Advances in Sustainability Science and Technology

Amit Joshi
Atulya K. Nagar
Gabriela Marín-Raventós *Editors*

Sustainable Intelligent Systems

KES International

Springer

Advances in Sustainability Science and Technology

The book series aims at bringing together valuable and novel scientific contributions that address the critical issues of renewable energy, sustainable building, sustainable manufacturing, and other sustainability science and technology topics that have an impact in this diverse and fast-changing research community in academia and industry.

The areas to be covered are

- Climate change and mitigation, atmospheric carbon reduction, global warming
- Sustainability science, sustainability technologies
- Sustainable building technologies
- Intelligent buildings
- Sustainable energy generation
- Combined heat and power and district heating systems
- Control and optimization of renewable energy systems
- Smart grids and micro grids, local energy markets
- Smart cities, smart buildings, smart districts, smart countryside
- Energy and environmental assessment in buildings and cities
- Sustainable design, innovation and services
- Sustainable manufacturing processes and technology
- Sustainable manufacturing systems and enterprises
- Decision support for sustainability
- Micro/nanomachining, microelectromechanical machines (MEMS)
- Sustainable transport, smart vehicles and smart roads
- Information technology and artificial intelligence applied to sustainability
- Big data and data analytics applied to sustainability
- Sustainable food production, sustainable horticulture and agriculture
- Sustainability of air, water and other natural resources
- Sustainability policy, shaping the future, the triple bottom line, the circular economy

High quality content is an essential feature for all book proposals accepted for the series. It is expected that editors of all accepted volumes will ensure that contributions are subjected to an appropriate level of reviewing process and adhere to KES quality principles.

The series will include monographs, edited volumes, and selected proceedings.

More information about this series at http://www.springer.com/series/16477

Amit Joshi · Atulya K. Nagar ·
Gabriela Marín-Raventós
Editors

Sustainable Intelligent Systems

Springer

Editors
Amit Joshi
Global Knowledge Research Foundation
Ahmedabad, Gujarat, India

Gabriela Marín-Raventós
University of Costa Rica
San José, Costa Rica

Atulya K. Nagar
School of Mathematics
Computer Science and Engineering
Liverpool Hope University
Liverpool, UK

ISSN 2662-6829 ISSN 2662-6837 (electronic)
Advances in Sustainability Science and Technology
ISBN 978-981-33-4900-1 ISBN 978-981-33-4901-8 (eBook)
https://doi.org/10.1007/978-981-33-4901-8

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

Sustainable Intelligent Systems explains and aims at giving a gist to exigency of the technological solutions with a forward move towards sustainability. The book presents issues related to ICT, intelligent systems, data science, AI, machine learning, sustainable development and overall their impacts on sustainability and provides an overview of the technologies awaiting unveiling. The book also discusses novel intelligent algorithms and their applications to move from a data-centric world to sustainable world. The book also includes the research paradigms on SDGs and societal impacts. The book provides an overview of cutting-edge techniques towards sustainability and ideas to help researchers who want to understand the challenges and opportunities of using for Smart Management Prospective for Sustainable Society. It covers wide ranges of audience from computer science, data analysts, AI technocrats and management researchers.

A great pool of more than 100 papers were submitted for this volume by researchers related to the theme intelligent sustainable systems. This book appeals to a broad readership, including academics, researchers and industry professionals.

Ahmedabad, India	Amit Joshi, Ph.D.
Liverpool, UK	Atulya K. Nagar, Ph.D.
San José, Costa Rica	Gabriela Marín-Raventós, Ph.D.

Contents

Editors and Contributors

About the Editors

Amit Joshi is currently Director of Global Knowledge Research Foundation and also Entrepreneur Researcher who has completed his masters and research in the areas of cloud computing and cryptography in medical imaging. Dr. Joshi has an experience of around 10 years in academic and industry in prestigious organizations. Dr. Joshi is an active member of ACM, IEEE, CSI, AMIE, IACSIT, Singapore, IDES, ACEEE, NPA and many other professional societies. Currently, Dr. Joshi is International Chair of InterYIT at International Federation of Information Processing (IFIP, Austria). He has presented and published more than 50 papers in national and international journals/conferences of IEEE and ACM. Dr. Joshi has also edited more than 40 books which are published by Springer, ACM and other reputed publishers. Dr. Joshi has also organized more than 50 national and international conferences and programs in association with ACM, Springer and IEEE to name a few across different countries including India, UK, Europe, USA, Canada, Thailand, Egypt and many more.

Atulya K. Nagar holds Foundation Chair as Professor of Mathematical Sciences and is Pro-Vice-Chancellor for Research and Dean of the Faculty of Science at Liverpool Hope University, UK. He is also Head of the School of Mathematics, Computer Science and Engineering, which he established at the University. He is an internationally respected scholar working at the cutting edge of theoretical computer science, applied mathematical analysis, operations research and systems engineering. He received a prestigious Commonwealth Fellowship to pursue his doctorate (D.Phil.) in Applied Nonlinear Mathematics, which he earned from the University of York (UK) in 1996. He holds a B.Sc. (Hons.), an M.Sc. and M.Phil. (with distinction) in Mathematical Physics from the MDS University of Ajmer, India. His research expertise spans both applied mathematics and computational methods for nonlinear, complex and intractable problems arising in science, engineering and industry.

Gabriela Marín-Raventós received a M.Sc. in Computer Science from Case Western Reserve University in 1985 and a Ph.D. in Business Analysis and Research from Texas A&M University, USA, in 1993. She has been a Computer Science faculty member at Universidad de Costa Rica (UCR) since 1980. She was Dean of Graduate Studies and Director of the Research Center for Communication and Information Technologies (CITIC), both at UCR. Currently, she is Director of the Graduate Program in Computer Science and Informatics. She has organized several international and national conferences, and has been, and actually is, Chair of several program and editorial committees. From 2012 to 2016, she was President of the Latin American Center for Computer Studies (CLEI), becoming the first woman to occupy such a distinguished position. Since September 2016, she is Vice-President of the International Federation for Information Processing (IFIP), in charge of the Digital Equity Committee. Her research interests include smart cities, human–computer interaction, decision support systems, gender in IT and digital equity.

Contributors

Roberto O. Andrade Facultad de Ingeniería de Sistemas. Escuela Politécnica Nacional, Quito, Ecuador

Viacheslav Belenko LG Electronics Inc., Seoul, South Korea

David Bird The Institution of Engineering Technology, London, UK

Jose A. Brenes Research Center for Communication and Information Technologies (CITIC), University of Costa Rica, San José, Costa Rica

María Cazares IDEIAGEOCA Research Group, Universidad Politécnica Salesiana, Quito, Ecuador

Valery Chernenko LG Electronics Inc., Seoul, South Korea

Haejin Cho LG Electronics Inc., Seoul, South Korea

Franco Cicirelli CNR—National Research Council of Italy, Institute for High Performance Computing and Networking—(ICAR), Rende, Italy

César A. Collazos Universidad del Cauca, Popayán, Colombia

Paweł Dymora Faculty of Electrical and Computer Engineering, Rzeszów University of Technology, Rzeszów, Poland

Markus Eger Graduate Program (PPCI), University of Costa Rica, San José, Costa Rica

Mezzour Ghita National and High School of Electricity and Mechanic (ENSEM), HASSAN II University, Casablanca, Morocco;
Research Foundation for Development and Innovation in Science and Engineering, Casablanca, Morocco;
Innovation Lab for Operations (ILO), Mohammed VI Polytechnic University (UM6P), Ben Guerir, Morocco

Hafid Griguer Innovation Lab for Operations (ILO), Mohammed VI Polytechnic University (UM6P), Ben Guerir, Morocco

César D. Guerrero Universidad Autónoma de Bucaramanga, Bucaramanga, Colombia

Medromi Hicham National and High School of Electricity and Mechanic (ENSEM), HASSAN II University, Casablanca, Morocco;
Research Foundation for Development and Innovation in Science and Engineering, Casablanca, Morocco

Mihaiela Iliescu Department of Mechatronics and Robotics, Institute of Solid Mechanics of the Romanian Academy, Bucharest, Romania

Ashok Jetawat Pacific Academy of Higher Education & Research Society, Udaipur, Rajasthan, India

Manuj Joshi Pacific Academy of Higher Education & Research Society, Udaipur, Rajasthan, India

Mustapha Kabil Hassan II University, Casablanca, Morocco

Maxim Kalinin Peter the Great St.Petersburg Polytechnic University, St. Petersburg, Russia

Vasiliy Krundyshev Peter the Great St.Petersburg Polytechnic University, St. Petersburg, Russia

My Driss Laanaoui Cadi Ayyad University, Marrakesh, Morocco

Sangwon Lim LG Electronics Inc., Seoul, South Korea

Gina Maestre-Gongora Universidad Cooperativa de Colombia, Medellín, Colombia

V. V. Mahesh Division of Electronics Engineering, School of Engineering, Cochin University of Science and Technology, Kochi, India

Marvy Badr Monir Mansour Department of Electrical Engineering, Faculty of Engineering, The British University in Egypt, Cairo, Egypt

Gabriela Marín-Raventós Graduate Program (PPCI), University of Costa Rica, San José, Costa Rica

Mirosław Mazurek Faculty of Electrical and Computer Engineering, Rzeszów University of Technology, Rzeszów, Poland

Yurley Medina-Cárdenas Universidad Francisco de Paula Santander Ocaña, Ocaña, Colombia

Libero Nigro DIMES—Department of Informatics Modelling Electronics and Systems Science, University of Calabria, Rende, Italy

Iván Ortiz Universidad de Las Américas, Quito, Ecuador

Marius Pandelea Department of Mechatronics and Robotics, Institute of Solid Mechanics of the Romanian Academy, Bucharest, Romania

Shrikant Pawar Yale University, New Haven, CT, USA

Julio Proaño IDEIAGEOCA Research Group, Universidad Politécnica Salesiana, Quito, Ecuador

Corina Radu Frenţ Department of Mechatronics and Robotics, Institute of Solid Mechanics of the Romanian Academy, Bucharest, Romania

Alexander Raikov Institute of Control Sciences RAS, Moscow, Russia

Dewar Rico-Bautista Universidad Francisco de Paula Santander Ocaña, Ocaña, Colombia

Abdelrahman Said Department of Electrical Engineering, Faculty of Engineering, The British University in Egypt, Cairo, Egypt

T. K. Shahana Division of Electronics Engineering, School of Engineering, Cochin University of Science and Technology, Kochi, India

Benhadou Siham National and High School of Electricity and Mechanic (ENSEM), HASSAN II University, Casablanca, Morocco;
Research Foundation for Development and Innovation in Science and Engineering, Casablanca, Morocco

Mouad Tantaoui Hassan II University, Casablanca, Morocco

Isabela Todiriţe Department of Mechatronics and Robotics, Institute of Solid Mechanics of the Romanian Academy, Bucharest, Romania

Luige Vlădăreanu Department of Mechatronics and Robotics, Institute of Solid Mechanics of the Romanian Academy, Bucharest, Romania

Performance Analysis of Classification Algorithms Used for Software Defect Prediction

Manuj Joshi and Ashok Jetawat

Abstract The software quality and reliability are being the major issues while developing software, and as in the software project management, the major cost overruns are due to the deployment of bug prone software at the customer side. Mainly, warehouses are used for maintaining the databases regarding fixing the defects. In this research work, six different classification techniques were being selected based on previous researches which showed better results while developing efficient predictive model. The classification techniques being considered were Naive Bayes, multilayer perceptron, random forest, support vector machine, PART and simple CART, and these various techniques were being measured on various performance measure included as correctly classified instances or accuracy, recall, precision, ROC, F-measure, etc. The experimental analysis used the WEKA environmental tool with calibration setting of tenfolds cross-validation which was being applied on PROMISE repository datasets AR1, AR3, AR4, AR5 and AR6 so as to measure the performance of various algorithms on parameters like accuracy or correctly classified instances, recall, F-measure, precision and ROC measure values, and finally, the overall performance was being evaluated. For the all over performance evaluation, the method of average and correlation was being used. The findings suggest that support vector machine classification algorithm was the best and most promising when applied on software defect prediction supported by the highest average accuracy measure of 87.22%. Similarly, other measures like recall, F-measure, precision have higher values of 0.872, 0.847 and 0.857, respectively.

Keywords PART · Support vector machine · Naive Bayes

1 Introduction

Classification approach used for software defect prediction is basically based on finding an appropriate predictive model which divides the dataset into different

M. Joshi (✉) · A. Jetawat
Pacific Academy of Higher Education & Research Society, Udaipur, Rajasthan, India
e-mail: manujjoshi@gmail.com

A. Joshi et al. (eds.), *Sustainable Intelligent Systems*, Advances in Sustainability Science and Technology, https://doi.org/10.1007/978-981-33-4901-8_1

classes. In this approach, the unclassified data elements are being classified by providing a particular class label based on the developed model. Applying the classification algorithms in software defect prediction helps to identify defect in the code and find whether the code is defect free or not so that it can further be reused for software development. The test dataset is being classified based on the existing trained model. In this paper, various classification algorithms are being compared and analyzed on different software defect datasets.

The efficient prediction models used for software defect identification are required to find the defects in the code so that by proper approach the latent defects can be rectified before the deployment of the software after post delivery the cost of bug correction is quite high as compared to pre-delivery. Almost every year billion of dollars are being spent on bug fixing or correction. Here, the efficiency of the predictive models comes in play if the models are having high accuracy then these can save large amount of money for the software companies as in case small bug identification and correction can save huge amount of cost.

In the research work, basically six data mining classification algorithms were being chosen, the selection was based on the previous research works being conducted for building an effective predictive model, so the algorithms have better performances were being considered. The paper included the classification algorithms like multilayer perceptron, Naive Bayes, random forest, support vector machine, PART and simple CART which were analyzed on various performance measures or parameters like precision, accuracy, recall, ROC, F-measure, etc. For evaluating the accuracy, the total number of correctly classified instance was being taken in consideration, and for error value, the incorrectly classified instances were taken in consideration. The total performance score was being calculated using the average and correlation methods. According to correlation method, the measures showing either positive or negative values were being used for further analysis, and the remaining were not being taken under consideration.

2 Background

Software products are today used in every functional area of the business organizations. According to the present scenario, software quality plays an important role, and it can be defined as the parameter to evaluate design and implementation aspects. The previous researchers suggest many measures based on attributes for evaluating software quality, and the various factors or attributes are product quality, accuracy (correctness), scalability, error free, etc. Although the quality measure adopted by one organization may differ from the other organization, so software metrics standard measures should be used for quality assessment. The software defect predictor can use the software metrics parameters as input to obtain the software quality. The study covers the application of machine learning classifier algorithms to predict the software quality [1].

Quality and reliability are being considered as the major issues while developing software. According to software project management, the major cost overruns are due to the deployment of bug prone software at the customer side. Mainly, software warehouses are used for maintaining the databases regarding fixing the defects. The research work focused on identifying the causes of software failures in recent past with detailed literature review. The timely identification of these bugs requires the use of data mining techniques and also emphasized on the development of efficient bug tracking system. The author suggests that bug repositories play important role in bug tracking system which contains the history of particular codes success or failure, the prediction models related to bug fix time are also useful for marinating the quality and reliability of the software products [2].

According to the Peng experimental study which focused on the software bug prediction using the software measures for the evaluation of the classification algorithm. The approach adopted by author mainly aimed at performance measure evaluation in the study experiment was conducted on ten open projects (that were 34 releases) of the PROMISE repository. The major finding in the study suggested that top-k parameters or measure gives better outcomes as evaluated to standard predictors [3].

Software defect prediction is one of the important aspects in decreasing the costs of the software design, development & maintaining the standards of quality assurance for building quality software solutions. Mainly, classification approaches including the extraction by features and categorization in classes are basically applied to identify bugs or defects which are present in software code at the time of testing conducted in each phase. To improve the accuracy of software bug prediction model, various algorithms and techniques which are based on statistical and mathematical models are being used particularly in the feature extraction phase. Lastly, it can be suggested that the process actually reduces the cost by applying the prediction model as compared to the previous models having lower accuracy measure values and also helps in finding the most promising algorithm for software defect prediction [4].

Software defect prediction is an approach in which an appropriate prediction model is being created so as to predict the future outcomes based on the software faults previous dataset as obtained from software warehouse repository. There are many approaches for developing the prediction model, the author proposed NB, DT and ANNs techniques which were being used or applied on various experimental datasets, and also, various performance measures were being used for evaluating the classifiers. The study basically evaluated the above-mentioned three machine learning algorithms in software defect prediction problem or case. The evaluation process was conducted using three real-time datasets. The performance measures like accuracy, mean absolute error, precision, recall, F-measure, ROC and RMSE were being evaluated for all the three datasets, and the results revealed that machine learning classification algorithms are the best approaches for developing a predictive model so as to maintain the software quality. When comparative analysis was done between the three algorithms, it was found that the decision tree (DT) classifier was the best or appropriate over the others algorithms. The outcomes suggested that machine learning approaches like the classifier are far more better than the AR linear approach and POWM model [5].

3 Methodology

The proposed work includes the analysis of six different classification algorithms which are mainly applied for the software defect prediction, and according to the previous studies, the most appropriate algorithms are Naïve Bayes, multilayer perceptron, random forest, support vector machine, PART & simple CART. Selection of the classification techniques was based on the concept that they are built on different mathematical models along with different properties as the Naive Bayes approach is based on the concept of combined probability in which dependent variable being related to various categories of independent variables, and the algorithm requires both type variables to be categorial. The algorithm PART is basically adopted from CART approach which is classification & regression trees, and in this approach, particularly a decision tree is being formed based on the measure information entropy by subdividing the datasets as training based on various independent variables. When considering the SVM classifier, the model develops a hyper-plane that basically categorize the training datasets into two different class labels. Random forest is basically a collaborative technique which combines various CARTs based on various features.

The proposed methodology includes the following steps:

Firstly, the identification & data collection related to software defect prediction datasets which consists of attributes Blank_LOC, Total_LOC, Unique_operands, Comment_LOC, Code_and_comment_LOC, etc., related to software measures.

The second step includes data pre-processing consisting of cleaning, transformation, etc. The data cleaning includes the removal of missing or inconsistent data from the dataset, while transformation is used to convert datasets in different formats to the desired format which is being further considered as input to the WEKA tool.

Thirdly, the experimental environment is basically used for applying the six suggested classification algorithms on five software defect prediction datasets using tenfold cross-validation testing configuration as the algorithm calibration for measuring, analyzing & comparing the algorithms on six different parameters.

In the fourth step, various performance analysis measures such as correctly classified instances or accuracy, incorrectly classified instances, recall, F-measure, ROC curve values were being obtained for further evaluation. Mainly, the observations were taken at tenfold cross-validation configuration settings that is datasets were portioned using random sample partition method by using WEKA tool.

Finally, the best algorithms would be selected based on various performance measure comparison, averages and correlation methods.

The experiment was being conducted on PROMISE repository datasets AR1 [6], AR3 [7], AR4 [8], AR5 [9] and AR6 [10] with configuration settings tenfold cross-validation on the WEKA platform. The observations were taken separately for all the five datasets from software units. All the different algorithms were being individually compared based on different performance measures. Lastly, the overall score of performance is being obtained to rank the algorithms.

4 Results

According to the experimental analysis, the outcomes as shown in the table below suggest that when applying the configuration setting with cross-validation of tenfolds on the dataset or database AR1, it was found that support vector machine was the best data mining classification algorithm with the highest accuracy or correctly classified instances with value of 91.73%. The second most promising classification algorithm was found to be the rule-based PART which was having the accuracy measure value of 90.90%. Further applying the other measure like mean absolute error on the classification algorithms, it was obtained that the accuracy measure and mean absolute error value justify each other as support vector machine has the lowest mean absolute value of error that was calculated as 0.0826 (Table 1).

The table below for the dataset AR3 suggests that accuracy measure of multilayer perceptron was found to be 93.65% which is higher than the other algorithms. Similarly, the algorithms random forest and Naive Bayes have the accuracy values 92.063 and 90.48%, respectively. Here, for this particular dataset, the support vector machine algorithm has shown lower accuracy with value 88.89% as compared to the dataset AR1. According to the mean absolute error (MAE) measure, the least value was calculated to be of Naive Bayes algorithm with 0.1085, and similarly, the multilayer perceptron algorithm has also shown lower value of 0.1101. The performance evaluation results for AR3 dataset reveal that Naive Bayes algorithm &

Table 1 Classifiers performance analysis with cross-validation folds (10) on AR1 dataset

S. No.	Classifier	Accuracy (%)	Mean absolute error	Precision	Recall	F-measure	ROC area
1	Naïve Bayes	85.12	0.1519	0.899	0.851	0.871	0.688
2	Multilayer perceptron	90.08	0.1037	0.89	0.901	0.895	0.701
3	Random forest	90.08	0.1289	0.855	0.901	0.877	0.641
4	Support vector machine	91.73	0.0826	0.856	0.917	0.886	0.496
5	PART	90.90	0.1048	0.914	0.909	0.911	0.756
6	Simple CART	90.08	0.1583	0.855	0.901	0.877	0.417

multilayer perceptron algorithms were the most appropriate algorithms in software defect prediction (Table 2).

According to the experimental outcomes generated for the dataset AR4, it is observed that correctly classified instances or accuracy value for support vector machine was obtained to be the most appropriate and best having value 85.05% and also has the lowest mean absolute value of 0.1495 as compared to other classification algorithms. Similarly, the second best algorithm was found to be Naive Bayes having the accuracy and MAE values 84.11% and 0.153, respectively (Table 3).

Table 2 Classifiers performance analysis with cross-validation folds (10) on AR3 dataset

S. No.	Classifier	Accuracy (%)	Mean absolute error	Precision	Recall	F-measure	ROC area
1	Naïve Bayes	90.4762	0.1085	0.916	0.905	0.909	0.824
2	Multilayer perceptron	93.6508	0.1101	0.933	0.937	0.933	0.748
3	Random forest	92.0635	0.1651	0.914	0.921	0.913	0.718
4	Support vector machine	88.8889	0.1111	0.87	0.889	0.866	0.616
5	PART	88.8889	0.142	0.883	0.889	0.886	0.567
6	Simple CART	88.8889	0.1568	0.87	0.889	0.866	0.639

Table 3 Classifiers performance analysis with cross-validation folds (10) on AR4 dataset

S. No.	Classifier	Accuracy (%)	Mean absolute error	Precision	Recall	F-measure	ROC area
1	Naïve Bayes	84.1121	0.153	0.828	0.841	0.832	0.809
2	Multilayer perceptron	81.3084	0.2081	0.799	0.813	0.805	0.71
3	Random forest	82.243	0.2187	0.792	0.822	0.794	0.794
4	Support vector machine	85.0467	0.1495	0.838	0.85	0.823	0.639
5	PART	81.3084	0.2046	0.793	0.813	0.8	0.603
6	Simple CART	81.3084	0.26	0.787	0.813	0.794	0.69

Table 4 Classifiers performance analysis with cross-validation folds (10) on AR5 dataset

S. No.	Classifier	Accuracy (%)	Mean absolute error	Precision	Recall	F-measure	ROC area
1	Naïve Bayes	83.3333	0.1666	0.851	0.833	0.84	0.907
2	Multilayer perceptron	69.4444	0.2665	0.708	0.694	0.701	0.799
3	Random Forest	83.3333	0.2	0.833	0.833	0.833	0.893
4	Support vector machine	83.3333	0.1667	0.833	0.833	0.833	0.759
5	PART	83.3333	0.2018	0.833	0.833	0.833	0.717
6	Simple CART	77.7778	0.2642	0.778	0.778	0.778	0.676

The accuracy measure values for support vector machine, random forest, PART & Naive Bayes were found to be highest when applied on dataset AR5 with values 83.33% each, but support vector machine and Naive Bayes were only having the lowest mean absolute error values of 0.167 each (Table 4).

The table below for the dataset AR6 suggests that accuracy value of support vector machine was obtained to be 87.13% which is higher than the other algorithms. Similarly, the algorithm random has the accuracy values 84.15%. According to the mean absolute error measure, the lowest value was obtained to be of the support vector machine (SVM) algorithm having value 0.1287, and similarly, the Naïve Bayes algorithm has also shown lower value of 0.1744. The performance evaluation results for AR6 dataset reveal that SVM algorithm was observed to be the most appropriate algorithm in software defect prediction (Table 5).

The overall performance evaluation based on average and correlations suggests that support vector machine classification algorithm is the most appropriate algorithm for the software defect prediction as it has the highest average accuracy value of 87.22%, lowest MAE error value of 0.1277 and high precision, F-measure & recall values of 0.857, 0.847 and 0.872, respectively, as compared with other algorithms (Table 6).

5 Conclusion

The results conclude that for the above-mentioned datasets of software defect prediction the accuracy and mean absolute error performance measures for all classification algorithms have shown variation. The accuracy and mean absolute error follow the negative correlation between them. In all the cases except AR3 dataset, it was found

Table 5 Classifiers performance analysis with cross-validation folds (10) on AR6 dataset

S. No.	Classifier	Accuracy (%)	Mean absolute error	Precision	Recall	F-measure	ROC area
1	Naïve Bayes	82.1782	0.1744	0.812	0.822	0.816	0.652
2	Multilayer perceptron	82.1782	0.2074	0.812	0.822	0.816	0.751
3	Random forest	84.1584	0.2198	0.806	0.842	0.816	0.663
4	Support vector machine	87.1287	0.1287	0.888	0.871	0.827	0.567
5	PART	80.198	0.235	0.764	0.802	0.781	0.526
6	Simple CART	84.1584	0.2573	0.724	0.842	0.778	0.397

Table 6 Overall performance based on average measure including all datasets

S. No.	Classifier	Accuracy (%)	Mean absolute error	Precision	Recall	F-measure	ROC area
1	Naive Bayes	85.04	0.1509	0.8612	0.8504	0.8536	0.776
2	Multilayer perceptron	83.33	0.1792	0.8284	0.8334	0.83	0.7418
3	Random forest	86.37	0.1865	0.84	0.8638	0.8466	0.7418
4	Support vector machine	87.22	0.1277	0.857	0.872	0.847	0.6154
5	PART	84.92	0.1776	0.8374	0.8492	0.8422	0.6338
6	Simple CART	84.44	0.2193	0.8028	0.8446	0.8186	0.5638

that support vector machine classification algorithm is the most appropriate algorithm for the software defect prediction. The results were being further justified by comparing the algorithms based on average overall performance which were in accordance with support vector machine classification algorithm being the best algorithm with highest average accuracy value of 87.22% also taking in consideration the measure mean absolute error value which was found to be lowest with 0.1277. Similarly, the other measures recall, F-measure and precision were found to be on higher side with values 0.872, 0.847 and 0.857, respectively.

References

1. F. Akmel, E. Birihanu, B. Siraj, A literature review study of software defect prediction using machine learning techniques. Int. J. Emerg. Res. Manage. Technol. **6**, 300 (2018). https://doi.org/10.23956/ijermt.v6i6.286
2. T. Hall, S. Beecham, D. Bowes, D. Gray, S. Counsell, A systematic literature review on fault prediction performance in software engineering. IEEE Trans. Softw. Eng. **38**(6), 1276–1304 (2012)
3. P. He et al. An empirical study on software defect prediction with a simplified metric set. Inf. Softw. Technol **59**, 170–190 (2015)
4. A. Parameswari, Comparing data mining techniques for software defect prediction. Int. J. Sci. Eng. Res. (IJ0SER) **3**(5), 3221–5687 (2015).
5. A. Hammouri, M. Hammad, M. Alnabhan, F. Alsarayrah, Software bug prediction using machine learning approach. Int. J. Adv. Comput. Sci. Appl. **9**(2), 78–83 (2018)
6. G. Boetticher, T. Menzies, T. Ostrand, AR1/Software Defect Prediction Dataset (Classification), PROMISE Repository of Empirical Software Engineering Data https://promisedata.org/repository (West Virginia University, Department of Computer Science, 2007)
7. G. Boetticher, T. Menzies, T. Ostrand, AR3/Software Defect Prediction Dataset (Classification), PROMISE Repository of Empirical Software Engineering Data https://promisedata.org/repository (West Virginia University, Department of Computer Science, 2007)
8. G. Boetticher, T. Menzies, T. Ostrand, AR4/Software Defect Prediction Dataset (Classification), PROMISE Repository of Empirical Software Engineering Data https://promisedata.org/repository (West Virginia University, Department of Computer Science, 2007)
9. G. Boetticher, T. Menzies, T. Ostrand, AR5/Software Defect Prediction Dataset (Classification), PROMISE Repository of Empirical Software Engineering Data https://promisedata.org/repository (West Virginia University, Department of Computer Science, 2007)
10. G. Boetticher, T. Menzies, T. Ostrand, AR6/Software Defect Prediction Dataset (Classification), PROMISE Repository of Empirical Software Engineering Data https://promisedata.org/repository (West Virginia University, Department of Computer Science, 2007)

Early Detection of Diseases in Precision Agriculture Processes Supported by Technology

Jose A. Brenes, Markus Eger, and Gabriela Marín-Raventós

Abstract One of the biggest challenges for farmers is the prevention of disease appearance on crops. Governments around the world control border product entry to reduce the number of foreign diseases affecting local producers. Evenmore, it is also important to reduce the spread of crop diseases as quickly as possible and in early stages of propagation, to enable farmers to attack them on time, or to remove the affected plants. In this research, we propose the use of convolutional neural networks to detect diseases in horticultural crops. We compare the results of disease classification in images of plant leaves, in terms of performance, time execution, and classifier size. In the analysis, we implement two distinct classifiers, a densenet-161 pre-trained model and a custom created model. We concluded that for disease detection in tomato crops, our custom model has better execution time and size, and the classification performance is acceptable. Therefore, the custom model could be useful to use to create a solution that helps small farmers in rural areas in resource-limited mobile devices.

Keywords Diseases detection · Precision agriculture · Machine learning · Feature selection · Convolutional neural networks

1 Introduction

Currently, the world economy is facing big challenges. Today, most countries are involved in a socioeconomic slowdown, augmented by the COVID-19 pandemic,

---

J. A. Brenes (✉)
Research Center for Communication and Information Technologies (CITIC), University of Costa Rica, 11501 San José, Costa Rica
e-mail: joseantonio.brenes@ucr.ac.cr

M. Eger · G. Marín-Raventós
Graduate Program (PPCI), University of Costa Rica, 11501 San José, Costa Rica
e-mail: markus.eger@ucr.ac.cr

G. Marín-Raventós
e-mail: gabriela.marin@ucr.ac.cr

A. Joshi et al. (eds.), *Sustainable Intelligent Systems*, Advances in Sustainability Science and Technology, https://doi.org/10.1007/978-981-33-4901-8_2

requiring all sectors to innovate. In Costa Rica, for example, the agricultural sector had a growth equivalent to the 3.7% of GDP, in 2017. Meanwhile, in 2018, the growth of the sector was only 2.4% of GDP. That represents a decrease in the contribution to the country's economy compared to the previous year [1]. Governments are expecting bigger decrements due to the pandemic recession. It is important to consider that with the reported decrease in production, a lot of families saw, in the last years, and will soon see their income reduced. According to the Costa Rican Performance Report of the Agricultural Sector [1], the reported GDP growth in 2018 was possible due to improvements in agricultural activities related to pineapple crops. In that area, farmers applied actions to control pests and diseases affecting pineapple crops fields.

Tomato and bell pepper crops are two of the most produced crops in the world [2]. In 2018, both crops reached high production rates in the order of 182 and 36 million tons of tomatoes and bell peppers, respectively.[1] In Costa Rica, tomatoes, and bell peppers are the two most consumed vegetables, reaching an intake of about 18.2 and 3.3 kg per capita of tomato and bell pepper, respectively [3, 4].

Nevertheless, even though horticulture is particularly important in Costa Rica, Ramírez and Nienhuis [5] emphasize on pests and diseases suffered by these types of crops. Pests and diseases force farmers to apply chemical pesticides with harmful consequences for the environment. Thus, detecting them as early as possible is especially important in order to minimize the need for these chemicals.

In addition, Abarca [6] performed an analysis of climate change and plagues in crops cultivated in the tropics. He mentioned that it is crucial to create solutions that enable early alarms against disruptive plagues. In this way, it can be possible to react fast to the plague and avoid the appearance of the disease with a high impact in crops.

Dimitrios et al. [7] state that tools implementing emergent technologies to support farmers in pest and disease management are imperative now more than ever. The high incidence of pests and plagues on horticultural crops forces us to join efforts to create tools focused on this type of crops. These solutions must enable farmers to act immediately. For practical applications, we must also consider that farms may not have ready access to powerful hardware. An important consideration is therefore also to provide solutions that can run with very little resources, such as on a standard smartphone device.

There are some IT research efforts around the world focused on contributing to the agricultural productive sector, but mainly on automatic fertigation [8], while pest and disease management have been less studied. We center our research on this area. We propose the use of machine learning techniques to detect disease appearance on crops early to enable farmers to attack them on time or to remove the affected plants avoiding its spread. To achieve it, we keep in mind all the time, that the resulting solution must have to be runnable on mobile devices with very low resources to make it accessible to small-scale farmers in rural areas.

This paper is organized as follows. In the next section, we define some concepts needed to understand our research. In the following section, we mention different studies made by other researchers that are related to our research. Afterward, we

[1] https://www.fao.org/faostat/en/#data/QC.

present statistical analysis to select the features and artificial intelligence techniques to use in the tool to be provided to small farmers. Next, we present a study case in which we evaluate the feasibility of the proposal by comparing two different convolution neural networks in terms of performance, the training execution time, and the model's resultant size. Finally, we present the conclusions of this research process.

2 Theoretical Framework

In this section, we introduce some technical terms that must be clear to achieve a comprehension of our research.

2.1 Precision Agriculture

Dwivedi et al. [9] define precision agriculture as a farming managing method based on observing, measuring, and responding to variability in crops. The goal of precision agriculture is to define a decision support system (DSS) that can be used to manage the production optimizing its processes and increasing its yield.

Bongiovanni and Lowenberg-DeBoer [10] mention that precision agriculture is based on the idea of automation of site-specific management (SSM). This is achieved by using information technology to tailor input use to reach desired outcomes or to monitor those outcomes. Precision agriculture merges technologies with agricultural processes to achieve an integrated crop management [11].

2.2 Horticultural Crops

The horticultural crops are defined as herbaceous plants cultivated with the goal of self-consumption as well as for commercialization in internal and external markets [12]. Vegetables are particularly important crops for mankind because they are an extraordinarily rich source of vitamins and nutrients. They cover crops cultivated directly in the land as lettuce, celery, and parsley; tubers including potatoes and onions; fruits such as tomatoes and bell peppers; and flowers such as broccolis and artichokes [12]. Crops that will be used in our research are described next.

Bell Pepper

Bell peppers belong to the family of Solanaceae. *Capsicum annuum* is the most cultivated species in countries located in the tropics [13]. According to [2], bell peppers are one of the most frequently produced vegetables in Costa Rica. It is cultivated in an opencast way, but it is also produced inside greenhouses. Currently, producers can get access to a hybrid version known as "Dulcitico" developed by the researcher Carlos R. Echandi at the Fabio Baudrit Moreno Agricultural Experimental Station (EEAFBM) of the University of Costa Rica in 2013 [3]. This permits producers to get access to high quality, cheap seeds locally. "Dulcitico" is a highly productive hybrid which is also adapted climatologically to the region [14].

Tomato

Tomato crops also belong to the family of Solanaceae. The more cultivated species is *Solanum lycopersicum,* which has its origins in the Andes region [15]. The tomato plant is a perennial herb that can be cultivated annually [16]. Its fruits are harvested for consumption. According to [15], tomato is the most cultivated and consumed vegetable in Costa Rica.

Under adverse climatic conditions (high temperatures and humidity), tomato crops suffer many problems related to the presence of diseases which affect their production [16]. Diseases cause low-yield and quality and can even cause total crop losses. Authors point out that due to diseases' aggressiveness, it is important to prevent their appearance.

2.3 Diseases in Crops

Diseases are the main limiting factor in the production of vegetables in the tropics, according to [17]. In some cases, such as in tomato crops, there are pathogens resistant to existing treatments. Since plants do not respond to agrochemicals, farmers must remove affected plants from crops to avoid a greater affectation.

The high incidence of disease appearance is due to the use of seeds created for regions distinct to the one in which the crops are cultivated [17]. For example, farmers sometimes use seeds created for temperate zones in tropic regions. This limits the plants' resistance to diseases since crops are not adapted to the tropical climatic conditions.

The creation of hybrid plants, like "Dulcitico" in Costa Rica, enables farmers to fight some diseases. Nevertheless, even though we have seeds prepared for the prevailing climatic conditions, the best way to deal with pests and diseases is the prevention and know-how acquired over time [18].

It is important to provide farmers with tools for the early detection and control of diseases, so that they can reduce the uncontrolled propagation of diseases and can avoid loss of production. Disease detection is regularly done by examining the plants, and especially their leaves. This can be done manually but requires time and expertise. It can also be automated.

2.4 Computer Vision

Artificial intelligence (AI) techniques focused on computer vision and machine vision have been used in agriculture [19]. These techniques represent an alternative for the identification of diseases in crops achieving good results. The techniques used are based on the extraction of information (features) from images (photographs). Image information is used to create classifiers that permit categorizing new samples of images [19].

Tools that use computer vision techniques enable farmers to make a better use of pesticides [20]. When farmers detect a possible specific disease in the crop, they can apply the correct pesticide and control the quantity to apply according to the infection. Both parameters can be obtained automatically or semi-automatically through trained computer tools.

2.5 Classification Algorithms

In addition to computer vision, there are other techniques in the field of artificial intelligence that can be useful to classify images based on crop data. Many techniques are used to create classifiers which can be used to detect diseases in plant leaves images [21].

In this paper, we propose the use of techniques such as Naïve Bayes, linear discriminant analysis, and neural networks to classify images of plant leaves affected by diseases.

2.6 Image Classification

To classify images based on computer vision and machine learning techniques, it is required to extract information from the images to be analyzed. Features extracted from images represent characteristics associated with color, shape, and texture. Those features are used by algorithms in the learning process to carry out the classification.

It is possible to extract the following features from images:

- Color: It can be obtained by using different color models:
  - Color (RGB): The commonly used three channels representing the combination (additive) of colors Red, Green, and Blue. Each color is represented by a decimal number in the range 0–255. The color combinations can be analyzed as a cube in which the three dimensions are the three colors (RGB) [22].
  - Color (CYMK): This is a representation by using complementary colors (Cyan, Magenta, Yellow, and Black). This representation is mostly used for output devices like printers [22].

- Color (HSV): This is a representation of colors in a way that is more closely aligned with the way human vision perceives color making attributes [22]. The H represents the hue and takes values from 0 to 179; the S corresponds to the saturation which takes values from 0 to 255, and the V is the value in the range 0–255. Another representation uses lightness instead of value (in the same range). That last representation is referred to as HSL.
- Color Grayscale: This corresponds to a model that represents the amount of light in an image. The value for pixels can be in the range 0–255.

- Texture: texture is recognizable in tactile and optical ways. Texture analysis in computer vision is complex. According to [23], texture in images can be defined as a function of spatial variation of the brightness intensity of the pixels. To calculate the texture of the images, it is possible to use the Haralick method [24] with which it can be obtained 13 values representing textural features.
- Shape: for the calculation of the shape in the images, we use the Hu-moments invariants which are seven numbers, known as moment invariants, which are calculated by using weighted average of image pixel intensities. According to [25], moment invariants store all information about shapes in an image, so they can be used for shape recognition in images.

In the next section, we describe some of the existing techniques to help farmers detect diseases.

3 Related Work

The use of technological tools in agriculture is an active research field. Precision agriculture seeks to maximize agricultural production using technological solutions. This research area has been growing in the last years. Researchers around the world have been dedicated to the creation of systems and applications which enable farmers to process data collected from crops to transform it into valuable information for decision making.

Studies seeking the automation of irrigation in crops using data sensed from the cropfields in real time can be found in the literature. Meanwhile, other researchers focus on pest and disease prevention, control, and avoidance [8]. It is possible to find studies which propose automatic detection of diseases in tomato crops [26, 27], wheat [28, 29], grapes [28], among others. All these studies have similarities. They all propose solutions where real-time image processing is used to detect diseases affecting crops. Some of the mentioned studies use automatic analysis of plant leaves using machine learning and computer vision techniques, but in these cases, authors are more concerned on the precision and do not focus on the complexity of solutions and their computational cost.

The research presented in [27] shows a completely automated solution that includes from the image capturing process to the classification results. Nevertheless, even though the solution does not require farmers' participation, the system performance is quite similar to the one we propose in this research. So, we consider it is not necessary to automate all the process. Farmers can identify affected plants and take some pictures to feed the disease detection system. Additionally, putting static cameras in the crop field may be very expensive, and if few are used, it reduces the possibility to capture pictures of infected plants out of camera range.

Authors in [30, 31] analyze distinct techniques to recognize species through leaves color images analysis. A combination of techniques with which the researchers process color, shape, and texture features extracted from images to classify them is presented in [30]. Meanwhile, in [31], authors present a novel method to recognize plant species even with fragmented images, i.e.,, when the complete plant leaf is not present in the image. Both studies result relevant to this work. In both cases, the authors keep in mind a low computational cost requirement, a goal we also are seeking. Our research is somehow different. We are working with the classification of plant leaf images but we focused on disease detection present in leaves and not trying to determine the identity of the plant leaves belong to. We propose to do so analyzing only the color feature and by using convolution neural networks.

Other authors [32, 33] focus their studies on the creation of mobile applications to execute the identification of diseases. They depict that processing power in mobile devices results in a limitation. For that reason, authors recommend considering it in the creation of the solutions. A strategy to mitigate the lack of processing power is the use of cloud computing. However, we consider that solutions including data analysis in the cloud limit their use in rural areas in developing countries. In this setting, there is commonly no Internet connectivity or it is very poor.

It is also possible to find studies focusing on specific crops. First, a literature review of published proposals related to disease detection in bell pepper crops is presented in [34]. They conclude that in case of fungal diseases, automatic detection enables farmers to avoid high affectation by removing the affected plants (fungal diseases spread through spores that travel in the air).

In our research, we evaluate and propose an efficient algorithm for early detection of diseases in horticultural crops. The value of our proposal is related to our focus on two main requirements: mobile devices with very limited resources and local processing. These two requirements can enable farmers to use the solution without Internet access. Our focus also lies on the automation of the identification process, not in the control process. Our goal is to provide suggestions to farmers to help them act quickly.

4 Feature Selection for Automatic Diseases Detection: An Experiment

In 2019, a bell pepper crop (*capsicum annuum*) inside a greenhouse at EEAFBM was suffering the presence of a disease with high affectation and an extremely fast

Fig. 1 Example of leaves from the analyzed crop

propagation. The disease was causing plants death and a lot of problems for the researchers.

Leaves from healthy and infected plants are different. It is possible to see that the leaves of disease affected plants present degradation of colors in some rounded formations across the leaf. That affectation can also be present in other parts of the plant like the stem and the fruits. Additionally, it was possible to note that rounded formations in leaves had a different texture in comparison with the healthy sections of the leaves. In Fig. 1, we show an example of a healthy leaf (Fig. 1a) and an affected leaf (Fig. 1b), both taken from the mentioned crop. In Fig. 1b, it is possible to see the described affectation, with the rounded formations.

Different diseases exhibit different visual markers on the leaves. Thus, computer vision techniques could be used for image classification to identify the presence of a disease, as well as to distinguish between different diseases. To do that we started exploring the different features that can be extracted from images to describe them. We decided to use color (RGB), texture, and shape for the analysis, considering that the disease affects leaf color, creates circles (rounded formations), and changes the texture of the leaves. We designed an experiment to identify which feature (or combination of features) provided the best results. Different artificial intelligence techniques were used for the classification task. We also wanted to identify if there is a significant difference in which technique to use for the classification.

4.1 Objectives

The proposed three specific objectives for the experiment were:

- To determine if the features extracted from an image can affect the detection of a disease in an affected crop leaf in terms of the classification precision.
- To identify if the used classification technique can affect the precision in the detection of a disease in an image of an affected leaf.
- To determine if there exists an interaction between the features extracted from an image and the techniques used for the classification task.