naïve Bayes.

All the above-mentioned algorithms have been applied to the data set to calculate accuracy score, recall score, precision score, and f1 score. As of now, ample of articles on breast cancer prediction [20] has used test accuracy score for comparison of different machine learning algorithms whereas in this paper, instead of using test accuracy score for comparison, 10-fold cross-validation accuracy score has been used. In 10-fold cross-validation, data is divided into 10 number of folds of the same size. Now 9 folds of the data are used for training while the remaining onefold is used for testing and this process is then repeated 10 times. Thus, here the data is being used much more efficiently as every observation has been used for both training

Table 1 Measurement of performance using the different ML classification algorithms

	Accuracy	Recall	Precision	f1_score
Logistic Regression	0.982485	0.96688	0.986107	0.976053
SVC (RBF kernel)	0.982455	0.95735	0.995455	0.975339
MLP	0.982424	0.96212	0.990238	0.975717
Linear SVC	0.978975	0.95757	0.985627	0.970937
Adaboost	0.977189	0.96233	0.976605	0.969076
Gradient boosting	0.968509	0.94350	0.971605	0.956739
KNN	0.966694	0.91969	0.990097	0.952358
XG boost	0.963183	0.92943	0.971623	0.949542
Random forest	0.959734	0.93896	0.953338	0.945226
LDA	0.956227	0.89177	0.990238	0.93681
Decision tree	0.93169	0.91103	0.905976	0.906979
Naive bayes	0.931568	0.90064	0.919914	0.907504

and testing. After evaluating the accuracy score, we have manually selected all the hyperparameters for the algorithms. Eventually, all the algorithms have been sorted according to the accuracy score as shown in Table 1.

3 Feature Selection Methods

Feature Selection is the process to select the features which contribute most to predict the output variable. Having irrelevant features in the data set can negatively impact the performance of an ML algorithm. Thus, feature selection methods are used to remove unnecessary irrelevant features from the data set. Some recent studies [21] also show that a poor machine learning model with key important features works much efficiently than a low-error ML model that lack such important features. The following feature selection methods are incorporated into our study:

- Correlation-based feature selection

Correlation is a well-known method to measure the similarity between two features. For two linearly dependent features, their correlation is ± 1. When two features are uncorrelated, the correlated coefficient is 0. In this method, a correlation matrix is formed with the help of the Pearson correlation by which the features are filtered. Given a pair of features x, y, we can define Pearson correlation as:

$$\rho_{x,y} = \frac{\mathrm{cov}(X,Y)}{\rho_X \rho_y} \tag{1}$$

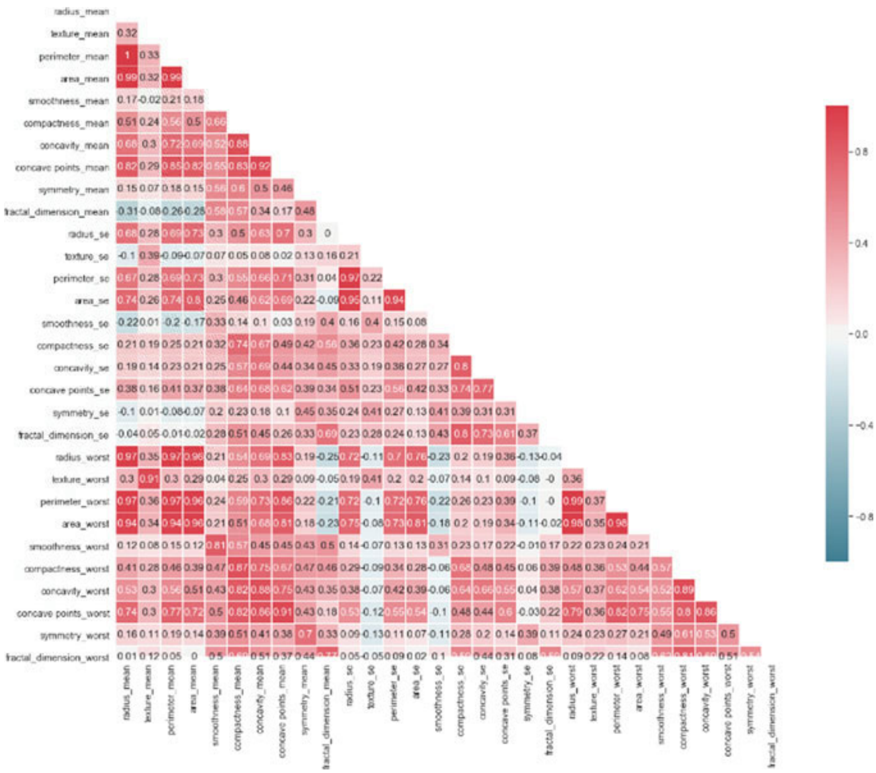


Fig. 1 Pearson correlation heatmap

where cov represents the covariance, σX represents the standard deviation of X, and σY represents the standard deviation of Y.

From this Pearson correlation heatmap in Fig. 1, it is observed that the following features are highly correlated:

(1). Radius_mean, perimeter_mean and area_mean, (2). compactness_mean, concavity_mean and concave points_mean, (3). radius_se, perimeter_se and area_se, (4). radius_worst, perimeter_worst and area_worst, (5). compactness_worst, concavity_worst and concave points_worst, (6). compactness_se, concavity_se, and concave points_se, (7). texture_mean and texture_worst, 8. area_worst and area_mean.

One feature is selected from a set of correlated features and remaining redundant features are removed from the data set. Therefore, in our experiment the following features are removed from the dataset: perimeter_mean, radius_mean, compactness_mean, concave points_mean, radius_se, perimeter_se, radius_worst, perimeter_worst, compactness_worst, concave points_worst, compactness_se, concave points_se, texture_worst, area_worst.

Table 2 Measurement of performance after applying Correlation-based feature selection

	Accuracy	Recall	Precision	f1_score
SVC(RBF Kernel)	0.975374	0.952597	0.982016	0.966019
MLP	0.973619	0.947835	0.982016	0.963458
Logistic regression	0.971865	0.947835	0.977016	0.96114
Adaboost	0.970201	0.948485	0.971536	0.958609
Linear SVC	0.966662	0.929221	0.981818	0.953567
Random forest	0.964907	0.938961	0.967452	0.95187
XG boost	0.964907	0.938961	0.967148	0.951273
Gradient boosting	0.959643	0.938961	0.954573	0.944716
KNN	0.954411	0.900649	0.976517	0.935511
LDA	0.947455	0.86342	0.995238	0.922951
Decision tree	0.941972	0.938745	0.913302	0.923911
Naive bayes	0.928091	0.886797	0.919804	0.901187

After removing those 13 features, machine learning algorithms have been applied over the remaining 17 features and the accuracy, recall, precision, and f1 score are shown in Table 2.

- Univariate feature selection

In this method features with the strongest relationship with the target-variable are selected. In our experiment chi squared (chi^2) statistical test has been used to select best n number of features, where n is a positive integer and then machine learning algorithms are applied on those selected feature sets. At first, we have chosen n = 5 to get the 5 most important features. The selected features are area worst, area mean, area se, perimeter worst, perimeter mean. Using these 5 features, again all the scores have been calculated and shown in Table 3.

Next, we have chosen n = 20, and selected features are area worst, area mean, area se, perimeter worst, perimeter mean, radius worst, radius mean, perimeter se, texture worst, texture mean, concavity worst, radius se, concavity mean, compactness worst, concave points worst, concave points mean, compactness mean, symmetry worst, concavity se, compactness se. Similarly calculated Scores using these 20 features have been shown in Table 4.

- Tree-based feature selection

This feature selection method can be applied to tree-based methods as in these methods computing importance of individual features is possible. The importance of a feature can be computed by calculating how much that feature is decreasing the impurity. Features that can decrease the impurity more are more important. For classification problems, the impurity can be either Gini impurity or the information entropy. The Gini impurity and the information entropy can be defined as:

Table 3 Measurement of performance with 5 best features obtained from univariate feature selection

	Accuracy	Recall	Precision	f1_score
XG Boost	0.938679	0.901948	0.934503	0.915624
Random forest	0.937017	0.89697	0.932019	0.913451
Gradient boosting	0.933506	0.901948	0.920883	0.909626
Logistic regression	0.928181	0.869264	0.937902	0.898672
KNN	0.92815	0.85	0.954636	0.897001
Linear SVC	0.926551	0.836147	0.962058	0.892617
Adaboost	0.926427	0.873377	0.926985	0.897516
MLP	0.926426	0.878571	0.923856	0.897936
SVC (RBF kernel)	0.924674	0.854762	0.941671	0.892958
LDA	0.921041	0.830952	0.953482	0.88608
Naive bayes	0.912421	0.822078	0.939639	0.8723
Decision tree	0.910387	0.882468	0.881863	0.88035

Table 4 Measurement of performance with 20 best features obtained from univariate feature selection

	Accuracy	Recall	Precision	f1_score
Adaboost	0.971986	0.957792	0.967619	0.962203
SVC (RBF kernel)	0.971958	0.952814	0.971755	0.961515
Logistic regression	0.971927	0.957576	0.96721	0.961628
Linear SVC	0.971895	0.948268	0.976339	0.96118
MLP	0.970202	0.95303	0.966972	0.959081
XG boost	0.968601	0.948268	0.967708	0.957166
Random FOREST	0.966784	0.938961	0.971345	0.954209
Gradient boosting	0.963276	0.943506	0.958423	0.950198
KNN	0.963184	0.929437	0.971006	0.948468
LDA	0.958014	0.891558	0.995455	0.938591
Decision tree	0.938741	0.934416	0.903869	0.918418
Naive bayes	0.935078	0.895887	0.929886	0.910582

$$\text{Gini} = 1 - \sum_{i=1}^{n} p^2(c_i) \tag{2}$$

$$\text{Entropy} = \sum_{i=1}^{n} -p(c_i)\log_2(p(c_i)) \tag{3}$$

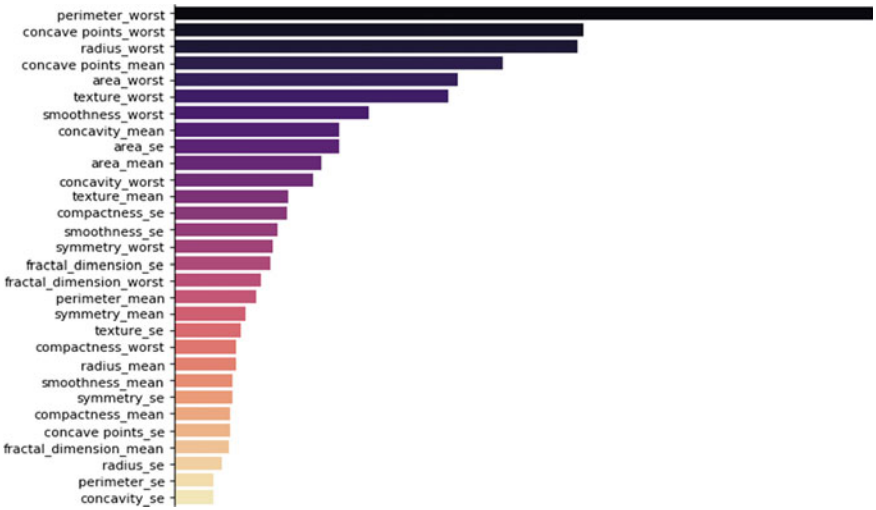


Fig. 2 List of important features based on the tree-based feature selection method

where $p(ci)$ is the probability of choosing a data point with class i. After calculating the impurity decrease from each feature, the average impurity decrease for the whole tree is calculated and the final importance of the feature is calculated. In this way, the importance of each feature has been calculated for Decision Tree, Random Forest, Gradient Boosting, AdaBoost, XGB algorithm. For the comparison purpose, all these important values have been scaled in the range of 0 and 1 using the following formula:

$$z = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{4}$$

Finally, the sum of the value of features importance has been calculated for each model and has been plotted in Fig. 2.

Now, the top 5 features from Fig. 2 has been selected, and after applying all the algorithms on those features, calculated scores are shown in Table 5.

Similarly, the top 20 features from Fig. 2 have been selected and the calculated scores are shown in Table 6.

4 Results and Discussion

From Table 1 it is observed that when all the 30 features are taken, Logistic Regression, Support Vector Classification and Multilayer Perceptron got an accuracy score over 98% and all the other algorithms have performed well with more than 93% accuracy score. After applying correlation-based feature selection by considering 17 features, the accuracy score of XG boost, Random Forest, and Decision tree algorithm

Table 5 Measurement of performance with top 5 features obtained from tree-based feature selection

	Accuracy	Recall	Precision	f1_score
SVC (RBF kernel)	0.949175	0.905628	0.96087	0.930479
Gradient boosting	0.947545	0.92013	0.940549	0.929096
Logistic regression	0.947452	0.924675	0.938608	0.929537
Linear SVC	0.94742	0.900866	0.96087	0.927645
Naive bayes	0.943973	0.92013	0.93276	0.924973
XG boost	0.943942	0.915368	0.935812	0.92434
MLP	0.943912	0.91039	0.94337	0.923914
LDA	0.942219	0.872944	0.971292	0.917983
KNN	0.940464	0.89632	0.943743	0.918403
Decision tree	0.936954	0.911039	0.921575	0.914616
Random forest	0.936894	0.905628	0.927818	0.914293
Adaboost	0.931567	0.89632	0.921741	0.906558

Table 6 Measurement of performance with top 20 features obtained from tree-based feature selection

	Accuracy	Recall	Precision	f1_score
Linear SVC	0.978945	0.957576	0.985213	0.970836
Adaboost	0.97725	0.962554	0.976583	0.968966
Logistic regression	0.977222	0.957576	0.980668	0.968511
SVC (RBF kernel)	0.977191	0.957576	0.9811	0.968383
MLP	0.977191	0.957576	0.980451	0.968513
Random forest	0.970232	0.95303	0.967276	0.959542
XG Boost	0.966693	0.948052	0.963342	0.95473
KNN	0.966601	0.919481	0.990652	0.952724
LDA	0.963276	0.906061	0.995238	0.947488
Gradient boosting	0.96146	0.938745	0.957514	0.947209
Naive bayes	0.943943	0.919697	0.932772	0.924323
Decision tree	0.940277	0.929437	0.914695	0.920899

increased a bit, whereas the accuracy of all the other algorithms decreased by around 1%. Then univariate feature selection has been used to select the top 5 and top 20 important features. Now with 20 features, the accuracy score for XG boost, Random Forest, Decision tree, LDA, and Naive Bayes has increased; and the accuracy of the remaining algorithms has decreased by around 1%. And with the top 5 features, the accuracy score of all the algorithms are in the range from 91 to 93%. Similarly, the top 20 and top 5 features have been selected with tree-based feature selection. However, after obtaining the performance with 20 features, the accuracy of Adaboost, Random Forest, XG Boost, LDA, Naive Bayes, and Decision Tree has subsequently increased whereas the accuracy score of Linear SVC and KNN is almost same, and the accuracy score of Logistic Regression, SVC (RBF kernel), MLP, Gradient Boosting has decreased. With the top 5 features, the accuracy score of all algorithm ranges between 93 and 94%. In Fig. 3, we have compared the results which were obtained without feature selection with those three methods obtained after applying feature selection. In Fig. 4, we have compared univariate and tree-based feature selection method with 5 features. So, by comparing univariate feature selection using tree-based feature selection using the same number of features, tree-based feature selection has given

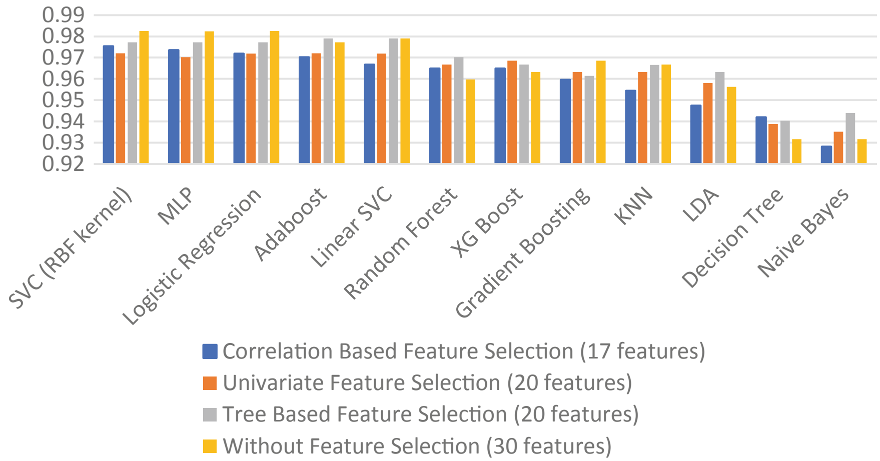


Fig. 3 Comparison of different feature selection methods

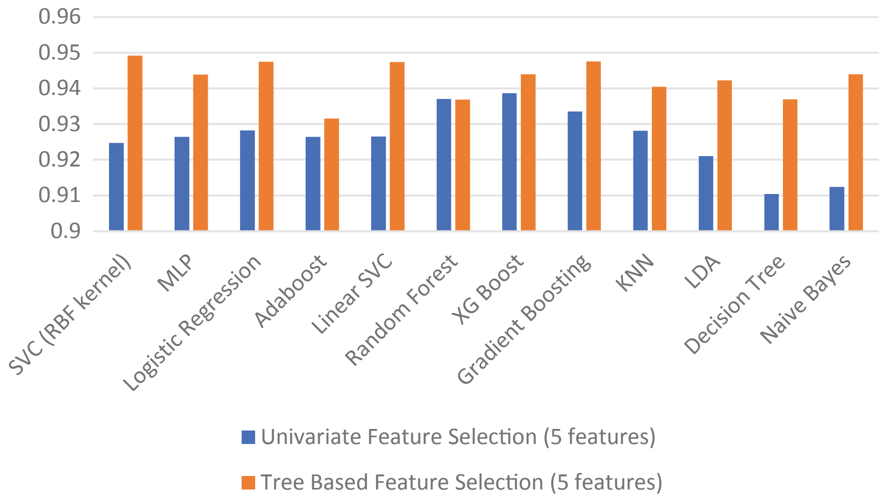


Fig. 4 Comparison between univariate and tree-based feature selection with 5 features

comparatively better results. Another observation is that the Tree-based feature selection has given better results than correlation-based feature selection though only 17 features are used in the case of correlation-based feature selection method. Finally, if we compare all the algorithms in terms of accuracy score, Logistic Regression, Support Vector Classification and Multilayer Perceptron are giving better results in most of the cases.

5 Conclusion

In the healthcare industry, early-stage breast cancer detection is an important issue. Detecting it at a preliminary stage is extremely crucial to prevent the huge death toll. In this paper, eleven machine learning algorithms and three feature selection methods are compared for breast cancer prediction to find the best algorithm and feature selection method in terms of accuracy. Here, we have used k-fold cross-validation for the comparisons which have facilitated efficient data utilization and guarantees a much accurate measurement of the model's performance than the traditional test accuracy. In most cases, it is observed that Logistic Regression, Support Vector Classification, and Multilayer Perceptron outweigh in terms of accuracy than other machine learning algorithms. It is also clearly visible that, tree-based feature selection is giving better results in terms of accuracy score when compared to other feature selection methods.

References

1. Ponnuraja CC, Lakshmanan B, Srinivasan V, Prasanth BK (2017) Decision tree classification and model evaluation for breast cancer survivability: a data mining approach. Biomed Pharmacol J 10:281–289
2. Malehi AS (2014) Diagnostic classification scheme in Iranian breast cancer patients using a decision tree. Asian Pac J Cancer Prev 15:5593–5596
3. Shrivastava SS, Sant A, Aharwal RP (2013) An overview on data mining approach on breast Cancer data. Int J Adv Comput Res. 3(4):256–262
4. Chen W, Zheng R, Baade PD, Zhang S, Zeng H, Bray F, Jemal A, Yu XQ, He J (2016) Cancer statistics in China, 2015. CA Cancer J Clin 66:115–132
5. Siegel RL, Miller KD, Jemal A (2016) Cancer statistics, 2016. CA: Cancer J Clin 66:7–30. https://doi.org/10.3322/caac.21332
6. Torre LA, Bray F, Siegel RL, Ferlay J, Lortet-Tieulent J, Jemal A (2015) Global cancer statistics, 2012. CA Cancer J Clin 65:87–108
7. Pavlopoulos SA, Delopoulos AN (1999) Designing and implementing the transition to a fully digital hospital. IEEE Trans Inf Technol Biomed 3:6–19
8. Yue W, Wang Z, Chen H, Payne A, Liu X (2018) Machine learning with applications in breast cancer diagnosis and prognosis. Designs 2(2):13
9. Bhoo-Pathy N, Verkooijen HM, Tan E-Y, Miao H, Taib NAM, Brand JS et al (2015) Trends in presentation, management and survival of patients with de novo metastatic breast cancer in a southeast Asian setting. Sci Rep 5:16252

10. Yip CH, BhooPathy N, Uiterwaal CS, Taib NA, Tan GH, Mun KS et al (2011) Factors affecting estrogen receptor status in a multiracial Asian country: an analysis of 3557 cases. Breast 20:S60–S64
11. Ng CH, Pathy NB, Taib NA, Ho GF, Mun KS, Rhodes A et al (2014) Do clinical features and survival of single hormone receptor positive breast cancers differ from double hormone receptor positive breast cancers? Asian Pac J Cancer Prev 15:7959–7964
12. Cruz JA, Wishart DS (2006) Applications of machine learning in cancer prediction and prognosis. Cancer Inform. 2:59–77
13. Delen D, Walker G, Kadam A (2005) Predicting breast cancer survivability: a comparison of three data mining methods. Artif Intell Med 34:113–127
14. Mangasarian OL, Setiono R, Wolberg WH (1990) Pattern recognition via linear programming: theory and application to medical diagnosis. In: Coleman TF, Li Y (eds) Large-scale numerical optimization, SIAM Publications, Philadelphia, pp 22–30
15. Bhardwaj A, Tiwari A (2015) Breast cancer diagnosis using genetically optimized neural network model. Expert Syst Appl 42:4611–4620
16. Koza JR, Rice JP (1991) Genetic generation of both the weights and architecture for a neural network. In: IJCNN-91-Seattle international joint conference on neural networks vol 2, pp 397–404. https://doi.org/10.1109/IJCNN.1991.155366
17. Witten IH, Frank E, Trigg L, Hall M, Holmes G. Cunningham SJ (1999) Weka: practical machine learning tools and techniques with Java implementations. (Working paper 99/11). Hamilton, New Zealand: University of Waikato, Department of Computer Science
18. Sumbaly R, Vishnusri N, Jeyalatha S (2014) Diagnosis of breast cancer using decision tree data mining technique. Int J Comput Appl 98:16–24
19. Frank A, Asuncion A (2010) UCI machine learning repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science
20. Agarap AFM (2018) On breast cancer detection: an application of machine learning algorithms on the wisconsin diagnostic dataset. In: Proceedings of the 2nd international conference on machine learning and soft computing (ICMLSC '18). Association for computing machinery, New York, NY, USA, pp 5–9. https://doi.org/10.1145/3184066.3184080
21. Balachandran PV, Xue D, Theiler J, Hogden J, Gubernatis JE, Lookman T (2018) Importance of feature selection in machine learning and adaptive design for materials. In: Lookman T, Eidenbenz S, Alexander F, Barnes C (eds) Materials discovery and design. Springer Series in Materials Science, vol 280. Springer, Cham. https://doi.org/10.1007/978-3-319-99465-9_3

Word Boundary Detection Using Convolutional Neural Network (CNN) and Decision Tree Method

Kaushik Sarkar, Arnab Sadhukhan, Atreyee Mukherjee, Shramana Guchait, and Sudipta Banerjee

Abstract Speech recognition is a vast area for research. In speech recognition word boundary detection is an important part. Understanding and fixing the problems of efficiently detecting where the words are present in a signal is still challenging. In this paper, we have discussed our work on word boundary detection using two different approaches: (1) Convolutional Neural Network and (2) Decision Tree Method. CNN is efficiently used in areas like face recognition, object detection, image classification, etc. whereas a decision tree is broadly used in decision analysis. This study can further help with speech recognition.

Keywords Signal processing · Word boundary detection · Convolution neural network · Decision tree

1 Introduction

Speech signal consists of the words spoken along with some silent parts comprising random noises. During continuous speaking, the consecutive words have very small pauses between them. Thus, it becomes difficult to separate them within a signal. In this paper, the first approach to detect the word boundary is Convolutional Neural Network (CNN).

CNN model is a type of neural network which is used for visual imagery analysis. It recognizes the pattern and based on them it classifies the images. It finds the relation between two nearest pixels and tries to recognize any pattern or higher dimensional features which is present in an image.

In CNN approach, first, we have created word profiles of the audio signals. Then we have done some preprocessing to reduce the input and output shape. Finally, Mel spectrogram transformation is done to create the training dataset. The next

A. Sadhukhan (✉) · A. Mukherjee · S. Guchait · S. Banerjee
Narula Institute of Technology, Kolkata, India

K. Sarkar
Springer Heidelberg, Tiergartenstr. 17, 69121 Heidelberg, Germany

M. Mitra et al. (eds.), *Computational Advancement in Communication, Circuits and Systems*, Lecture Notes in Electrical Engineering 786,
https://doi.org/10.1007/978-981-16-4035-3_2

approach Decision tree method is a type of algorithm that uses a tree-like structure for classification purpose. It creates the branches from the input data and in every branch, it applies some operations based on a comparison among some quantities.

In this method, some transformations on the raw audio signal are done to get the features from the audio signals. Here we have used MFCC matrix as the input to the model. Word profiles are used as the labels for the model.

2 Methodology

For this total work, first, we have to trace that raw audio signal's portion where the words are present. We can get the word boundaries by following two techniques:

- Convolutional Neural Network (CNN)
- Decision Tree Method

We will further discuss these techniques. Before that, we will explain few signal processing techniques which we used in this project.

2.1 Mel Spectrogram

Spectrogram is a powerful visualization tool to represent the signal strength of a signal. It is basically a two-dimensional graph where horizontal axis is the time axis, vertical axis is frequency axis, and amplitude is represented by color intensity where the dark color implies lower amplitude and the bright color implies higher amplitude.

In case of a speech signal, we transform the frequency axis into log scale and color (amplitude) axis into Decibels.

MFCC [3] The human auditory system interpretation of pitch is not in a linear manner. To represent this in a linear scale, the Mel Scale was developed by Stevens, Volkmann, and Newman in 1937 by experimenting with human ears interpretations of pitch [2]. This scale is constructed by assigning a perceptual pitch of 1000 Mels to a 1000 Hz frequency, 40 dB above the listener's threshold and above 1000 Hz the scale shows a logarithmic nature. The formula for Mel scale is:

$$Mel(f) = 2595\log(1 + \frac{f}{700})$$

Mel scale spectrogram is a spectrogram where the y axis is the Mel scale.

2.2 Mel Frequency Cepstral Coefficients (MFCC)

MFCC [3] Speech is the sound that is generated by a human by changing the shape of the vocal tract which includes tongue, teeth, etc. The shape of the vocal tract is responsible for the variation of sound [1]. If we can determine the shape accurately, then we can predict which phoneme is being produced. The shape of the vocal tract manifests itself in the envelope of the short time power spectrum. Here MFCC comes to accurately get the envelope from the audio signal.

Mel Frequency Cepstral Coefficients (MFCCs) is a mostly used feature in automatic speech and speaker recognition.

3 Word Boundary Detection Using Convolutional Neural Network

We are supposed to create a model which can take an audio signal as input and can tell what are each word's starting and ending position in that signal.

Let us consider the following audio signal.

Just by looking at this raw audio signal, we can't get any clue about where the word breaks will be and another problem is different audio signals have different lengths. The required model is supposed to work for audio signals of any length. The following audio signal shown above has five words.

3.1 Create Word Profile

We have used 150 audio signals as our dataset. We have manually taken each word's start time and its duration and saved it as a csv file. Then we have created a profile of words from the csv files.

The csv file looks like this:

Name	Start	Duration
marker 01	0:00.202	0:00.491
marker 02	0:00.718	0:00.517
marker 03	0:01.537	0:00.532

```
Steps:
length = length (audio signal)
word profile = zeros (shape = (length))
for each marker in (no. of markers):
  end time = start time + duration
```

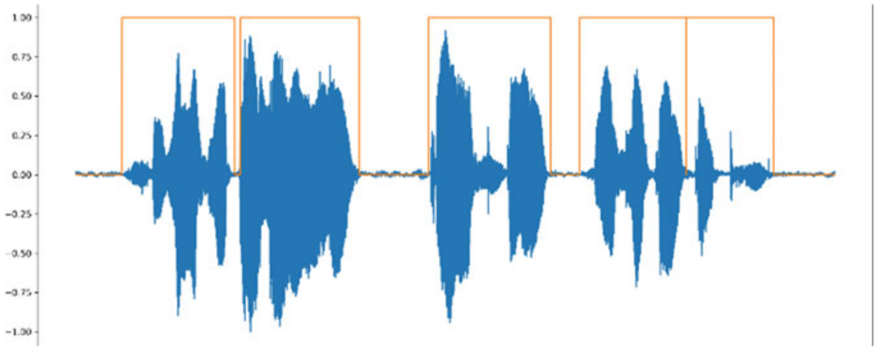


Fig. 1 Raw audio waveform with word boundary

```
  word profile [start time: end time] = 1
end for
```

In this way, we have created a word profile for the signals which looks like the orange profile shown in Fig. 1.

Note: The word profile has the same length as the audio signal.

Now the word profile holds all the information:

- where the words are
- how many words are there
- every word's starting and ending time or position

Now the main idea is, we're giving audio signal to a model as input and it will give the signal's word profile as output.

But this is difficult because suppose a normal audio length is 3 s and if its sampling frequency is 22,050/s, then we have a length of audio signal of $3 \times 22{,}050 = 66{,}150$ samples and the length of word profile is also the same, i.e., 66,150 samples and feeding raw audio signal to train a model is not a good idea. So, we have done some preprocessing on the audio signals.

3.2 Data Preprocessing

We have divided the signal into 0.5 s segments. We have done the same thing to the word profile (Fig. 2).

Now we have 0.5 s input and 0.5 s output. If sampling frequency = 22,050/s then input shape = $22{,}050 * 0.5 = 11{,}025$ samples and output (word profile) shape = 11,025 samples.

The red dots in Fig. 3 are the 5 ms interval points. The sequence is 500 ms, so we have taken one value for every 5 ms interval which is 0 or 1. These are enough

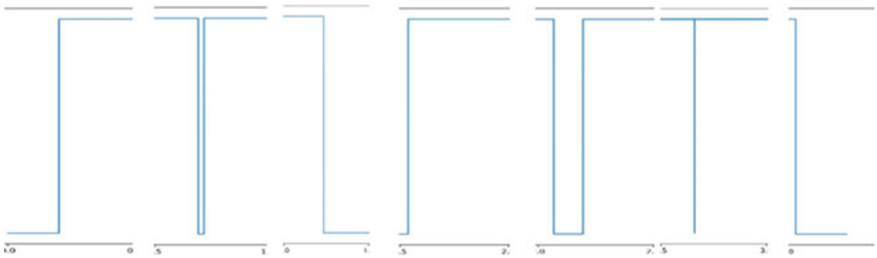


Fig. 2 0.5 s segment of a complete word profile

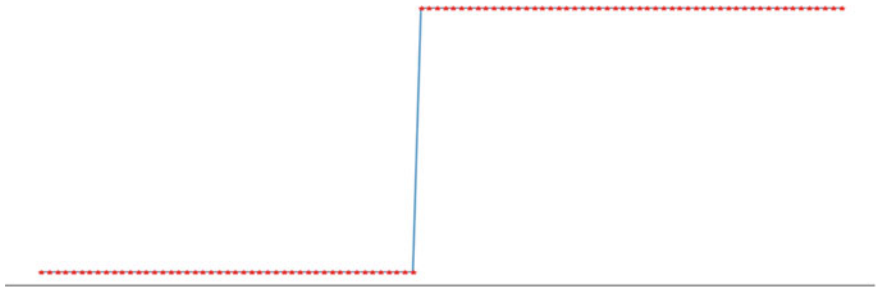
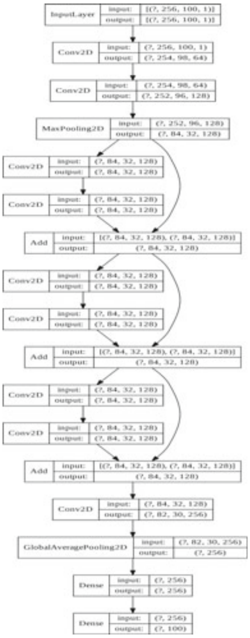


Fig. 3 Word profile with 5 ms interval points

values to represent the profile. So now, the number of points will be 500 ms /5 ms = 100 samples.

Finally, we have 11,025 numbers of points for the input signal for the model and 100 output points.

3.3 Model Architecture



3.4 Create Training Dataset

Now the model can be trained but there is still one problem left. Training the model with raw audio signals is not a good idea. We have to transform the audio from a time domain to any other domain.

The transformation options are: (1) FFT, (2) STFT, (3) Spectrogram, (4) Mel Spectrogram, (5) MFCC.

Among these Mel spectrogram gives the best result, so we have performed this transformation.

To get the 0.5 s Mel spectrogram we need to set the parameters:

window = 0.005 s = 5 ms, fs = 22,100.

We've used python library librosa to get the mel spectrogram.

Librosa [2] librosa.feature.melspectrogram(y = audio_samples_array, sr = fs, n_mels = 256, fmax = 8000, hop_length = int(fs*window), n_fft = 1024).

This gives a 256 × 100 matrix for every 0.5 s. Now we have a [256 × 100] matrix. This matrix is the input to the model.

4 Word Boundary Detection Using Decision Tree

This is also the same approach as CNN model. We have performed some transformations on the raw audio signal to get the features from the signals and we have used the word profiles as the labels for our model. In CNN based model the disadvantages are, this model takes too much time to train and lots of computation is required for the prediction. So instead of using CNN based model, we can use a Decision tree. In this method, we use the MFCC transformation for the raw audio signals. We have created training data from this MFCC matrix and given this matrix as input to the model and used the word profile as the output.

4.1 Create MFCC Matrix from Audio Signals

To get the MFCCs from the audio signal, we've performed the following steps:

- First, we break the audio signal into small 5 ms frames. So, if we take the signal which is sampled at 22050 Hz, we get 22,050*0.005 = 110 samples, If the speech file does not divide into an even number of frames, pad it with zeros so that it does.
- Then we take each frame and apply complex DFT to it. We take the absolute value of the complex Fourier transform and square the result. We would generally perform a 1024-point FFT and keep only the first 512 coefficients (this gives only the positive frequencies). This is known as periodogram power spectral.
- Compute the Mel-spaced filter bank. This is a set of 20–40 (26 is standard) triangular filters that we apply to the periodogram power spectral estimate from step 2. Our filter bank comes in the form of 26 vectors of length 512. Each vector is mostly zeros but is non-zero for a certain section of the spectrum. To calculate filter bank energies, we multiply each filter bank with the power spectrum, then add up the coefficients. Once this is performed, we are left with 26 numbers that give us an indication of how much energy was in each filter bank.
- Then take the log of each of the 26 energies from step 3. This leaves us with 26 log filter bank energies.
- Then we take the Discrete Cosine Transform (DCT) of the 26 log filter bank energies to give 26 cepstral coefficients. In our case, we take only the lower 20 of the 26 coefficients.

4.2 Creating Training Data for Decision Tree

Now we have the MFCC matrix as input and for the output we have a word profile, which is in binary form (0 or 1).